

---

**pbpstats**

**Sep 16, 2023**



---

# Contents

---

<b>1</b>	<b>Features</b>	<b>3</b>
<b>2</b>	<b>Quickstart Guide</b>	<b>5</b>
2.1	Installation . . . . .	5
2.2	Setup data directory (optional but recommended) . . . . .	5
2.2.1	Download data . . . . .	5
2.2.2	data.nba.com vs stats.nba.com . . . . .	5
2.3	Basic Usage . . . . .	6
2.4	Code Examples . . . . .	6
2.4.1	Game Data . . . . .	6
2.4.2	All Final Games For Season . . . . .	7
2.4.3	All Final Games For Day . . . . .	7
2.4.4	Doing Detailed Stuff with Possession Data . . . . .	7
2.4.5	Doing Detailed Stuff with Enhanced PBP Data . . . . .	7
2.5	Note on Ids . . . . .	8
2.6	Issues with raw play-by-play . . . . .	8
<b>3</b>	<b>Contents</b>	<b>9</b>
3.1	Documentation . . . . .	9
3.1.1	Data Loaders . . . . .	9
3.1.1.1	data.nba.com Data Loaders . . . . .	9
3.1.1.2	live Data Loaders . . . . .	15
3.1.1.3	stats.nba.com Data Loaders . . . . .	19
3.1.1.4	Data Loader Factory . . . . .	29
3.1.1.5	Enhanced PBP Loader . . . . .	30
3.1.1.6	Possession Loader . . . . .	30
3.1.2	Objects . . . . .	30
3.1.2.1	Day . . . . .	30
3.1.2.2	Game . . . . .	31
3.1.2.3	Season . . . . .	31
3.1.3	Resources . . . . .	32
3.1.3.1	Boxscore . . . . .	32
3.1.3.2	Enhanced PBP . . . . .	34
3.1.3.3	Games . . . . .	59
3.1.3.4	PBP . . . . .	60
3.1.3.5	Possessions . . . . .	62
3.1.3.6	Shots . . . . .	63

3.1.4 Client . . . . .	64
<b>Python Module Index</b>	<b>65</b>
<b>Index</b>	<b>69</b>

A package to scrape and parse NBA, WNBA and G-League play-by-play data.



# CHAPTER 1

---

## Features

---

- Adds lineup on floor for all events
- Adds detailed data for each possession including start time, end time, score margin, how the previous possession ended
- Shots, rebounds and assists broken down by shot zone
- Supports both stats.nba.com and data.nba.com endpoints
- Supports NBA, WNBA and G-League stats
- All stats on pbpstats.com are derived from these stats
- Fixes order of events for some common cases in which events are out of order





### 2.1 Installation

```
$ pip install pbpstats
```

### 2.2 Setup data directory (optional but recommended)

To avoid repeating the same requests multiple times, and to allow for manually fixing issues with the raw play-by-play data I recommend setting up a data directory to save the response data. This will limit the number of requests you make to the NBA Stats API and allow you to manually fix issues with the raw data. Within this directory you will need to add four subdirectories(game\_details, overrides, pbp and schedule). To use the directory, just include it in your settings when initializing the client like in the example below. This directory is also where [override files](#) to fix periods starters and handle issues with parsing pbp that can't be fixed by editing pbp file are placed.

#### 2.2.1 Download data

If you want data with pbp event order fixed as well as overrides that are up to date as of this release you can download the files from [here](#). Unzip it and use the unzipped directory as your data directory.

#### 2.2.2 data.nba.com vs stats.nba.com

The data from both these sources is mostly the same but there are a few small differences. The data.nba.com play-by-play has the offense team id in all events, which makes it easier to track possession changes. The stats.nba.com events do not have this attribute and to get possession counts the play-by-play needs to be parsed, which makes it more sensitive to events being in the correct order. The stats.nba.com play-by-play also has lots of shots and rebounds that are out of order that need to be fixed manually. If you download the data from the link above, I have fixed these for previous seasons, but going forward if you use this you will have to keep up with fixing them manually yourself. If

you don't care if there may be an occasional possession count being off or you don't want to deal with manually fixing the event order I suggest using `data.nba.com` for your play-by-play provider.

The `stats.nba.com` has more older season data available. So it should be used if you want to work with older data.

Another difference is that the `data.nba.com` play-by-play updates in real time, so if you are looking to do live stats during a game you can use it for that.

## 2.3 Basic Usage

Within your settings that will be passed into the client when you instantiate it you can set the data directory and set the resources and data source you want.

Options for `source` are 'file' and 'web'. When source is 'file', it will pull data from the `dir` specified in the settings. When source is 'web' it will make an API request to get the data. If you want to save the response data to disk set `dir` within the settings and it will be saved in the appropriate subdirectory.

Options for `data_provider` are 'stats\_nba' and 'data\_nba' and 'live'.

See the code examples below for some examples settings.

Resource options are:

- `Boxscore` - basic boxscore stats
- `EnhancedPbp` - more detail than the basic play-by-play
- `Games` - for getting all games for a season or date
- `Pbp` - raw play-by-play
- `Possessions` - splits enhanced pbp data up into possessions.
- `Shots` (stats.nba.com only)

## 2.4 Code Examples

### 2.4.1 Game Data

The following code will instantiate the client and instantiate the Game data object for the given game id with boxscore and possession data.

```
from pbpstats.client import Client

settings = {
    "dir": "/response_data",
    "Boxscore": {"source": "file", "data_provider": "stats_nba"},
    "Possessions": {"source": "file", "data_provider": "stats_nba"},
}
client = Client(settings)
game = client.Game("0021900001")
```

Resource data can be accessed by calling `game.<snake_case_resource_name>.items`. In this case, since 'Boxscore' and 'Possessions' were provided in the settings dict boxscore and possessions data can be accessed via `game.boxscore.items` and `game.possessions.items`. See [Possessions](#) for properties for working with possessions data. See [Boxscore](#) for properties for working with boxscore data.

## 2.4.2 All Final Games For Season

The following code can be used to get all final games for a season.

```
from pbpstats.client import Client

settings = {
    "Games": {"source": "web", "data_provider": "data_nba"},
}
client = Client(settings)
season = client.Season("nba", "2019-20", "Regular Season")

for final_game in season.games.final_games:
    print(final_game)
```

## 2.4.3 All Final Games For Day

The following code can be used to get all final games for a season. Note that for day, data\_provider must be stats\_nba

```
from pbpstats.client import Client

settings = {
    "Games": {"source": "web", "data_provider": "stats_nba"},
}
client = Client(settings)
day = client.Day("12/05/2019", "nba")

for final_game in day.games.final_games:
    print(final_game)
```

## 2.4.4 Doing Detailed Stuff with Possession Data

The following code will get all possessions that start off a missed field goal.

```
off_rim_miss = [possession for possession in game.possessions.items if possession.
    ↳possession_start_type == "OffAtRimMiss"]
```

For more on what is available for possession data see *Possession* docs

## 2.4.5 Doing Detailed Stuff with Enhanced PBP Data

The following code will get the average 2pt shot distance on all missed field goals.

```
from pbpstats.resources.enhanced_pbp import FieldGoal

...

shot_dists = []
for possession in game.possessions.items:
    for possession_event in possession.events:
        if isinstance(possession_event, FieldGoal) and not possession_event.is_made_
    ↳and possession_event.shot_value == 2:
```

(continues on next page)

(continued from previous page)

```
shot_dists.append(possession_event.distance)
print(sum(shot_dists) / len(shot_dists))
```

For more on what is available for enhanced pbp data see `pbpstats.resources.enhanced_pbp` docs

## 2.5 Note on Ids

Player and team Ids the same player and team ids used by stats.nba.com. Lineup ids are '-' separated player ids (with player ids sorted as strings).

## 2.6 Issues with raw play-by-play

If you need to fix event order in the play-by-play file you will need to open the pbp file for the game in your data directory and change the order of the events list. Each event is a list and searching for the event number (the event number should be in the exception text somewhere) to find the event is a good place to start to figure out which event needs to be moved around.

## 3.1 Documentation

### 3.1.1 Data Loaders

The data loader modules are used to load data for a specific resource either from file stored on disk or via an API request.

#### 3.1.1.1 data.nba.com Data Loaders

data.nba.com data loaders are used to load data for a specific resource either from file stored on disk or via an API request to [data.nba.com](https://data.nba.com).

#### Boxscore

#### Boxscore

DataNbaBoxscoreLoader loads boxscore data for a game and creates *DataNbaBoxscoreItem* objects for each player and team

The following code will load boxscore data for game id “0021900001” from a file located in a subdirectory of the /data directory

```
from pbpstats.data_loader import DataNbaBoxscoreFileLoader, DataNbaBoxscoreLoader

source_loader = DataNbaBoxscoreFileLoader("/data")
boxscore_loader = DataNbaBoxscoreLoader("0021900001", source_loader)
print(boxscore_loader.items[0].data) # prints dict with a player's boxscore data for_
↪ game
```

```
class pbpstats.data_loader.data_nba.boxscore.loader.DataNbaBoxscoreLoader (game_id,  
                                                                    source_loader)
```

Bases: `pbpstats.data_loader.data_nba.base.DataNbaLoaderBase`

Loads data.nba.com source boxscore data for game. Team/Player data is stored in items attribute as `DataNbaBoxscoreItem` objects

**Parameters**

- **game\_id** (*str*) – NBA Stats Game Id
- **source\_loader** – `DataNbaBoxscoreFileLoader` or `DataNbaBoxscoreWebLoader` object

**data**

returns raw JSON response data

**data\_provider** = 'data\_nba'

**parent\_object** = 'Game'

**resource** = 'Boxscore'

```
class pbpstats.data_loader.data_nba.boxscore.file.DataNbaBoxscoreFileLoader (file_directory)
```

Bases: `pbpstats.data_loader.data_nba.file_loader.DataNbaFileLoader`

A `DataNbaBoxscoreFileLoader` object should be instantiated and passed into `DataNbaBoxscoreLoader` when loading data from file

**Parameters** **file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be `data_<game_id>.json` in the `/game_details` subdirectory.

**load\_data** (*game\_id*)

```
class pbpstats.data_loader.data_nba.boxscore.web.DataNbaBoxscoreWebLoader (file_directory=None)
```

Bases: `pbpstats.data_loader.data_nba.web_loader.DataNbaWebLoader`

A `DataNbaBoxscoreWebLoader` object should be instantiated and passed into `DataNbaBoxscoreLoader` when loading data directly from the NBA Stats API

**Parameters** **file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be `data_<game_id>.json` in the `/game_details` subdirectory. If not provided response data will not be saved on disk.

**load\_data** (*game\_id*)

## Enhanced PBP

## Enhanced PBP

`DataNbaEnhancedPbpLoader` loads pbp data for a game and creates `EnhancedPbpItem` objects for each event

Enhanced data for each event includes current players on floor, score, fouls to give and number of fouls committed by each player, plus additional data depending on event type

The following code will load pbp data for game id “0021900001” from a file located in a subdirectory of the `/data` directory

```

from pbpstats.data_loader import DataNbaEnhancedPbpFileLoader,   

↳DataNbaEnhancedPbpLoader

source_loader = DataNbaEnhancedPbpFileLoader("/data")
pbp_loader = DataNbaEnhancedPbpLoader("0021900001", source_loader)
print(pbp_loader.items[0].data) # prints dict with the first event of the game

```

**class** pbpstats.data\_loader.data\_nba.enhanced\_pbp.loader.**DataNbaEnhancedPbpLoader** (*game\_id*, *source\_loader*)

Bases: *pbpstats.data\_loader.data\_nba.pbp.loader.DataNbaPbpLoader*, *pbpstats.data\_loader.nba\_enhanced\_pbp\_loader.NbaEnhancedPbpLoader*

Loads data.nba.com source enhanced pbp data for game. Events are stored in items attribute as *EnhancedPbpItem* objects

#### Parameters

- **game\_id** (*str*) – NBA Stats Game Id
- **source\_loader** – *DataNbaEnhancedPbpFileLoader* or *DataNbaEnhancedPbpWebLoader* object

**Raises** *InvalidNumberOfStartersException*: If all 5 players that start the period for a team can't be determined. You can add the correct period starters to over-rides/missing\_period\_starters.json in your data directory to fix this.

**data\_provider** = 'data\_nba'

**parent\_object** = 'Game'

**resource** = 'EnhancedPbp'

**class** pbpstats.data\_loader.data\_nba.enhanced\_pbp.file.**DataNbaEnhancedPbpFileLoader** (*file\_directory*)

Bases: *pbpstats.data\_loader.data\_nba.pbp.file.DataNbaPbpFileLoader*

A *DataNbaEnhancedPbpFileLoader* object should be instantiated and passed into *DataNbaEnhancedPbpLoader* when loading data from file

**Parameters** **file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be *data\_<game\_id>.json* in the */pbp* subdirectory.

**class** pbpstats.data\_loader.data\_nba.enhanced\_pbp.web.**DataNbaEnhancedPbpWebLoader** (*file\_directory*)

Bases: *pbpstats.data\_loader.data\_nba.pbp.web.DataNbaPbpWebLoader*

A *DataNbaEnhancedPbpWebLoader* object should be instantiated and passed into *DataNbaEnhancedPbpLoader* when loading data directly from the NBA Stats API

**Parameters** **file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be *data\_<game\_id>.json* in the */pbp* subdirectory. If not provided response data will not be saved on disk.

## PBP

## PBP

*DataNbaPbpLoader* loads pbp data for a game and creates *DataNbaPbpItem* objects for each event

The following code will load pbp data for game id “0021900001” from a file located in a subdirectory of the */data* directory

```
from pbpstats.data_loader import DataNbaPbpFileLoader, DataNbaPbpLoader

source_loader = DataNbaPbpFileLoader("/data")
pbp_loader = DataNbaPbpLoader("0021900001", source_loader)
print(pbp_loader.items[0].data)  # prints dict with the first event of the game
```

[illegible]

Loads data.nba.com source pbp data for game. Events are stored in items attribute as *DataNbaPbpItem* objects

## Parameters

- **game\_id**(*str*) – NBA Stats Game Id
- **source\_loader** – *DataNbaPbpFileLoader* or *DataNbaPbpWebLoader* object

**data**  
returns raw JSON response data

```
data_provider = 'data_nba'
```

```
parent_object = 'Game'
```

```
resource = 'Pbp'
```

```
class pbpstats.data_loader.data_nba.pbp.file.DataNbaPbpFileLoader (file_directory=None)
    Bases: pbpstats.data_loader.data_nba.file_loader.DataNbaFileLoader
```

A `DataNbaPbpFileLoader` object should be instantiated and passed into `DataNbaPbpLoader` when loading data from file

**Parameters** `file_directory` (*str*) – Directory in which data should be loaded from. The specific file location will be `data_<game_id>.json` in the `/pbp` subdirectory.

```
load_data (game_id)
```

```
class pbpstats.data_loader.data_nba.pbp.web.DataNbaPbpWebLoader (file_directory=None)
    Bases: pbpstats.data_loader.data_nba.web_loader.DataNbaWebLoader
```

A `DataNbaPbpWebLoader` object should be instantiated and passed into `DataNbaPbpLoader` when loading data directly from the NBA Stats API

**Parameters** **file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be *data\_<game\_id>.json* in the */pbp* subdirectory. If not provided response data will not be saved on disk.

```
load_data(game_id)
```

## Possessions

## Possessions

`DataNbaPossessionLoader` loads possession data for a game and creates *Possession* objects for each possession

The following code will load possession data for game id “0021900001” from a pbp file located in the /pbp subdirectory of the /data directory



```

from pbpstats.data_loader import DataNbaPossessionFileLoader, DataNbaPossessionLoader

source_loader = DataNbaPossessionFileLoader("/data")
possession_loader = DataNbaPossessionLoader("0021900001", source_loader)
print(possession_loader.items[0].data)  # prints dict with the first possession of
↳ the game

```

**class** pbpstats.data\_loader.data\_nba.possessions.loader.**DataNbaPossessionLoader** (*game\_id*, *source\_loader*)

Bases: *pbpstats.data\_loader.nba\_possession\_loader.NbaPossessionLoader*

Loads data.nba.com source possession data for game. Possessions are stored in items attribute as *Possession* objects

#### Parameters

- **game\_id** (*str*) – NBA Stats Game Id
- **source\_loader** – *DataNbaPossessionFileLoader* or *DataNbaPossessionWebLoader* object

**data\_provider** = 'data\_nba'

**parent\_object** = 'Game'

**resource** = 'Possessions'

**class** pbpstats.data\_loader.data\_nba.possessions.file.**DataNbaPossessionFileLoader** (*file\_directory*)

Bases: object

A *DataNbaPossessionFileLoader* object should be instantiated and passed into *DataNbaPossessionLoader* when loading data from file

**Parameters** **file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be *data\_<game\_id>.json* in the */pbp* subdirectory.

**class** pbpstats.data\_loader.data\_nba.possessions.web.**DataNbaPossessionWebLoader** (*file\_directory*=None)

Bases: object

A *DataNbaPbpWebLoader* object should be instantiated and passed into *DataNbaPossessionLoader* when loading data directly from the NBA Stats API

**Parameters** **file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be *data\_<game\_id>.json* in the */pbp* subdirectory. If not provided response data will not be saved on disk.

## Schedule

### Schedule

*DataNbaScheduleLoader* loads schedule data for a season and creates *DataNbaGameItem* objects for each game

The following code will load schedule data for 2019-20 NBA Regular Season

```

from pbpstats.data_loader import DataNbaScheduleFileLoader, DataNbaScheduleLoader

source_loader = DataNbaScheduleFileLoader("/data")

```

(continues on next page)

(continued from previous page)

```
schedule_loader = DataNbaScheduleLoader("nba", "2019-20", "Regular Season", source_
↪loader)
print(schedule_loader.items[0].data)  # prints dict with the first game of the season
```

```
class pbpstats.data_loader.data_nba.schedule.loader.DataNbaScheduleLoader(league,
                                                                              sea-
                                                                              son,
                                                                              sea-
                                                                              son_type,
                                                                              source_loader)
```

Bases: `pbpstats.data_loader.data_nba.base.DataNbaLoaderBase`

Loads `data.nba.com` source schedule data for season. Games are stored in `items` attribute as `DataNbaGameItem` objects

#### Parameters

- **league** (*str*) – Options are ‘nba’, ‘wnba’ or ‘gleague’
- **season** (*str*) – Can be formatted as either 2019-20 or 2019.
- **season\_type** (*str*) – Options are ‘Regular Season’ or ‘Playoffs’ or ‘Play In’
- **source\_loader** – `DataNbaScheduleFileLoader` or `DataNbaScheduleWebLoader` object

#### data

returns raw JSON response data

**data\_provider** = 'data\_nba'

**parent\_object** = 'Season'

**resource** = 'Games'

```
class pbpstats.data_loader.data_nba.schedule.file.DataNbaScheduleFileLoader(file_directory=None)
```

Bases: `pbpstats.data_loader.data_nba.file_loader.DataNbaFileLoader`

A `DataNbaScheduleFileLoader` object should be instantiated and passed into `DataNbaScheduleLoader` when loading data from file

**Parameters file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be `data_<league>_<season_year>.json` in the `/schedule` subdirectory.

**load\_data** (*league, season*)

```
class pbpstats.data_loader.data_nba.schedule.web.DataNbaScheduleWebLoader(file_directory=None)
```

Bases: `pbpstats.data_loader.data_nba.web_loader.DataNbaWebLoader`

A `DataNbaScheduleWebLoader` object should be instantiated and passed into `DataNbaScheduleLoader` when loading data directly from the NBA Stats API

**Parameters file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be `data_<league>_<season_year>.json` in the `/schedule` subdirectory. If not provided response data will not be saved on disk.

#### league\_id

Returns League Id for league.

00 for nba, 10 for wnba, 20 for g-league

`load_data (league, season)`

## File Loader

**class** `pbpstats.data_loader.data_nba.file_loader.DataNbaFileLoader`

Bases: `pbpstats.data_loader.data_nba.base.DataNbaLoaderBase`

Base class for loading data.nba.com files saved on disk.

All data.nba.com data loader classes should inherit from this class.

This class should not be instantiated directly.

## Web Loader

**class** `pbpstats.data_loader.data_nba.web_loader.DataNbaWebLoader`

Bases: `pbpstats.data_loader.data_nba.base.DataNbaLoaderBase`

Base class for loading data from data.nba.com API request.

All data.nba.com data loader classes should inherit from this class.

This class should not be instantiated directly.

### 3.1.1.2 live Data Loaders

live data loaders are used to load data for a specific resource either from file stored on disk or via an API request to live endpoint at [nba.com](https://nba.com).

## Boxscore

## Boxscore

`LiveBoxscoreLoader` loads boxscore data for a game and creates `LiveBoxscoreItem` objects for each player and team

The following code will load boxscore data for game id “0021900001” from a file located in a subdirectory of the /data directory

```
from pbpstats.data_loader import LiveBoxscoreFileLoader, LiveBoxscoreLoader

source_loader = LiveBoxscoreFileLoader("/data")
boxscore_loader = LiveBoxscoreLoader("0021900001", source_loader)
print(boxscore_loader.items[0].data) # prints dict with a player's boxscore data for_
↪ game
```

**class** `pbpstats.data_loader.live.boxscore.loader.LiveBoxscoreLoader` (*game\_id*, *source\_loader*)

Bases: `pbpstats.data_loader.live.base.LiveLoaderBase`

Loads data.nba.com source boxscore data for game. Team/Player data is stored in items attribute as `LiveBoxscoreItem` objects

### Parameters

- **game\_id** (*str*) – NBA Stats Game Id

- **source\_loader** – *LiveBoxscoreFileLoader* or *LiveBoxscoreWebLoader* object

**data**  
returns raw JSON response data

**data\_provider** = 'live'

**parent\_object** = 'Game'

**resource** = 'Boxscore'

**class** `pbpstats.data_loader.live.boxscore.file.LiveBoxscoreFileLoader` (*file\_directory*)  
Bases: `pbpstats.data_loader.live.file_loader.LiveFileLoader`

A *LiveBoxscoreFileLoader* object should be instantiated and passed into *LiveBoxscoreLoader* when loading data from file

**Parameters** **file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be *live\_<game\_id>.json* in the */game\_details* subdirectory.

**load\_data** (*game\_id*)

**class** `pbpstats.data_loader.live.boxscore.web.LiveBoxscoreWebLoader` (*file\_directory=None*)  
Bases: `pbpstats.data_loader.live.web_loader.LiveWebLoader`

*LiveBoxscoreWebLoader* object should be instantiated and passed into *LiveBoxscoreLoader* when loading data directly from the NBA Stats API

**Parameters** **file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be *live\_<game\_id>.json* in the */game\_details* subdirectory. If not provided response data will not be saved on disk.

**load\_data** (*game\_id*)

## Enhanced PBP

### Enhanced PBP

*LiveEnhancedPbpLoader* loads pbp data for a game and creates *EnhancedPbpItem* objects for each event

Enhanced data for each event includes current players on floor, score, fouls to give and number of fouls committed by each player, plus additional data depending on event type

The following code will load pbp data for game id “0021900001” from a file located in a subdirectory of the /data directory

```
from pbpstats.data_loader import LiveEnhancedPbpFileLoader, LiveEnhancedPbpLoader

source_loader = LiveEnhancedPbpFileLoader("/data")
pbp_loader = LiveEnhancedPbpLoader("0021900001", source_loader)
print(pbp_loader.items[0].data)  # prints dict with the first event of the game
```

**class** `pbpstats.data_loader.live.enhanced_pbp.loader.LiveEnhancedPbpLoader` (*game\_id*, *source\_loader*)  
Bases: `pbpstats.data_loader.live.pbp.loader.LivePbpLoader`, `pbpstats.data_loader.nba_enhanced_pbp_loader.NbaEnhancedPbpLoader`

Loads data.nba.com source enhanced pbp data for game. Events are stored in items attribute as *EnhancedPbpItem* objects

**Parameters**

- **game\_id**(*str*) – NBA Stats Game Id
- **source\_loader** – *LiveEnhancedPbpFileLoader* or *LiveEnhancedPbpWebLoader* object

**data\_provider** = 'live'

**parent\_object** = 'Game'

**resource** = 'EnhancedPbp'

**class** pbpstats.data\_loader.live.enhanced\_pbp.file.**LiveEnhancedPbpFileLoader**(*file\_directory*)  
 Bases: *pbpstats.data\_loader.live.pbp.file.LivePbpFileLoader*

A *LiveEnhancedPbpFileLoader* object should be instantiated and passed into *LiveEnhancedPbpLoader* when loading data from file

**Parameters**

- **game\_id**(*str*) – NBA Stats Game Id
- **file\_directory**(*str*) – Directory in which data should be loaded from. The specific file location will be *live\_<game\_id>.json* in the */pbp* subdirectory.

**class** pbpstats.data\_loader.live.enhanced\_pbp.web.**LiveEnhancedPbpWebLoader**(*file\_directory=None*)  
 Bases: *pbpstats.data\_loader.live.pbp.web.LivePbpWebLoader*

A *LiveEnhancedPbpWebLoader* object should be instantiated and passed into *LiveEnhancedPbpLoader* when loading data directly from the NBA Stats API

**Parameters** **file\_directory**(*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be *live\_<game\_id>.json* in the */pbp* subdirectory. If not provided response data will not be saved on disk.

**PBP****PBP**

*LivePbpLoader* loads pbp data for a game and creates *LivePbpItem* objects for each event

The following code will load pbp data for game id “0021900001” from a file located in a subdirectory of the /data directory

```
from pbpstats.data_loader import LivePbpFileLoader, LivePbpLoader

source_loader = LivePbpFileLoader("/data")
pbp_loader = LivePbpLoader("0021900001", source_loader)
print(pbp_loader.items[0].data)  # prints dict with the first event of the game
```

**class** pbpstats.data\_loader.live.pbp.loader.**LivePbpLoader**(*game\_id*, *source\_loader*)  
 Bases: *pbpstats.data\_loader.live.base.LiveLoaderBase*

Loads live data source pbp data for game. Events are stored in items attribute as *LivePbpItem* objects

**Parameters**

- **game\_id**(*str*) – NBA Stats Game Id
- **source\_loader** – *LivePbpFileLoader* or *LivePbpWebLoader* object

**data**  
returns raw JSON response data

**data\_provider** = 'live'

**parent\_object** = 'Game'

**resource** = 'Pbp'

**class** `pbpstats.data_loader.live.pbp.file.LivePbpFileLoader` (*file\_directory*)

Bases: `pbpstats.data_loader.live.file_loader.LiveFileLoader`

A `LivePbpFileLoader` object should be instantiated and passed into `LivePbpLoader` when loading data from file

#### Parameters

- **game\_id** (*str*) – NBA Stats Game Id
- **file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be `live_<game_id>.json` in the `/pbp` subdirectory.

**load\_data** (*game\_id*)

**class** `pbpstats.data_loader.live.pbp.web.LivePbpWebLoader` (*file\_directory=None*)

Bases: `pbpstats.data_loader.live.web_loader.LiveWebLoader`

A `LivePbpWebLoader` object should be instantiated and passed into `LivePbpLoader` when loading data directly from the NBA Stats API

**Parameters** **file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be `live_<game_id>.json` in the `/pbp` subdirectory. If not provided response data will not be saved on disk.

**load\_data** (*game\_id*)

## Possessions

### Possessions

`LivePossessionLoader` loads possession data for a game and creates `Possession` objects for each possession

The following code will load possession data for game id “0021900001” from a pbp file located in the `/pbp` subdirectory of the `/data` directory

```
from pbpstats.data_loader import LivePossessionFileLoader, LivePossessionLoader

source_loader = LivePossessionFileLoader("/data")
pbp_loader = LivePossessionLoader("0021900001", source_loader)
print(pbp_loader.items[0].data)  # prints dict with the first event of the game
```

**class** `pbpstats.data_loader.live.possessions.loader.LivePossessionLoader` (*game\_id*, *source\_loader*)

Bases: `pbpstats.data_loader.nba_possession_loader.NbaPossessionLoader`

Loads live data source possession data for game. Possessions are stored in items attribute as `Possession` objects

#### Parameters

- **game\_id** (*str*) – NBA Stats Game Id

- **source\_loader** – *LivePossessionFileLoader* or *LivePossessionWebLoader* object

**data\_provider** = 'live'

**parent\_object** = 'Game'

**resource** = 'Possessions'

**class** pbpstats.data\_loader.live.possessions.file.**LivePossessionFileLoader** (*file\_directory*)  
Bases: object

A *LivePossessionFileLoader* object should be instantiated and passed into *LivePossessionLoader* when loading data from file

**Parameters**

- **game\_id** (*str*) – NBA Stats Game Id
- **file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be *live\_<game\_id>.json* in the */pbp* subdirectory.

**class** pbpstats.data\_loader.live.possessions.web.**LivePossessionWebLoader** (*file\_directory=None*)  
Bases: object

A *LivePossessionWebLoader* object should be instantiated and passed into *LivePossessionLoader* when loading data directly from the NBA Stats API

**Parameters** **file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be *live\_<game\_id>.json* in the */pbp* subdirectory. If not provided response data will not be saved on disk.

## File Loader

**class** pbpstats.data\_loader.live.file\_loader.**LiveFileLoader**  
Bases: pbpstats.data\_loader.live.base.LiveLoaderBase

Base class for loading live data files saved on disk.

All live data data loader classes should inherit from this class.

This class should not be instantiated directly.

## Web Loader

**class** pbpstats.data\_loader.live.web\_loader.**LiveWebLoader**  
Bases: pbpstats.data\_loader.live.base.LiveLoaderBase

Base class for loading data from live data API request.

All live data data loader classes should inherit from this class.

This class should not be instantiated directly.

### 3.1.1.3 stats.nba.com Data Loaders

stats.nba.com data loaders are used to load data for a specific resource either from file stored on disk or via an API request to [stats.nba.com](https://stats.nba.com).

## Base

**class** `pbpstats.data_loader.stats_nba.base.StatsNbaLoaderBase`

Bases: `object`

Base Class for all stats.nba.com data loaders

This class should not be instantiated directly

**data**

returns data from response JSON as a list of dicts

**static dedupe\_events\_row\_set** (*events\_row\_set*)

Dedupes list of results while preserving order

Used to dedupe events rowSets pbp response because some games have duplicate events

**Parameters** *events\_row\_set* (*list*) – List of results from API Response

**Returns** deduped list of results

**league**

Returns League for game id.

First 2 in game id represent league - 00 for nba, 10 for wnba, 20 for g-league

**league\_id**

Returns League Id for league.

00 for nba, 10 for wnba, 20 for g-league

**make\_list\_of\_dicts** (*results\_set\_index=0*)

Creates list of dicts from source data

**Parameters** *results\_set\_index* (*int*) – Index results are in. Default is 0

**Returns** list of dicts with data for results

**season**

Returns season for game id

4th and 5th characters in game id represent season year ex. for 2016-17 season 4th and 5th characters would be 16 and season should return 2016-17 For WNBA just returns season year

**season\_type**

Returns season type for game id

3rd character in game id represent season type - 2 for reg season, 4 for playoffs, 5 for play in

## Boxscore

### Boxscore

`StatsNbaBoxscoreLoader` loads boxscore data for a game and creates `StatsNbaBoxscoreItem` objects for each player and team

The following code will load boxscore data for game id “0021900001” from a file located in a subdirectory of the /data directory



```

from pbpstats.data_loader import StatsNbaBoxscoreFileLoader, StatsNbaBoxscoreLoader

source_loader = StatsNbaBoxscoreFileLoader("/data")
boxscore_loader = StatsNbaBoxscoreLoader("0021900001", source_loader)
print(boxscore_loader.items[0].data) # prints dict with a player's boxscore data for
↪ game

```

**class** pbpstats.data\_loader.stats\_nba.boxscore.loader.**StatsNbaBoxscoreLoader** (*game\_id*, *source\_loader*)  
 Bases: *pbpstats.data\_loader.stats\_nba.base.StatsNbaLoaderBase*

Loads stats.nba.com source boxscore data for game. Team/Player data is stored in items attribute as *StatsNbaBoxscoreItem* objects

#### Parameters

- **game\_id** (*str*) – NBA Stats Game Id
- **source\_loader** – *StatsNbaBoxscoreFileLoader* or *StatsNbaBoxscoreWebLoader* object

**data\_provider** = 'stats\_nba'

**parent\_object** = 'Game'

**resource** = 'Boxscore'

**class** pbpstats.data\_loader.stats\_nba.boxscore.file.**StatsNbaBoxscoreFileLoader** (*file\_directory*)  
 Bases: *pbpstats.data\_loader.stats\_nba.file\_loader.StatsNbaFileLoader*

A *StatsNbaBoxscoreFileLoader* object should be instantiated and passed into *StatsNbaBoxscoreLoader* when loading data from file

**Parameters** **file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be *stats\_boxscore\_<game\_id>.json* in the */game\_details* subdirectory.

**load\_data** (*game\_id*)

**class** pbpstats.data\_loader.stats\_nba.boxscore.web.**StatsNbaBoxscoreWebLoader** (*file\_directory=None*)  
 Bases: *pbpstats.data\_loader.stats\_nba.web\_loader.StatsNbaWebLoader*

A *StatsNbaBoxscoreWebLoader* object should be instantiated and passed into *StatsNbaBoxscoreLoader* when loading data directly from the NBA Stats API

**Parameters** **file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be *stats\_boxscore\_<game\_id>.json* in the */game\_details* subdirectory. If not provided response data will not be saved on disk.

**load\_data** (*game\_id*)

## Enhanced PBP

## Enhanced PBP

*StatsNbaEnhancedPbpLoader* loads pbp data for a game and creates *EnhancedPbpItem* objects for each event

Enhanced data for each event includes current players on floor, score, fouls to give and number of fouls committed by each player, plus additional data depending on event type

The following code will load pbp data for game id “0021900001” from a file located in a subdirectory of the /data directory

```
from pbpstats.data_loader import StatsNbaEnhancedPbpFileLoader, \
↳ StatsNbaEnhancedPbpLoader

source_loader = StatsNbaEnhancedPbpFileLoader("/data")
pbp_loader = StatsNbaEnhancedPbpLoader("0021900001", source_loader)
print(pbp_loader.items[0].data) # prints dict with the first event of the game
```

**class** pbpstats.data\_loader.stats\_nba.enhanced\_pbp.loader.**StatsNbaEnhancedPbpLoader** (*game\_id*, *source\_loader*)

Bases: *pbpstats.data\_loader.stats\_nba.pbp.loader.StatsNbaPbpLoader*, *pbpstats.data\_loader.nba\_enhanced\_pbp\_loader.NbaEnhancedPbpLoader*

Loads stats.nba.com source enhanced pbp data for game. Events are stored in items attribute as *EnhancedPbpItem* objects

#### Parameters

- **game\_id** (*str*) – NBA Stats Game Id
- **source\_loader** – *StatsNbaEnhancedPbpFileLoader* or *StatsNbaEnhancedPbpWebLoader* object

**Raises** *InvalidNumberOfStartersException*: If all 5 players that start the period for a team can’t be determined. You can add the correct period starters to overrides/missing\_period\_starters.json in your data directory to fix this.

**Raises** *EventOrderError*: If rebound event is not immediately following a missed shot event. You can manually edit the event order in the pbp file stored on disk to fix this.

**data\_provider** = 'stats\_nba'

**parent\_object** = 'Game'

**resource** = 'EnhancedPbp'

**class** pbpstats.data\_loader.stats\_nba.enhanced\_pbp.file.**StatsNbaEnhancedPbpFileLoader** (*file\_directory*)

Bases: *pbpstats.data\_loader.stats\_nba.pbp.file.StatsNbaPbpFileLoader*

A *StatsNbaEnhancedPbpFileLoader* object should be instantiated and passed into *StatsNbaEnhancedPbpLoader* when loading data from file

**Parameters** **file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be *stats\_<game\_id>.json* in the */pbp* subdirectory.

**class** pbpstats.data\_loader.stats\_nba.enhanced\_pbp.web.**StatsNbaEnhancedPbpWebLoader** (*file\_directory*)

Bases: *pbpstats.data\_loader.stats\_nba.pbp.web.StatsNbaPbpWebLoader*

A *StatsNbaEnhancedPbpWebLoader* object should be instantiated and passed into *StatsNbaEnhancedPbpLoader* when loading data directly from the NBA Stats API

**Parameters** **file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be *stats\_<game\_id>.json* in the */pbp* subdirectory. If not provided response data will not be saved on disk.

## PBP

## PBP

StatsNbaPbpLoader loads pbp data for a game and creates *StatsNbaPbpItem* objects for each event

The following code will load pbp data for game id “0021900001” from a file located in a subdirectory of the /data directory

```
from pbpstats.data_loader import StatsNbaPbpFileLoader, StatsNbaPbpLoader

source_loader = StatsNbaPbpFileLoader("/data")
pbp_loader = StatsNbaPbpLoader("0021900001", source_loader)
print(pbp_loader.items[0].data)  # prints dict with the first event of the game
```

```
class pbpstats.data_loader.stats_nba.pbp.loader.StatsNbaPbpLoader (game_id,
                                                                    source_loader)
```

Bases: *pbpstats.data\_loader.stats\_nba.base.StatsNbaLoaderBase*

Loads stats.nba.com source pbp data for game. Events are stored in items attribute as *StatsNbaPbpItem* objects

### Parameters

- **game\_id** (*str*) – NBA Stats Game Id
- **source\_loader** – *StatsNbaPbpFileLoader* or *StatsNbaPbpWebLoader* object

**data\_provider** = 'stats\_nba'

**parent\_object** = 'Game'

**resource** = 'Pbp'

```
class pbpstats.data_loader.stats_nba.pbp.file.StatsNbaPbpFileLoader (file_directory)
```

Bases: *pbpstats.data\_loader.stats\_nba.file\_loader.StatsNbaFileLoader*

A *StatsNbaPbpFileLoader* object should be instantiated and passed into *StatsNbaPbpLoader* when loading data from file

**Parameters file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be *stats\_<game\_id>.json* in the */pbp* subdirectory.

**load\_data** (*game\_id*)

```
class pbpstats.data_loader.stats_nba.pbp.web.StatsNbaPbpWebLoader (file_directory=None)
```

Bases: *pbpstats.data\_loader.stats\_nba.web\_loader.StatsNbaWebLoader*

A *StatsNbaPbpWebLoader* object should be instantiated and passed into *StatsNbaPbpLoader* when loading data directly from the NBA Stats API

**Parameters file\_directory** (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be *stats\_<game\_id>.json* in the */pbp* subdirectory. If not provided response data will not be saved on disk.

**load\_data** (*game\_id*)

## Possessions

## Possessions

StatsNbaPossessionLoader loads possession data for a game and creates *Possession* objects for each possession

The following code will load possession data for game id “0021900001” from a pbp file located in the /pbp subdirectory of the /data directory

```
from pbpstats.data_loader import StatsNbaPossessionFileLoader, StatsNbaPossessionLoader

source_loader = StatsNbaPossessionFileLoader("/data")
pbp_loader = StatsNbaPossessionLoader("0021900001", source_loader)
print(pbp_loader.items[0].data)  # prints dict with the first event of the game
```

**class** pbpstats.data\_loader.stats\_nba.possessions.loader.**StatsNbaPossessionLoader** (*game\_id*, *source\_loader*)

Bases: *pbpstats.data\_loader.nba\_possession\_loader.NbaPossessionLoader*

Loads stats.nba.com source possession data for game. Possessions are stored in items attribute as *Possession* objects

### Parameters

- **game\_id** (*str*) – NBA Stats Game Id
- **source\_loader** – *StatsNbaPossessionFileLoader* or *StatsNbaPossessionWebLoader* object

**Raises** TeamHasBackToBackPossessionsException: If team has the ball on back-to-back possessions.

**data\_provider** = 'stats\_nba'

### league

Returns League for game id.

First 2 in game id represent league - 00 for nba, 10 for wnba, 20 for g-league

**parent\_object** = 'Game'

**resource** = 'Possessions'

**exception** pbpstats.data\_loader.stats\_nba.possessions.loader.**TeamHasBackToBackPossessionsException**

Bases: Exception

Class for exception when a team is credited with back-to-back possessions.

You can manually edit the event order in the pbp file stored on disk or add an event to the overrides file in your data directory to fix this.

**class** pbpstats.data\_loader.stats\_nba.possessions.file.**StatsNbaPossessionFileLoader** (*file\_directory*)

Bases: object

A *StatsNbaPossessionFileLoader* object should be instantiated and passed into *StatsNbaPossessionLoader* when loading data from file

**Parameters** **file\_directory** (*str*) – Directory in which data should be loaded from. The specific file location will be *stats\_<game\_id>.json* in the /pbp subdirectory.

**class** pbpstats.data\_loader.stats\_nba.possessions.web.**StatsNbaPossessionWebLoader** (*file\_directory*)

Bases: object

A `StatsNbaPossessionWebLoader` object should be instantiated and passed into `StatsNbaPossessionLoader` when loading data directly from the NBA Stats API

**Parameters** `file_directory` (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be `stats_<game_id>.json` in the `/pbp` subdirectory. If not provided response data will not be saved on disk.

## Game Finder

### Boxscore

`StatsNbaLeagueGameLogLoader` loads all games for a season and creates `StatsNbaGameItem` objects for each game

The following code will load data for the 2019-20 NBA Regular Season

```
from pbpstats.data_loader import StatsNbaLeagueGameLogWebLoader, \
    StatsNbaLeagueGameLogLoader

source_loader = StatsNbaLeagueGameLogWebLoader("/data")
league_game_log_loader = StatsNbaLeagueGameLogLoader("nba", "2019-20", "Regular Season",
    source_loader)
print(league_game_log_loader.items[0].data) # prints dict for first game
```

```
class pbpstats.data_loader.stats_nba.league_game_log.loader.StatsNbaLeagueGameLogLoader (league_game_log_loader)
    sea
    son
    sea
    son
    sou
```

Bases: `pbpstats.data_loader.stats_nba.base.StatsNbaLoaderBase`

Loads stats.nba.com source data for season. Games are stored in items attribute as `StatsNbaGameItem` objects

#### Parameters

- **league** (*str*) – Options are 'nba', 'wnba' or 'gleague'
- **season** (*str*) – Formatted as 2019-20 for NBA and G-League, 2019 of WNBA.
- **season\_type** (*str*) – Options are 'Regular Season' or 'Playoffs' or 'Play In'
- **source\_loader** – `StatsNbaLeagueGameLogFileLoader` or `StatsNbaLeagueGameLogWebLoader` object

**data\_provider** = 'stats\_nba'

**parent\_object** = 'Season'

**resource** = 'Games'

```
class pbpstats.data_loader.stats_nba.league_game_log.file.StatsNbaLeagueGameLogFileLoader (league_game_log_loader)
    Bases: pbpstats.data_loader.stats_nba.file_loader.StatsNbaFileLoader
```

A `StatsNbaLeagueGameLogFileLoader` object should be instantiated and passed into `StatsNbaLeagueGameLogLoader` when loading data from file

**Parameters** `file_directory` (*str*) – Directory in which data should be loaded from. The specific file location will be `stats_leaguegame_log_<league>_<season>_<season_type>.json` in the `/schedule` subdirectory.

`load_data` (*league, season, season\_type*)

**class** `pbpstats.data_loader.stats_nba.league_game_log.web.StatsNbaLeagueGameLogWebLoader` (*file\_directory*)

Bases: `pbpstats.data_loader.stats_nba.web_loader.StatsNbaWebLoader`

A `StatsNbaLeagueGameLogWebLoader` object should be instantiated and passed into `StatsNbaLeagueGameLogLoader` when loading data directly from the NBA Stats API

**Parameters** `file_directory` (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be `stats_<league>_<season>_<season_type>.json` in the `/schedule` subdirectory. If not provided response data will not be saved on disk.

`load_data` (*league, season, season\_type*)

## Scoreboard

### Boxscore

`StatsNbaScoreboardLoader` loads all games for a date and creates `StatsNbaGameItem` objects for each game

The following code will load data for 02/03/2020

```
from pbpstats.data_loader import StatsNbaScoreboardFileLoader, \
    StatsNbaScoreboardLoader

source_loader = StatsNbaScoreboardFileLoader("/data")
scoreboard_loader = StatsNbaScoreboardLoader("02/03/2020", "nba", source_loader)
print(scoreboard_loader.items[0].data) # prints dict for first game
```

**class** `pbpstats.data_loader.stats_nba.scoreboard.loader.StatsNbaScoreboardLoader` (*date, league\_string, source\_loader*)

Bases: `pbpstats.data_loader.stats_nba.file_loader.StatsNbaFileLoader, pbpstats.data_loader.stats_nba.web_loader.StatsNbaWebLoader`

Loads stats.nba.com source data for date. Games are stored in items attribute as `StatsNbaGameItem` objects

#### Parameters

- **date** (*str*) – Formatted as MM/DD/YYYY
- **league\_string** (*str*) – Options are 'nba', 'wnba' or 'gleague'
- **source\_loader** – `StatsNbaScoreboardFileLoader` or `StatsNbaScoreboardWebLoader` object

`data_provider` = 'stats\_nba'

`parent_object` = 'Day'

`resource` = 'Games'

**class** `pbpstats.data_loader.stats_nba.scoreboard.file.StatsNbaScoreboardFileLoader` (*file\_directory*)

Bases: `pbpstats.data_loader.stats_nba.file_loader.StatsNbaFileLoader`

A `StatsNbaScoreboardFileLoader` object should be instantiated and passed into `StatsNbaScoreboardLoader` when loading data from file

**Parameters** `file_directory` (*str*) – Directory in which data should be loaded from. The specific file location will be `stats_<league>_<date>.json` in the `/schedule` subdirectory.

`load_data` (*date*, *league\_string*)

**class** `pbpstats.data_loader.stats_nba.scoreboard.web.StatsNbaScoreboardWebLoader` (*file\_directory*=  
Bases: `pbpstats.data_loader.stats_nba.web_loader.StatsNbaWebLoader`

A `StatsNbaScoreboardWebLoader` object should be instantiated and passed into `StatsNbaScoreboardLoader` when loading data directly from the NBA Stats API

**Parameters** `file_directory` (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be `stats_<league>_<date>.json` in the `/schedule` subdirectory. If not provided response data will not be saved on disk.

`league_id`

Returns League Id for league.

00 for nba, 10 for wnba, 20 for g-league

`load_data` (*date*, *league\_string*)

## Shots

### Boxscore

`StatsNbaShotsLoader` loads shot data for a game and creates `StatsNbaShot` objects for all shots

The following code will load shot data for game id “0021900001” from a file located in a subdirectory of the `/data` directory

```
from pbpstats.data_loader import StatsNbaShotsFileLoader, StatsNbaShotsLoader

source_loader = StatsNbaShotsFileLoader("/data")
shot_loader = StatsNbaShotsLoader("0021900001", source_loader)
print(shot_loader.items[0].data) # prints dict with data for one shot from game
```

**class** `pbpstats.data_loader.stats_nba.shots.loader.StatsNbaShotsLoader` (*game\_id*,  
*source\_loader*)

Bases: `pbpstats.data_loader.stats_nba.base.StatsNbaLoaderBase`

Loads stats.nba.com source shot data for game. Shots are stored in items attribute as `StatsNbaShot` objects

#### Parameters

- **game\_id** (*str*) – NBA Stats Game Id
- **source\_loader** – `StatsNbaShotsFileLoader` or `StatsNbaShotsWebLoader` object

`data_provider` = 'stats\_nba'

**make\_list\_of\_dicts** (*results\_set\_index*=0)

Creates list of dicts from home and away source data

**Parameters** `results_set_index` (*int*) – Index results are in. Default is 0

**Returns** list of dicts with shot data for all shots

```
parent_object = 'Game'
```

```
resource = 'Shots'
```

```
class pbpstats.data_loader.stats_nba.shots.file.StatsNbaShotsFileLoader (file_directory)
    Bases: pbpstats.data_loader.stats_nba.file_loader.StatsNbaFileLoader
```

A StatsNbaShotsFileLoader object should be instantiated and passed into StatsNbaShotsLoader when loading data from file

**Parameters** `file_directory` (*str*) – Directory in which data should be loaded from. The specific file location will be `stats_home_shots_<game_id>.json` and `stats_away_shots_<game_id>.json` in the `/game_details` subdirectory.

```
load_data (game_id)
```

```
class pbpstats.data_loader.stats_nba.shots.web.StatsNbaShotsWebLoader (file_directory=None)
    Bases: pbpstats.data_loader.stats_nba.web_loader.StatsNbaWebLoader
```

A StatsNbaShotsWebLoader object should be instantiated and passed into StatsNbaShotsLoader when loading data directly from the NBA Stats API

**Parameters** `file_directory` (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be `stats_home_shots_<game_id>.json` and `stats_away_shots_<game_id>.json` in the `/game_details` subdirectory. If not provided response data will not be saved on disk.

```
load_data (game_id)
```

## Game Summary

### Boxscore

StatsNbaSummaryLoader loads summary data for a game and creates *StatsNbaGameItem* objects for game. The following code will load summary data for game id “0021900001” from a file located in a subdirectory of the `/data` directory

```
from pbpstats.data_loader.stats_nba.summary.file import StatsNbaSummaryFileLoader
from pbpstats.data_loader.stats_nba.summary.loader import StatsNbaSummaryLoader

source_loader = StatsNbaSummaryFileLoader("/data")
summary_loader = StatsNbaSummaryLoader("0021900001", source_loader)
print(summary_loader.items[0].data) # prints game summary dict for game
```

```
class pbpstats.data_loader.stats_nba.summary.loader.StatsNbaSummaryLoader (game_id,
                                                                              source_loader)
    Bases: pbpstats.data_loader.stats_nba.base.StatsNbaLoaderBase
```

Loads stats.nba.com source summary data for game. Summary data is stored in items attribute as *StatsNbaGameItem* objects

#### Parameters

- `game_id` (*str*) – NBA Stats Game Id
- `source_loader` – *StatsNbaSummaryFileLoader* or *StatsNbaSummaryWebLoader* object

```
data_provider = 'stats_nba'
```

```
parent_object = 'Game'
```



```
resource = 'Games'
```

```
class pbpstats.data_loader.stats_nba.summary.file.StatsNbaSummaryFileLoader (file_directory)
    Bases: pbpstats.data_loader.stats_nba.file_loader.StatsNbaFileLoader
```

A StatsNbaSummaryFileLoader object should be instantiated and passed into StatsNbaSummaryLoader when loading data from file

**Parameters** `file_directory` (*str*) – Directory in which data should be loaded from. The specific file location will be `stats_summary_<game_id>.json` in the `/game_details` subdirectory.

```
load_data (game_id)
```

```
class pbpstats.data_loader.stats_nba.summary.web.StatsNbaSummaryWebLoader (file_directory=None)
    Bases: pbpstats.data_loader.stats_nba.web_loader.StatsNbaWebLoader
```

A StatsNbaSummaryWebLoader object should be instantiated and passed into StatsNbaSummaryLoader when loading data directly from the NBA Stats API

**Parameters** `file_directory` (*str*) – (optional, use it if you want to store the response data on disk) Directory in which data should be either stored. The specific file location will be `stats_summary_<game_id>.json` in the `/game_details` subdirectory. If not provided response data will not be saved on disk.

```
load_data (game_id)
```

## File Loader

```
class pbpstats.data_loader.stats_nba.file_loader.StatsNbaFileLoader
    Bases: pbpstats.data_loader.stats_nba.base.StatsNbaLoaderBase
```

Base class for loading stats.nba.com files saved on disk.

All stats.nba.com data loader classes should inherit from this class.

This class should not be instantiated directly.

## Web Loader

```
class pbpstats.data_loader.stats_nba.web_loader.StatsNbaWebLoader
    Bases: pbpstats.data_loader.stats_nba.base.StatsNbaLoaderBase
```

Base class for loading data from data.nba.com API request.

All stats.nba.com data loader classes should inherit from this class.

This class should not be instantiated directly.

### 3.1.1.4 Data Loader Factory

DataLoaderFactory can be used to create data loader objects from the `data_loader` module.

The following code will create a data loader object for loading enhanced pbp from stats.nba.com.

```
from pbpstats.data_loader.factory import DataLoaderFactory

data_loader = DataLoaderFactory()
stats_enhanced_pbp_data_loader = data_loader.get_data_loader("stats_nba", "EnhancedPbp
↪")
```

(continues on next page)

(continued from previous page)

```
print(stats_enhanced_pbp_data_loader[0])
# prints "<class 'pbpstats.data_loader.stats_nba.enhanced_pbp_loader.
↳ StatsNbaEnhancedPbpLoader'>"
```

**class** pbpstats.data\_loader.factory.DataLoaderFactory

Bases: object

Class for factory of data loader classes. On initialization will load in all data loader classes in data\_loader module

**get\_data\_loader** (data\_provider, resource)

Gets data loader classes for given data provider and resource.

**Parameters**

- **data\_provider** (*str*) – Which data provider should data be loaded from. Options are 'stats\_nba' or 'data\_nba' or 'live'
- **resource** (*str*) – Name of class from resources directory

**Returns** list of data loader classes**Return type** list

### 3.1.1.5 Enhanced PBP Loader

**class** pbpstats.data\_loader.nba\_enhanced\_pbp\_loader.NbaEnhancedPbpLoader

Bases: object

Class for shared methods between DataNbaEnhancedPbpLoader and StatsNbaEnhancedPbpLoader

Both DataNbaEnhancedPbpLoader and StatsNbaEnhancedPbpLoader should inherit from this class

This class should not be instantiated directly

### 3.1.1.6 Possession Loader

**class** pbpstats.data\_loader.nba\_possession\_loader.NbaPossessionLoader

Bases: object

Class for shared methods between DataNbaPossessionLoader and StatsNbaPossessionLoader

Both DataNbaPossessionLoader and StatsNbaPossessionLoader should inherit from this class

This class should not be instantiated directly

## 3.1.2 Objects

### 3.1.2.1 Day

Instantiating a Day object will load all resources for the Day object that were set in the settings when the client was instantiated

The following code will instantiate the client and get game data for games played on 02/03/2020

```

from pbpstats.client import Client

settings = {
    "dir": "/response_data",
    "Games": {"source": "web", "data_provider": "stats_nba"}
}
client = Client(settings)
day = client.Day("02/03/2020", "nba")
for game in day.games.items:
    print(game.data)

```

**class** pbpstats.objects.day.**Day** (*date*, *league*)

Bases: object

Class for loading resource data from data loaders with a parent\_object of Day

**Parameters**

- **date** (*str*) – Formatted as MM/DD/YYYY
- **league** (*str*) – Options are ‘nba’, ‘wnba’ or ‘gleague’

### 3.1.2.2 Game

Instantiating a Game object will load all resources for the Game object that were set in the settings when the client was instantiated

The following code will instantiate the client and get possession data for game id 0021900001 from files in / response\_data subdirectories

```

from pbpstats.client import Client

settings = {
    "dir": "/response_data",
    "Possessions": {"source": "file", "data_provider": "stats_nba"}
}
client = Client(settings)
game = client.Game('0021900001')
for possession in game.possessions.items:
    print(possession)

```

**class** pbpstats.objects.game.**Game** (*game\_id*)

Bases: object

Class for loading resource data from data loaders with a parent\_object of Game

**Parameters** **game\_id** (*str*) – NBA Stats Game Id

### 3.1.2.3 Season

Instantiating a Season object will load all resources for the Season object that were set in the settings when the client was instantiated

The following code will instantiate the client and get all games for the 2019-20 NBA Regular Season and store the schedule response in a / response\_data subdirectory

```
from pbpstats.client import Client

settings = {
    "dir": "/response_data",
    "Games": {"source": "web", "data_provider": "data_nba"}
}
client = Client(settings)
season = client.Season("nba", "2019-20", "Regular Season")
for game in season.games.items:
    print(game)
```

**class** pbpstats.objects.season.**Season** (*league, season, season\_type*)  
Bases: object

Class for loading resource data from data loaders with a parent\_object of Season

**Parameters**

- **league** (*str*) – Options are ‘nba’, ‘wnba’ or ‘gleague’
- **season** (*str*) – Can be formatted as either 2019-20 or 2019.
- **season\_type** (*str*) – Options are ‘Regular Season’ or ‘Playoffs’ or ‘Play In’

### 3.1.3 Resources

#### 3.1.3.1 Boxscore

##### Boxscore

The Boxscore class has some basic properties for handling boxscore data

**class** pbpstats.resources.boxscore.boxscore.**Boxscore** (*items*)  
Bases: pbpstats.resources.base.Base

Class for boxscore items

**Parameters** **items** (*list*) – list of either *StatsNbaBoxscoreItem* or *LiveBoxscoreItem* or *DataNbaBoxscoreItem* items, typically from a boxscore data loader

**data**  
returns dict with boxscore items split up by player and team

**player\_items**  
returns list of player boxscore items

**player\_name\_map**  
returns dict mapping player id to player name

**player\_team\_map**  
returns dict mapping player id to team id

**team\_items**  
returns list of team boxscore items

### data.nba.com Boxscore Item

```
class pbpstats.resources.boxscore.data_nba_boxscore_item.DataNbaBoxscoreItem(item,
                                                                              team_id=None,
                                                                              team_abbreviation=None)
```

Bases: object

Class for boxscore items from data.nba.com

#### Parameters

- **item** (*dict*) – dict with boxscore stats from response
- **team\_id** (*int*) – (optional) team id is not in dict with stats and can be added in here
- **team\_abbreviation** (*str*) – (optional) team abbreviation is not in dict with stats and can be added in here

**data**  
returns boxscore data dict

### live Boxscore Item

```
class pbpstats.resources.boxscore.live_boxscore_item.LiveBoxscoreItem(item,
                                                                           team_id=None,
                                                                           team_abbreviation=None)
```

Bases: object

Class for boxscore items from live data endpoint

#### Parameters

- **item** (*dict*) – dict with boxscore stats from response
- **team\_id** (*int*) – (optional) team id is not in dict with stats and can be added in here
- **team\_abbreviation** (*str*) – (optional) team abbreviation is not in dict with stats and can be added in here

**data**  
returns boxscore data dict

**total\_seconds**

### stats.nba.com Boxscore Item

```
class pbpstats.resources.boxscore.stats_nba_boxscore_item.StatsNbaBoxscoreItem(item)
Bases: object
```

Class for boxscore items from stats.nba.com

**Parameters** **item** (*dict*) – dict with boxscore stats from response

**data**  
returns boxscore data dict

### 3.1.3.2 Enhanced PBP

#### data.nba.com Enhanced PBP

##### Ejection

```
class pbpstats.resources.enhanced_pbp.data_nba.ejection.DataEjection(*args)
    Bases: pbpstats.resources.enhanced_pbp.ejection.Ejection, pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem

    Class for Ejection events

    event_type = 11
```

##### End of Period

```
class pbpstats.resources.enhanced_pbp.data_nba.end_of_period.DataEndOfPeriod(*args)
    Bases: pbpstats.resources.enhanced_pbp.end_of_period.EndOfPeriod, pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem

    Class for end of period events

    event_type = 13
```

##### Enhanced PBP Factory

DataNbaEnhancedPbpFactory can be used to create enhanced pbp event objects from the pbpstats.resources.enhanced\_pbp.data\_nba module based on the event type.

The following code will get the event class for event type 1 (which is the event type for a field goal make)

```
from pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_factory import _
↳ DataNbaEnhancedPbpFactory

factory = DataNbaEnhancedPbpFactory()
event_class = factory.get_event_class(1)
print(event_class) # prints "<class 'pbpstats.resources.enhanced_pbp.data_nba.field_
↳ goal.DataFieldGoal'>"
```

```
class pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_factory.DataNbaEnhancedPbpFactory
    Bases: object

    Class for factory of event type classes. On initialization will load in all event classes in the pbpstats.resources.enhanced_pbp.data_nba module

    get_event_class(event_type)
        Gets the class for the event based on the event_type

        Parameters event_type (int) – event action type for the event

        Returns class for event type
```

##### Enhanced PBP Item

DataEnhancedPbpItem is the base class for all data.nba.com enhanced pbp event types

---

```
class pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem(item,
                                                                    pe-
                                                                    riod,
                                                                    game_id)
```

Bases: `pbpstats.resources.enhanced_pbp.enhanced_pbp_item.EnhancedPbpItem`

Base class for enhanced pbp events from data.nba.com

#### Parameters

- **item** (*dict*) – dict with event data
- **period** (*int*) – period in which event occurs
- **game\_id** (*str*) – NBA Stats Game Id

#### data

returns event as a dict

#### event\_stats

returns list of dicts with all stats for event

#### get\_offense\_team\_id()

returns team id for team on offense for event

#### is\_possession\_ending\_event

returns True if event ends a possession, False otherwise

#### seconds\_remaining

returns seconds remaining in period as a float

### Field Goal

```
class pbpstats.resources.enhanced_pbp.data_nba.field_goal.DataFieldGoal(*args)
Bases:      pbpstats.resources.enhanced_pbp.field_goal.FieldGoal,      pbpstats.
resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem
```

Class for field goal events

**event\_type** = [1, 2]

#### is\_made

returns True if shot was made, False otherwise

#### shot\_value

returns 3 if shot is a 3 point attempt, 2 otherwise

### Foul

```
class pbpstats.resources.enhanced_pbp.data_nba.foul.DataFoul(*args)
Bases:      pbpstats.resources.enhanced_pbp.foul.Foul,      pbpstats.resources.
enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem
```

Class for foul events

**event\_type** = 6

**is\_away\_from\_play\_foul**

**is\_charge**

**is\_clear\_path\_foul**

`is_defensive_3_seconds`  
`is_delay_of_game`  
`is_double_foul`  
`is_double_technical`  
`is_flagrant1`  
`is_flagrant2`  
`is_inbound_foul`  
`is_loose_ball_foul`  
`is_offensive_foul`  
`is_personal_block_foul`  
`is_personal_foul`  
`is_personal_take_foul`  
`is_shooting_block_foul`  
`is_shooting_foul`  
`is_technical`  
`is_transition_take_foul`  
`number_of_fta_for_foul`  
    returns the number of free throws resulting from the foul

## Free Throw

```
class pbpstats.resources.enhanced_pbp.data_nba.free_throw.DataFreeThrow(*args)
    Bases: pbpstats.resources.enhanced_pbp.free_throw.FreeThrow, pbpstats.
resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem

    Class for free throw events

    event_type = 3

    is_flagrant_ft

    is_ft_1_of_1

    is_ft_1_of_2

    is_ft_1_of_3

    is_ft_1pt
        returns True if free throw is a 1 point free throw, False otherwise Only used in g-league, starting in 2019-20
        season

    is_ft_2_of_2

    is_ft_2_of_3

    is_ft_2pt
        returns True if free throw is a 2 point free throw, False otherwise Only used in g-league, starting in 2019-20
        season

    is_ft_3_of_3
```



**is\_ft\_3pt**  
returns True if free throw is a 3 point free throw, False otherwise Only used in g-league, starting in 2019-20 season

**is\_made**  
returns True if shot was made, False otherwise

**is\_technical\_ft**

## Jump Ball

**class** `pbpstats.resources.enhanced_pbp.data_nba.jump_ball.DataJumpBall(*args)`  
Bases: `pbpstats.resources.enhanced_pbp.jump_ball.JumpBall`, `pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem`  
Class for jump ball events  
**event\_type** = 10

## Rebound

**class** `pbpstats.resources.enhanced_pbp.data_nba.rebound.DataRebound(*args)`  
Bases: `pbpstats.resources.enhanced_pbp.rebound.Rebound`, `pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem`  
Class for rebound events  
**event\_type** = 4  
**is\_placeholder**  
returns True if rebound is a placeholder event, False otherwise.  
These are team rebounds on for example missed FT 1 of 2  
**missed\_shot**  
returns *FieldGoal* or *FreeThrow* object for shot that was missed  
**Raises** *EventOrderError*: If rebound event is not immediately following a missed shot event.  
**oreb**  
returns True if rebound is an offensive rebound, False otherwise

## Replay

**class** `pbpstats.resources.enhanced_pbp.data_nba.replay.DataReplay(*args)`  
Bases: `pbpstats.resources.enhanced_pbp.replay.Replay`, `pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem`  
Class for replay events  
**event\_type** = 18  
**overturn\_ruling**  
**ruling\_stands**  
**support\_ruling**

## Start of Period

```
class pbpstats.resources.enhanced_pbp.data_nba.start_of_period.DataStartOfPeriod(*args)
    Bases: pbpstats.resources.enhanced_pbp.start_of_period.StartOfPeriod,
pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_item.
DataEnhancedPbpItem
```

Class for start of period events

**event\_type** = 12

**get\_period\_starters** (*file\_directory=None*)

Gets player ids of players who started the period for each team If players can't be determined from parsing pbp, will try to find them by making API request to stats.nba.com boxscore filtered by time.

**Parameters** **file\_directory** (*str*) – directory in which overrides subdirectory exists containing period starter overrides when period starters can't be determined from parsing pbp events

**Returns** dict with list of player ids for each team with players on the floor at start of period

**Raises** *InvalidNumberOfStartersException*: If all 5 players that start the period for a team can't be determined.

## Substitution

```
class pbpstats.resources.enhanced_pbp.data_nba.substitution.DataSubstitution(*args)
    Bases: pbpstats.resources.enhanced_pbp.substitution.Substitution, pbpstats.
resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem
```

Class for Substitution events

**event\_type** = 8

**incoming\_player\_id**

returns player id of player coming in to the game

**outgoing\_player\_id**

returns player id of player going out of the game

## Timeout

```
class pbpstats.resources.enhanced_pbp.data_nba.timeout.DataTimeout(*args)
    Bases: pbpstats.resources.enhanced_pbp.timeout.Timeout, pbpstats.resources.
enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem
```

Class for timeout events

**event\_type** = 9

## Turnover

```
class pbpstats.resources.enhanced_pbp.data_nba.turnover.DataTurnover(*args)
    Bases: pbpstats.resources.enhanced_pbp.turnover.Turnover, pbpstats.resources.
enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem
```

Class for Turnover events

```
event_type = 5
is_3_second_violation
is_bad_pass
is_bad_pass_out_of_bounds
is_kicked_ball
is_lane_violation
is_lost_ball
is_lost_ball_out_of_bounds
is_no_turnover
is_offensive_goaltending
is_shot_clock_violation
is_step_out_of_bounds
is_travel
```

## Violation

```
class pbpstats.resources.enhanced_pbp.data_nba.violation.DataViolation(*args)
    Bases: pbpstats.resources.enhanced_pbp.violation.Violation, pbpstats.
            resources.enhanced_pbp.data_nba.enhanced_pbp_item.DataEnhancedPbpItem
    Class for violation events
    event_type = 7
    is_delay_of_game
    is_double_lane_violation
    is_goaltend_violation
    is_jumpball_violation
    is_kicked_ball_violation
    is_lane_violation
```

## data.nba.com Enhanced PBP

## Ejection

```
class pbpstats.resources.enhanced_pbp.live.ejection.LiveEjection(*args)
    Bases: pbpstats.resources.enhanced_pbp.ejection.Ejection, pbpstats.resources.
            enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem
    Class for Ejection events
    action_type = 'ejection'
```

## End of Period

```
class pbpstats.resources.enhanced_pbp.live.end_of_period.LiveEndOfPeriod(*args)
    Bases: pbpstats.resources.enhanced_pbp.end_of_period.EndOfPeriod, pbpstats.
resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem
```

Class for end of period events

**action\_type** = 'period'

**sub\_type** = 'end'

## Enhanced PBP Factory

LiveEnhancedPbpFactory can be used to create enhanced pbp event objects from the pbpstats.resources.enhanced\_pbp.live module based on the event type.

The following code will get the event class for a turnover event

```
from pbpstats.resources.enhanced_pbp.live.enhanced_pbp_factory import _
↳ LiveEnhancedPbpFactory

factory = LiveEnhancedPbpFactory()
event_class = factory.get_event_class('turnover', 'out-of-bounds')
print(event_class) # prints "<class 'pbpstats.resources.enhanced_pbp.live.turnover.
↳ LiveTurnover'>"
```

```
class pbpstats.resources.enhanced_pbp.live.enhanced_pbp_factory.LiveEnhancedPbpFactory
    Bases: object
```

Class for factory of event type classes. On initialization will load in all event classes in the pbpstats.resources.enhanced\_pbp.live module

**get\_event\_class** (*action\_type*, *sub\_type*)

Gets the class for the event based on the event\_type

**Parameters** **action\_type** (*str*) – event action type for the event

**Returns** class for event type

## Enhanced PBP Item

LiveEnhancedPbpItem is the base class for all live data enhanced pbp event types

```
class pbpstats.resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem(item,
                                                                                   game_id)
```

Bases: *pbpstats.resources.enhanced\_pbp.enhanced\_pbp\_item.EnhancedPbpItem*

Base class for enhanced pbp events from live data

**Parameters**

- **item** (*dict*) – dict with event data
- **period** (*int*) – period in which event occurs
- **game\_id** (*str*) – NBA Stats Game Id

**data**

returns event as a dict

**event\_stats**  
returns list of dicts with all stats for event

**get\_offense\_team\_id()**  
returns team id for team on offense for event

**is\_possession\_ending\_event**  
returns True if event ends a possession, False otherwise

**seconds\_remaining**  
returns seconds remaining in period as a float

**stripped\_descriptor**

**stripped\_sub\_type**

## Field Goal

```
class pbpstats.resources.enhanced_pbp.live.field_goal.LiveFieldGoal(*args)
    Bases: pbpstats.resources.enhanced_pbp.field_goal.FieldGoal, pbpstats.
            resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem

    Class for field goal events

    action_type = ['2pt', '3pt']

    is_made
        returns True if shot was made, False otherwise

    shot_value
        returns 3 if shot is a 3 point attempt, 2 otherwise
```

## Foul

```
class pbpstats.resources.enhanced_pbp.live.foul.LiveFoul(*args)
    Bases: pbpstats.resources.enhanced_pbp.foul.Foul, pbpstats.resources.
            enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem

    Class for foul events

    action_type = 'foul'

    is_away_from_play_foul

    is_charge

    is_clear_path_foul

    is_defensive_3_seconds

    is_delay_of_game

    is_double_foul

    is_double_technical

    is_flagrant1

    is_flagrant2

    is_inbound_foul

    is_loose_ball_foul
```

```
is_offensive_foul
is_personal_block_foul
is_personal_foul
is_personal_take_foul
is_shooting_block_foul
is_shooting_foul
is_technical
is_transition_take_foul
number_of_fta_for_foul
    returns the number of free throws resulting from the foul
```

## Free Throw

```
class pbpstats.resources.enhanced_pbp.live.free_throw.LiveFreeThrow(*args)
    Bases: pbpstats.resources.enhanced_pbp.free_throw.FreeThrow, pbpstats.
resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem

    Class for free throw events

    action_type = 'freethrow'

    is_away_from_play_ft
        returns True if free throw is from an away from the play foul, False otherwise.

    is_flagrant_ft
        returns True if free throw is from a flagrant foul, False otherwise.

    is_ft_1_of_1
    is_ft_1_of_2
    is_ft_1_of_3
    is_ft_1pt
        returns True if free throw is a 1 point free throw, False otherwise Only used in g-league, starting in 2019-20
        season
    is_ft_2_of_2
    is_ft_2_of_3
    is_ft_2pt
        returns True if free throw is a 2 point free throw, False otherwise Only used in g-league, starting in 2019-20
        season
    is_ft_3_of_3
    is_ft_3pt
        returns True if free throw is a 3 point free throw, False otherwise Only used in g-league, starting in 2019-20
        season
    is_made
        returns True if shot was made, False otherwise
    is_technical_ft
```

## Jump Ball

```
class pbpstats.resources.enhanced_pbp.live.jump_ball.LiveJumpBall(*args)
    Bases: pbpstats.resources.enhanced_pbp.jump_ball.JumpBall, pbpstats.
resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem

    Class for jump ball events

    action_type = 'jumpball'
```

## Rebound

```
class pbpstats.resources.enhanced_pbp.live.rebound.LiveRebound(*args)
    Bases: pbpstats.resources.enhanced_pbp.rebound.Rebound, pbpstats.resources.
enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem

    Class for rebound events

    action_type = 'rebound'

    is_placeholder
        returns True if rebound is a placeholder event, False otherwise.

        These are team rebounds on for example missed FT 1 of 2

    missed_shot
        returns FieldGoal or FreeThrow object for shot that was missed

        Raises EventOrderError: If rebound event is not immediately following a missed shot
        event.

    oreb
        returns True if rebound is an offensive rebound, False otherwise
```

## Replay

```
class pbpstats.resources.enhanced_pbp.live.replay.LiveReplay(*args)
    Bases: pbpstats.resources.enhanced_pbp.replay.Replay, pbpstats.resources.
enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem

    Class for replay events

    action_type = 'replay'

    overturn_ruling

    ruling_stands

    support_ruling
```

## Start of Period

```
class pbpstats.resources.enhanced_pbp.live.start_of_period.LiveStartOfPeriod(*args)
    Bases: pbpstats.resources.enhanced_pbp.start_of_period.StartOfPeriod,
pbpstats.resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem

    Class for start of period events

    action_type = 'period'
```

**get\_period\_starters** (*file\_directory=None, ignore\_missing\_starters=False*)

Gets player ids of players who started the period for each team

**Parameters**

- **file\_directory** (*str*) – directory in which overrides subdirectory exists containing period starter overrides when period starters can't be determined from parsing pbp events
- **ignore\_missing\_starters** (*bool*) – when True won't raise missing starters exception

**Returns** dict with list of player ids for each team with players on the floor at start of period

**Raises** *InvalidNumberOfStartersException*: If all 5 players that start the period for a team can't be determined.

**sub\_type** = 'start'

## Substitution

**class** pbpstats.resources.enhanced\_pbp.live.substitution.**LiveSubstitution** (\*args)  
Bases: *pbpstats.resources.enhanced\_pbp.substitution.Substitution, pbpstats.resources.enhanced\_pbp.live.enhanced\_pbp\_item.LiveEnhancedPbpItem*

Class for Substitution events

**action\_type** = 'substitution'

**current\_players**

returns dict with list of player ids for each team with players on the floor following the sub

**incoming\_player\_id**

returns player id of player coming in to the game

**outgoing\_player\_id**

returns player id of player coming in to the game

## Timeout

**class** pbpstats.resources.enhanced\_pbp.live.timeout.**LiveTimeout** (\*args)  
Bases: *pbpstats.resources.enhanced\_pbp.timeout.Timeout, pbpstats.resources.enhanced\_pbp.live.enhanced\_pbp\_item.LiveEnhancedPbpItem*

Class for timeout events

**action\_type** = 'timeout'

## Turnover

**class** pbpstats.resources.enhanced\_pbp.live.turnover.**LiveTurnover** (\*args)  
Bases: *pbpstats.resources.enhanced\_pbp.turnover.Turnover, pbpstats.resources.enhanced\_pbp.live.enhanced\_pbp\_item.LiveEnhancedPbpItem*

Class for Turnover events

**action\_type** = 'turnover'

**is\_3\_second\_violation**



```
is_bad_pass
is_bad_pass_out_of_bounds
is_kicked_ball
is_lane_violation
is_lost_ball
is_lost_ball_out_of_bounds
is_no_turnover
is_offensive_goaltending
is_shot_clock_violation
is_step_out_of_bounds
is_travel
```

## Violation

```
class pbpstats.resources.enhanced_pbp.live.violation.LiveViolation(*args)
    Bases: pbpstats.resources.enhanced_pbp.violation.Violation, pbpstats.
            resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem
    Class for violation events
    action_type = 'violation'
    is_delay_of_game
    is_double_lane_violation
    is_goaltend_violation
    is_jumpball_violation
    is_kicked_ball_violation
    is_lane_violation
```

## stats.nba.com Enhanced PBP

### Ejection

```
class pbpstats.resources.enhanced_pbp.stats_nba.ejection.StatsEjection(*args)
    Bases: pbpstats.resources.enhanced_pbp.ejection.Ejection, pbpstats.resources.
            enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem
    Class for Ejection events
    event_type = 11
```

## End of Period

```
class pbpstats.resources.enhanced_pbp.stats_nba.end_of_period.StatsEndOfPeriod(*args)
    Bases: pbpstats.resources.enhanced_pbp.end_of_period.EndOfPeriod, pbpstats.
            resources.enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem

    Class for end of period events

    event_type = 13
```

## Enhanced PBP Factory

StatsNbaEnhancedPbpFactory can be used to create enhanced pbp event objects from the pbpstats.resources.enhanced\_pbp.stats\_nba module based on the event type.

The following code will get the event class for event type 1 (which is the event type for a field goal make)

```
from pbpstats.resources.enhanced_pbp.stats_nba.enhanced_pbp_factory import _
↳ StatsNbaEnhancedPbpFactory

factory = StatsNbaEnhancedPbpFactory()
event_class = factory.get_event_class(1)
print(event_class) # prints "<class 'pbpstats.resources.enhanced_pbp.stats_nba.field_
↳ goal.StatsFieldGoal'>"
```

```
class pbpstats.resources.enhanced_pbp.stats_nba.enhanced_pbp_factory.StatsNbaEnhancedPbpFactory
    Bases: object

    Class for factory of event type classes. On initialization will load in all event classes in the pbpstats.
    resources.enhanced_pbp.stats_nba module

    get_event_class(event_type)
        Gets the class for the event based on the event_type

        Parameters event_type (int) – event action type for the event

        Returns class for event type
```

## Enhanced PBP Item

StatsEnhancedPbpItem is the base class for all stats.nba.com enhanced pbp event types

```
class pbpstats.resources.enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem(event_type, order)
    Bases: pbpstats.resources.enhanced_pbp.enhanced_pbp_item.EnhancedPbpItem

    Base class for enhanced pbp events from stats.nba.com

    Parameters
        • event (dict) – dict with event data
        • order (int) – sequential order in which event occurs

    data
        returns event as a dict

    get_offense_team_id()
        returns team id for team on offense for event
```

**is\_possession\_ending\_event**  
returns True if event ends a possession, False otherwise

**seconds\_remaining**  
returns seconds remaining in period as a float

**video\_url**  
returns url for mp4 video of play, if available

## Field Goal

```
class pbpstats.resources.enhanced_pbp.stats_nba.field_goal.StatsFieldGoal(*args)
    Bases: pbpstats.resources.enhanced_pbp.field_goal.FieldGoal, pbpstats.
resources.enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem

    Class for field goal events

    event_type = [1, 2]

    get_offense_team_id()
        returns team id that took the shot

    is_made
        returns True if shot was made, False otherwise

    shot_value
        returns 3 if shot is a 3 point attempt, 2 otherwise
```

## Foul

```
class pbpstats.resources.enhanced_pbp.stats_nba.foul.StatsFoul(*args)
    Bases: pbpstats.resources.enhanced_pbp.foul.Foul, pbpstats.resources.
enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem

    Class for foul events

    event_type = 6

    is_away_from_play_foul

    is_charge

    is_clear_path_foul

    is_defensive_3_seconds

    is_delay_of_game

    is_double_foul

    is_double_technical

    is_flagrant1

    is_flagrant2

    is_inbound_foul

    is_loose_ball_foul

    is_offensive_foul

    is_personal_block_foul
```

`is_personal_foul`  
`is_personal_take_foul`  
`is_shooting_block_foul`  
`is_shooting_foul`  
`is_technical`  
`is_transition_take_foul`  
`number_of_fta_for_foul`  
returns the number of free throws resulting from the foul

## Free Throw

```
class pbpstats.resources.enhanced_pbp.stats_nba.free_throw.StatsFreeThrow(*args)
    Bases: pbpstats.resources.enhanced_pbp.free_throw.FreeThrow, pbpstats.
            resources.enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem
    Class for free throw events

    event_type = 3

    get_offense_team_id()
        returns team id that took the shot

    is_flagrant_ft
    is_ft_1_of_1
    is_ft_1_of_2
    is_ft_1_of_3
    is_ft_1pt
        returns True if free throw is a 1 point free throw, False otherwise Only used in g-league, starting in 2019-20
        season
    is_ft_2_of_2
    is_ft_2_of_3
    is_ft_2pt
        returns True if free throw is a 2 point free throw, False otherwise Only used in g-league, starting in 2019-20
        season
    is_ft_3_of_3
    is_ft_3pt
        returns True if free throw is a 3 point free throw, False otherwise Only used in g-league, starting in 2019-20
        season
    is_made
        returns True if shot was made, False otherwise
    is_technical_ft
```

## Jump Ball

```
class pbpstats.resources.enhanced_pbp.stats_nba.jump_ball.StatsJumpBall(*args)
    Bases: pbpstats.resources.enhanced_pbp.jump_ball.JumpBall, pbpstats.resources.enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem

    Class for jump ball events

    event_type = 10

    get_offense_team_id()
        returns team id for team on offense for event
```

## Rebound

```
class pbpstats.resources.enhanced_pbp.stats_nba.rebound.StatsRebound(*args)
    Bases: pbpstats.resources.enhanced_pbp.rebound.Rebound, pbpstats.resources.enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem

    Class for rebound events

    event_type = 4

    get_offense_team_id()
        returns team id for team on offense for the shot that was rebounded

    is_placeholder
        returns True if rebound is a placeholder event, False otherwise.

        These are team rebounds on for example missed FT 1 of 2

    missed_shot
        returns FieldGoal or FreeThrow object for shot that was missed

        Raises EventOrderError: If rebound event is not immediately following a missed shot event.

    oreb
        returns True if rebound is an offensive rebound, False otherwise
```

## Replay

```
class pbpstats.resources.enhanced_pbp.stats_nba.replay.StatsReplay(*args)
    Bases: pbpstats.resources.enhanced_pbp.replay.Replay, pbpstats.resources.enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem

    Class for replay events

    event_type = 18

    overturn_ruling

    ruling_stands

    support_ruling
```

## Start of Period

```
class pbpstats.resources.enhanced_pbp.stats_nba.start_of_period.StatsStartOfPeriod(*args)
    Bases: pbpstats.resources.enhanced_pbp.start_of_period.StartOfPeriod,
pbpstats.resources.enhanced_pbp.stats_nba.enhanced_pbp_item.
StatsEnhancedPbpItem
```

Class for start of period events

**event\_type** = 12

**get\_period\_starters** (*file\_directory=None*)

Gets player ids of players who started the period for each team. If players can't be determined from parsing pbp, will try to find them by making API request to stats.nba.com boxscore filtered by time.

**Parameters** **file\_directory** (*str*) – directory in which overrides subdirectory exists containing period starter overrides when period starters can't be determined from parsing pbp events

**Returns** dict with list of player ids for each team with players on the floor at start of period

**Raises** *InvalidNumberOfStartersException*: If all 5 players that start the period for a team can't be determined.

## Substitution

```
class pbpstats.resources.enhanced_pbp.stats_nba.substitution.StatsSubstitution(*args)
    Bases: pbpstats.resources.enhanced_pbp.substitution.Substitution, pbpstats.
resources.enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem
```

Class for Substitution events

**event\_type** = 8

**incoming\_player\_id**

returns player id of player coming in to the game

**outgoing\_player\_id**

returns player id of player going out of the game

## Timeout

```
class pbpstats.resources.enhanced_pbp.stats_nba.timeout.StatsTimeout(*args)
    Bases: pbpstats.resources.enhanced_pbp.timeout.Timeout, pbpstats.resources.
enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem
```

Class for timeout events

**event\_type** = 9

## Turnover

```
class pbpstats.resources.enhanced_pbp.stats_nba.turnover.StatsTurnover(*args)
    Bases: pbpstats.resources.enhanced_pbp.turnover.Turnover, pbpstats.resources.
enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem
```

Class for Turnover events

```
event_type = 5
get_offense_team_id()
    returns team id for team on offense for event
is_3_second_violation
is_bad_pass
is_bad_pass_out_of_bounds
is_kicked_ball
is_lane_violation
is_lost_ball
is_lost_ball_out_of_bounds
is_no_turnover
is_offensive_goaltending
is_shot_clock_violation
is_step_out_of_bounds
is_travel
```

## Violation

```
class pbpstats.resources.enhanced_pbp.stats_nba.violation.StatsViolation(*args)
    Bases: pbpstats.resources.enhanced\_pbp.violation.Violation, pbpstats.resources.enhanced\_pbp.stats\_nba.enhanced\_pbp\_item.StatsEnhancedPbpItem
    Class for violation events
    event_type = 7
    is_delay_of_game
    is_double_lane_violation
    is_goaltend_violation
    is_jumpball_violation
    is_kicked_ball_violation
    is_lane_violation
```

## Ejection

```
class pbpstats.resources.enhanced_pbp.ejection.Ejection
    Bases: object
    Class for Ejection events
    event_stats
        returns list of dicts with all stats for event
```

## End of Period

```
class pbpstats.resources.enhanced_pbp.end_of_period.EndOfPeriod
    Bases: object

    Class for end of period events

    event_stats
        returns list of dicts with all stats for event
```

## Enhanced PBP

The EnhancedPbp class has some basic properties for handling enhanced pbp data

```
class pbpstats.resources.enhanced_pbp.enhanced_pbp.EnhancedPbp(items)
    Bases: pbpstats.resources.base.Base

    Class for enhanced play-by-play events

        Parameters events (list) – list of EnhancedPbpItem items, typically from a enhanced pbp
            data loader

    data
        returns list of pbp event dicts

    fgas
        returns list of FieldGoal events

    fgms
        returns list of FieldGoal events with all made FGs

    ftas
        returns list of FreeThrow events

    rebounds
        returns list of Rebound events

    turnovers
        returns list of Turnover events
```

## Enhanced PBP Item

EnhancedPbpItem is an abstract base class for all enhanced pbp event types

```
class pbpstats.resources.enhanced_pbp.enhanced_pbp_item.EnhancedPbpItem
    Bases: object

    base_stats
        returns list of dicts with all seconds played and possession count stats for event

    count_as_possession
        returns True if event is possession changing event that should count as a real possession, False otherwise.

        In order to not include possessions which a very low probability of scoring in possession counts, possession
        won't be counted as a possession if it starts with <= 2 seconds left and no points are scored before period
        ends

    current_players
        returns dict with list of player ids for each team with players on the floor for current event
```



For all non substitution events current players are just the same as previous event

This gets overwritten in *Substitution* since those are the only event types where players are not the same as the previous event

**event\_stats**

returns list of dicts with all stats for event

**get\_all\_events\_at\_current\_time()**

returns list of all events that take place at the same time as the current event

**get\_offense\_team\_id()**

returns team id for team on offense for event

**is\_penalty\_event()**

returns True if the team on offense is in the penalty, False otherwise

**is\_possession\_ending\_event**

returns True if event ends a possession, False otherwise

**is\_second\_chance\_event()**

returns True if the event takes place after an offensive rebound on the current possession, False otherwise

**lineup\_ids**

returns dict with lineup ids for each team for current event. Lineup ids are hyphen separated sorted player id strings.

**score\_margin**

returns the score margin from perspective of offense team before the event took place

**seconds\_remaining**

returns seconds remaining in period as a float

**seconds\_since\_previous\_event**

returns the number of seconds that have elapsed since the previous event

## Field Goal

```
class pbpstats.resources.enhanced_pbp.field_goal.FieldGoal
```

Bases: object

Class for field goal events

**distance**

returns shot distance in feet

**event\_stats**

returns list of dicts with all stats for event

**is\_and1**

returns True if shot was an and 1, False otherwise

**is\_assisted**

returns True if shot was assisted, False otherwise

**is\_blocked**

returns True if shot was blocked, False otherwise

**is\_corner\_3**

returns True if shot was a corner 3, False otherwise

**is\_heave**  
returns True if shot was a last second heave, False otherwise

**is\_made**  
returns True if shot was made, False otherwise

**is\_make\_that\_does\_not\_end\_possession**  
returns True if shot is a made shot that does not end the possession due to a foul, False otherwise

**is\_putback**  
returns True if shot is a 2pt attempt within 2 seconds of an offensive rebound attempted by the same player who got the rebound

**rebound**  
returns *Rebound* item for the rebound of the shot, if it was missed, None otherwise

**shot\_data**  
returns a dict with detailed shot data

**shot\_type**  
returns shot type string ('AtRim', 'ShortMidRange', 'LongMidRange', 'Arc3' or 'Corner3')

**shot\_value**  
returns 3 if shot is a 3 point attempt, 2 otherwise

## Foul

```
class pbpstats.resources.enhanced_pbp.foul.Foul
    Bases: object

    Class for foul events

    counts_as_personal_foul
        returns True if fouls is a foul type that counts as a personal foul, False otherwise

    counts_towards_penalty
        returns True if foul is a foul type that counts towards the penalty, False otherwise

    event_stats
        returns list of dicts with all stats for event

    foul_type_string
        returns string description of foul type

    is_away_from_play_foul

    is_charge

    is_clear_path_foul

    is_defensive_3_seconds

    is_delay_of_game

    is_double_foul

    is_double_technical

    is_flagrant

    is_flagrant1

    is_flagrant2
```

`is_inbound_foul`  
`is_loose_ball_foul`  
`is_offensive_foul`  
`is_personal_block_foul`  
`is_personal_foul`  
`is_personal_take_foul`  
`is_shooting_block_foul`  
`is_shooting_foul`  
`is_technical`  
`is_transition_take_foul`  
`number_of_fta_for_foul`  
returns the number of free throws resulting from the foul

## Free Throw

`class pbpstats.resources.enhanced_pbp.free_throw.FreeThrow`

Bases: `object`

Class for free throw events

`event_for_efficiency_stats`

returns *Foul* object for the foul that resulted in the free throw. Plus/minus points should go to the players on the floor at the time of the foul, not the free throw.

`event_stats`

returns list of dicts with all stats for event

`foul_that_led_to_ft`

returns *Foul* object for the foul that resulted in the free throw

`free_throw_type`

returns string description of free throw type

`is_away_from_play_ft`

returns True if free throw is from an away from the play foul, False otherwise.

`is_end_ft`

returns True if free throw is last of trip to the free throw line, False otherwise

`is_first_ft`

returns True if free throw is first of trip to the free throw line, False otherwise

`is_ft_1_of_1`

`is_ft_1_of_2`

`is_ft_1_of_3`

`is_ft_1pt`

`is_ft_2_of_2`

`is_ft_2_of_3`

`is_ft_2pt`

**is\_ft\_3\_of\_3**

**is\_ft\_3pt**

**is\_inbound\_foul\_ft**

returns True if free throw is from an inbound foul, False otherwise.

**is\_made**

returns True if shot was made, False otherwise

**is\_technical\_ft**

**is\_transition\_take\_foul\_ft**

returns True if free throw is from an transition take foul, False otherwise.

**num\_ft\_for\_trip**

returns number of shots for the trip to the free throw line

**shot\_type = 'FT'**

**shot\_value**

returns shot value of a free throw Starting in 2019-20 season, the G-League added 2 and 3 point FTs

## Jump Ball

**class** pbpstats.resources.enhanced\_pbp.jump\_ball.**JumpBall**

Bases: object

Class for jump ball events

**event\_stats**

returns list of dicts with all stats for event

**winning\_team**

returns team id that won the jump ball

## Rebound

**exception** pbpstats.resources.enhanced\_pbp.rebound.**EventOrderError**

Bases: Exception

Class for exception raised when rebound event is not immediately following a missed shot event.

You can manually edit the event order in the pbp file stored on disk to fix this.

**class** pbpstats.resources.enhanced\_pbp.rebound.**Rebound**

Bases: object

Class for rebound events

**event\_stats**

returns list of dicts with all stats for event

**is\_buzzer\_beater\_placeholder**

returns True if rebound is a placeholder event after a missed buzzer beater, False otherwise.

Rebounds occur after time has expired but are still logged in play-by-play, but should not be counted in rebound totals

**is\_buzzer\_beater\_rebound\_at\_shot\_time**

returns True if rebound is a placeholder event after a missed buzzer beater, False otherwise.

Sometimes rebound on buzzer beater is given the same time as shot - don't count these. Only don't count if rebound is last event before end of period event, ignoring replay events

**is\_non\_live\_ft\_placeholder**

returns True if rebound is a placeholder event after a missed free throw that is not a live ball, False otherwise.

Example: rebound after missed flagrant FT 2 of 2

**is\_placeholder**

returns True if rebound is a placeholder event, False otherwise.

These are team rebounds on for example missed FT 1 of 2

**is\_real\_rebound**

Returns True if rebound should be counted as a rebound, False otherwise.

All missed shots have a rebound in the play-by-play but not all of these rebounds should be counted as actual rebounds. Some are just placeholder events.

**is\_turnover\_placeholder**

returns True if rebound is a placeholder event when a turnover occurs, False otherwise.

Example shot clock violation or kicked ball turnover at time of team rebound

**missed\_shot****oreb****self\_reb**

returns True if rebound was gotten by player who missed the shot, False otherwise

## Replay

**class** pbpstats.resources.enhanced\_pbp.replay.Replay

Bases: object

Class for replay events

**event\_stats**

returns list of dicts with all stats for event

**overturn\_ruling****ruling\_stands****support\_ruling**

## Start of Period

**exception** pbpstats.resources.enhanced\_pbp.start\_of\_period.InvalidNumberOfStartersException

Bases: Exception

Class for exception when a team's 5 period starters can't be determined.

You can add the correct period starters to overrides/missing\_period\_starters.json in your data directory to fix this.

```
class pbpstats.resources.enhanced_pbp.start_of_period.StartOfPeriod
    Bases: object

    Class for start of period events

    current_players
        returns period starters

    event_stats
        returns list of dicts with all stats for event

    get_offense_team_id()
        returns team id for team on starting period on offense

    classmethod get_period_starters (file_directory)
        Gets player ids of players who started the period for each team

        Parameters file_directory (str) – directory in which overrides subdirectory exists containing period starter overrides when period starters can't be determined from parsing pbp events

        Returns dict with list of player ids for each team with players on the floor at start of period

        Raises InvalidNumberOfStartersException: If all 5 players that start the period for a team can't be determined.

    get_team_starting_with_ball()
        returns team id for team on starting period with the ball

    league
        Returns League for game id.

        First 2 in game id represent league - 00 for nba, 10 for wnba, 20 for g-league

    league_url_part
```

## Substitution

```
class pbpstats.resources.enhanced_pbp.substitution.Substitution
    Bases: object

    Class for Substitution events

    current_players
        returns dict with list of player ids for each team with players on the floor following the sub

    event_stats
        returns list of dicts with all stats for event

    incoming_player_id

    outgoing_player_id
```

## Timeout

```
class pbpstats.resources.enhanced_pbp.timeout.Timeout
    Bases: object

    Class for timeout events

    event_stats
        returns list of dicts with all stats for event
```

## Turnover

```
class pbpstats.resources.enhanced_pbp.turnover.Turnover
    Bases: object

    Class for Turnover events

    event_stats
        returns list of dicts with all stats for event

    classmethod is_3_second_violation()

    classmethod is_bad_pass()

    classmethod is_bad_pass_out_of_bounds()

    classmethod is_kicked_ball()

    classmethod is_lane_violation()

    classmethod is_lost_ball()

    classmethod is_lost_ball_out_of_bounds()

    classmethod is_no_turnover()

    classmethod is_offensive_goaltending()

    classmethod is_shot_clock_violation()

    is_steal

    classmethod is_step_out_of_bounds()

    classmethod is_travel()
```

## Violation

```
class pbpstats.resources.enhanced_pbp.violation.Violation
    Bases: object

    Class for violation events

    event_stats
        returns list of dicts with all stats for event

    classmethod is_delay_of_game()

    classmethod is_double_lane_violation()

    classmethod is_goaltend_violation()

    classmethod is_jumpball_violation()

    classmethod is_kicked_ball_violation()

    classmethod is_lane_violation()
```

### 3.1.3.3 Games

### data.nba.com Game Item

**class** pbpstats.resources.games.data\_nba\_game\_item.**DataNbaGameItem**(*item*)

Bases: object

Class for game data from data.nba.com

**Parameters** *item* (*dict*) – dict with game data

**data**

returns game dict

**is\_final**

returns True if game is final, False otherwise

### Games

The Games class has some basic properties for handling game data

**class** pbpstats.resources.games.games.**Games**(*items*)

Bases: pbpstats.resources.base.Base

Class for games items

**Parameters** *items* (*list*) – list of either *StatsNbaGameItem* or *DataNbaGameItem* items, typically from a games data loader

**data**

returns list of dicts with game items

**final\_games**

returns list of dicts with final game items

### stats.nba.com Game Item

**class** pbpstats.resources.games.stats\_nba\_game\_item.**StatsNbaGameItem**(*item*)

Bases: object

Class for game data from stats.nba.com

**Parameters** *item* (*dict*) – dict with game data

**data**

returns game dict

**is\_final**

returns True if game is final, False otherwise

### 3.1.3.4 PBP

#### data.nba.com PBP Item

**class** pbpstats.resources.pbp.data\_nba\_pbp\_item.**DataNbaPbpItem**(*item*, *period*)

Bases: object

Class for pbp events from data.nba.com

**Parameters**



- **item** (*dict*) – dict with event data
- **period** (*int*) – period in which event occurs

**data**  
returns pbp event dict

### live PBP Item

**class** `pbpstats.resources.pbp.live_pbp_item.LivePbpItem(item)`

Bases: `object`

Class for pbp events from live data

#### Parameters

- **item** (*dict*) – dict with event data
- **period** (*int*) – period in which event occurs

**data**  
returns pbp event dict

### PBP

The `Pbp` class has some basic properties for handling pbp data

**class** `pbpstats.resources.pbp.pbp.Pbp(items)`

Bases: `pbpstats.resources.base.Base`

Class for pbp items

**Parameters** **items** (*list*) – list of either `StatsNbaPbpItem` or `DataNbaPbpItem` items, typically from a pbp data loader

**data**  
returns list of dicts with each event

### stats.nba.com PBP Item

**class** `pbpstats.resources.pbp.stats_nba_pbp_item.StatsNbaPbpItem(event, order)`

Bases: `object`

Class for pbp events from stats.nba.com

#### Parameters

- **event** (*dict*) – dict with event data
- **order** (*int*) – sequential order in which event occurs

**data**  
returns pbp event dict

### 3.1.3.5 Possessions

#### Possession

The `Possession` class has some basic properties for handling possession data

**class** `pbpstats.resources.possessions.possession.Possession(events)`

Bases: `object`

Class for possession

**Parameters** `events` (*list*) – list of *EnhancedPbpItem* items for possession, typically from a possession data loader

**data**

returns dict possession data

**end\_time**

returns the time remaining (MM:SS) in the period when the possession ended

**get\_team\_ids()**

returns a list with the team ids of both teams playing

**offense\_team\_id**

returns team id for team on offense on possession

**possession\_has\_timeout**

returns True if there was a timeout called on the current possession, False otherwise

**possession\_start\_type**

returns possession start type string

**possession\_stats**

returns list of dicts with aggregate stats for possession

**previous\_possession\_end\_rebound\_player\_id**

returns player id of player who got rebound that ended previous possession. returns 0 if previous possession did not end with a live ball rebound

**previous\_possession\_end\_shooter\_player\_id**

returns player id of player who took shot (make or miss) that ended previous possession. returns 0 if previous possession did not end with made field goal or live ball rebound

**previous\_possession\_end\_steal\_player\_id**

returns player id of player who got steal that ended previous possession. returns 0 if previous possession did not end with a live ball turnover

**previous\_possession\_end\_turnover\_player\_id**

returns player id of player who turned ball over that ended previous possession. returns 0 if previous possession did not end with a live ball turnover

**previous\_possession\_ending\_event**

returns previous possession ending event - ignoring subs

**previous\_possession\_has\_timeout**

returns True if there was a timeout called at same time as possession ended, False otherwise

**start\_score\_margin**

returns the score margin from the perspective of the team on offense when the possession started

**start\_time**

returns the time remaining (MM:SS) in the period when the possession started

## Possessions

The `Possessions` class has some basic properties for aggregating possession stats

```
class pbpstats.resources.possessions.possessions.Possessions (items)
```

```
    Bases: pbpstats.resources.base.Base
```

```
    Class for possession items
```

```
        Parameters items (list) – list of Possession items, typically from a possession data loader
```

```
    data
```

```
        returns possessions dict
```

```
    lineup_opponent_stats
```

```
        returns list of dicts with aggregated stats by lineup opponent
```

```
    lineup_stats
```

```
        returns list of dicts with aggregated stats by lineup
```

```
    opponent_stats
```

```
        returns list of dicts with aggregated stats by opponent
```

```
    player_stats
```

```
        returns list of dicts with aggregated stats by player
```

```
    team_stats
```

```
        returns list of dicts with aggregated stats by team
```

### 3.1.3.6 Shots

#### Shots

The `Shots` class has some basic properties for handling shot data

```
class pbpstats.resources.shots.shots.Shots (items)
```

```
    Bases: pbpstats.resources.base.Base
```

```
    Class for games items
```

```
        Parameters items (list) – list of StatsNbaShot items, typically from a shots data loader
```

```
    data
```

```
        returns list of dicts with shots
```

#### stats.nba.com Shot

```
class pbpstats.resources.shots.stats_nba_shot.StatsNbaShot (item)
```

```
    Bases: object
```

```
    Class for shot data from stats.nba.com
```

```
        Parameters item (dict) – dict with shot data
```

```
    data
```

```
        returns shot data dict
```

### 3.1.4 Client

Instantiating a `Client` object will load data loader objects for resources specified in settings dict.

The following code will instantiate the client and get Possession data for game id 0021900001 from files in `/response_data` subdirectories

```
from pbpstats.client import Client

settings = {
    "dir": "/response_data",
    "Possessions": {"source": "file", "data_provider": "stats_nba"}
}
client = Client(settings)
game = client.Game('0021900001')
for possession in game.possessions.items:
    print(possession)
```

```
class pbpstats.client.Client(settings)
```

Bases: `object`

**Parameters** `settings` (*dict*) – Dict with data that specifies which data loaders should be used. `dir` key is optional, but recommended and should point to the directory you have set up that either already contains response data or where you want to store the response data. Other keys in the settings dict should be resources from the `resources` module and their values should be a dict with `source` ('file' or 'web') and `data_provider` ('stats\_nba' or 'data\_nba' or 'live')

---

## Python Module Index

---

### p

pbpstats.client, 64

pbpstats.data\_loader.data\_nba.boxscore.file, 10

pbpstats.data\_loader.data\_nba.boxscore.loader, 9

pbpstats.data\_loader.data\_nba.boxscore.web, 10

pbpstats.data\_loader.data\_nba.enhanced\_pbp.file, 11

pbpstats.data\_loader.data\_nba.enhanced\_pbp.loader, 10

pbpstats.data\_loader.data\_nba.enhanced\_pbp.web, 11

pbpstats.data\_loader.data\_nba.pbp.file, 12

pbpstats.data\_loader.data\_nba.pbp.loader, 11

pbpstats.data\_loader.data\_nba.pbp.web, 12

pbpstats.data\_loader.data\_nba.possessions.file, 13

pbpstats.data\_loader.data\_nba.possessions.loader, 12

pbpstats.data\_loader.data\_nba.possessions.web, 13

pbpstats.data\_loader.data\_nba.schedule.file, 14

pbpstats.data\_loader.data\_nba.schedule.loader, 13

pbpstats.data\_loader.data\_nba.schedule.web, 14

pbpstats.data\_loader.data\_nba.web\_loader, 15

pbpstats.data\_loader.factory, 29

pbpstats.data\_loader.live.boxscore.file, 16

pbpstats.data\_loader.live.boxscore.loader, 15

pbpstats.data\_loader.live.boxscore.web, 16

pbpstats.data\_loader.live.enhanced\_pbp.file, 17

pbpstats.data\_loader.live.enhanced\_pbp.loader, 16

pbpstats.data\_loader.live.enhanced\_pbp.web, 17

pbpstats.data\_loader.live.file\_loader, 19

pbpstats.data\_loader.live.pbp.file, 18

pbpstats.data\_loader.live.pbp.loader, 17

pbpstats.data\_loader.live.pbp.web, 18

pbpstats.data\_loader.live.possessions.file, 19

pbpstats.data\_loader.live.possessions.loader, 18

pbpstats.data\_loader.live.possessions.web, 19

pbpstats.data\_loader.live.web\_loader, 19

pbpstats.data\_loader.nba\_enhanced\_pbp\_loader, 30

pbpstats.data\_loader.nba\_possession\_loader, 30

pbpstats.data\_loader.stats\_nba.base, 20

pbpstats.data\_loader.stats\_nba.boxscore.file, 21

pbpstats.data\_loader.stats\_nba.boxscore.loader, 20

pbpstats.data\_loader.stats\_nba.boxscore.web, 21

pbpstats.data\_loader.stats\_nba.enhanced\_pbp.file, 22

pbpstats.data\_loader.stats\_nba.enhanced\_pbp.loader, 21

pbpstats.data\_loader.stats\_nba.enhanced\_pbp.web,

22 pbpstats.resources.enhanced\_pbp.data\_nba.end\_of\_per  
pbpstats.data\_loader.stats\_nba.file\_loader, 34  
29 pbpstats.resources.enhanced\_pbp.data\_nba.enhanced\_p  
pbpstats.data\_loader.stats\_nba.league\_game\_log, 34  
25 pbpstats.resources.enhanced\_pbp.data\_nba.enhanced\_p  
pbpstats.data\_loader.stats\_nba.league\_game\_log, 34  
25 pbpstats.resources.enhanced\_pbp.data\_nba.field\_goal  
pbpstats.data\_loader.stats\_nba.league\_game\_log, 35  
26 pbpstats.resources.enhanced\_pbp.data\_nba.foul,  
pbpstats.data\_loader.stats\_nba.pbp.file, 35  
23 pbpstats.resources.enhanced\_pbp.data\_nba.free\_throw  
pbpstats.data\_loader.stats\_nba.pbp.loader, 36  
23 pbpstats.resources.enhanced\_pbp.data\_nba.jump\_ball,  
pbpstats.data\_loader.stats\_nba.pbp.web, 37  
23 pbpstats.resources.enhanced\_pbp.data\_nba.rebound,  
pbpstats.data\_loader.stats\_nba.possessions.file, 37  
24 pbpstats.resources.enhanced\_pbp.data\_nba.replay,  
pbpstats.data\_loader.stats\_nba.possessions.loader, 37  
24 pbpstats.resources.enhanced\_pbp.data\_nba.start\_of\_p  
pbpstats.data\_loader.stats\_nba.possessions.web, 38  
24 pbpstats.resources.enhanced\_pbp.data\_nba.substitut  
pbpstats.data\_loader.stats\_nba.scoreboard.file, 38  
26 pbpstats.resources.enhanced\_pbp.data\_nba.timeout,  
pbpstats.data\_loader.stats\_nba.scoreboard.loader, 38  
26 pbpstats.resources.enhanced\_pbp.data\_nba.turnover,  
pbpstats.data\_loader.stats\_nba.scoreboard.web, 38  
27 pbpstats.resources.enhanced\_pbp.data\_nba.violation,  
pbpstats.data\_loader.stats\_nba.shots.file, 39  
28 pbpstats.resources.enhanced\_pbp.ejection,  
pbpstats.data\_loader.stats\_nba.shots.loader, 51  
27 pbpstats.resources.enhanced\_pbp.end\_of\_period,  
pbpstats.data\_loader.stats\_nba.shots.web, 52  
28 pbpstats.resources.enhanced\_pbp.enhanced\_pbp,  
pbpstats.data\_loader.stats\_nba.summary.file, 52  
29 pbpstats.resources.enhanced\_pbp.enhanced\_pbp\_item,  
pbpstats.data\_loader.stats\_nba.summary.loader, 52  
28 pbpstats.resources.enhanced\_pbp.field\_goal,  
pbpstats.data\_loader.stats\_nba.summary.web, 53  
29 pbpstats.resources.enhanced\_pbp.foul,  
pbpstats.data\_loader.stats\_nba.web\_loader, 54  
29 pbpstats.resources.enhanced\_pbp.free\_throw,  
pbpstats.objects.day, 30 55  
pbpstats.objects.game, 31 pbpstats.resources.enhanced\_pbp.jump\_ball,  
pbpstats.objects.season, 31 56  
pbpstats.resources.boxscore.boxscore, pbpstats.resources.enhanced\_pbp.live.ejection,  
32 39  
pbpstats.resources.boxscore.data\_nba\_boxscore\_item, pbpstats.resources.enhanced\_pbp.live.end\_of\_period,  
33 40  
pbpstats.resources.boxscore.live\_boxscore\_item, pbpstats.resources.enhanced\_pbp.live.enhanced\_pbp\_  
33 40  
pbpstats.resources.boxscore.stats\_nba\_boxscore\_item, pbpstats.resources.enhanced\_pbp.live.enhanced\_pbp\_  
33 40  
pbpstats.resources.enhanced\_pbp.data\_nba.ejection, pbpstats.resources.enhanced\_pbp.live.field\_goal,  
34 41

pbpstats.resources.enhanced\_pbp.live.foul, 41  
pbpstats.resources.enhanced\_pbp.live.free\_throws, 42  
pbpstats.resources.enhanced\_pbp.live.jump\_ball, 43  
pbpstats.resources.enhanced\_pbp.live.rebound, 43  
pbpstats.resources.enhanced\_pbp.live.replay, 43  
pbpstats.resources.enhanced\_pbp.live.start\_of\_period, 43  
pbpstats.resources.enhanced\_pbp.live.substitution, 44  
pbpstats.resources.enhanced\_pbp.live.timeout, 44  
pbpstats.resources.enhanced\_pbp.live.turnover, 44  
pbpstats.resources.enhanced\_pbp.live.violation, 45  
pbpstats.resources.enhanced\_pbp.rebound, 56  
pbpstats.resources.enhanced\_pbp.replay, 57  
pbpstats.resources.enhanced\_pbp.start\_of\_period, 57  
pbpstats.resources.enhanced\_pbp.stats\_nba.end\_of\_period, 46  
pbpstats.resources.enhanced\_pbp.stats\_nba.enhanced\_pbp\_factory, 46  
pbpstats.resources.enhanced\_pbp.stats\_nba.enhanced\_pbp\_item, 46  
pbpstats.resources.enhanced\_pbp.stats\_nba.field\_goal, 47  
pbpstats.resources.enhanced\_pbp.stats\_nba.foul, 47  
pbpstats.resources.enhanced\_pbp.stats\_nba.free\_throw, 48  
pbpstats.resources.enhanced\_pbp.stats\_nba.jump\_ball, 49  
pbpstats.resources.enhanced\_pbp.stats\_nba.rebound, 49  
pbpstats.resources.enhanced\_pbp.stats\_nba.replay, 49  
pbpstats.resources.enhanced\_pbp.stats\_nba.start\_of\_period, 50  
pbpstats.resources.enhanced\_pbp.stats\_nba.substitution, 50  
pbpstats.resources.enhanced\_pbp.stats\_nba.timeout, 50  
pbpstats.resources.enhanced\_pbp.stats\_nba.turnover, 50

pbpstats.resources.enhanced\_pbp.stats\_nba.violation, 51  
pbpstats.resources.enhanced\_pbp.substitution, 58  
pbpstats.resources.enhanced\_pbp.timeout, 58  
pbpstats.resources.enhanced\_pbp.turnover, 59  
pbpstats.resources.enhanced\_pbp.violation, 59  
pbpstats.resources.games.data\_nba\_game\_item, 60  
pbpstats.resources.games.games, 60  
pbpstats.resources.games.stats\_nba\_game\_item, 60  
pbpstats.resources.pbp.data\_nba\_pbp\_item, 60  
pbpstats.resources.pbp.live\_pbp\_item, 61  
pbpstats.resources.pbp.pbp, 61  
pbpstats.resources.pbp.stats\_nba\_pbp\_item, 61  
pbpstats.resources.possessions.possession, 62  
pbpstats.resources.possessions.possessions, 63  
pbpstats.resources.shots.shots, 63  
pbpstats.resources.shots.stats\_nba\_shot, 63





## A

counts\_as\_personal\_foul (pbp-  
 action\_type (pbpstats.resources.enhanced\_pbp.live.ejection.LiveEjection  
 attribute), 39 stats.resources.enhanced\_pbp.foul.Foul  
 attribute), 54  
 counts\_towards\_penalty (pbp-  
 action\_type (pbpstats.resources.enhanced\_pbp.live.end\_of\_period.LiveEndOfPeriod  
 attribute), 40 stats.resources.enhanced\_pbp.foul.Foul  
 attribute), 54  
 current\_players (pbp-  
 action\_type (pbpstats.resources.enhanced\_pbp.live.field\_goal.LiveFieldGoal  
 attribute), 41 stats.resources.enhanced\_pbp.enhanced\_pbp\_item.EnhancedPbpItem  
 attribute), 52  
 current\_players (pbp-  
 action\_type (pbpstats.resources.enhanced\_pbp.live.free\_throw.LiveFreeThrow  
 attribute), 42 stats.resources.enhanced\_pbp.live.substitution.LiveSubstitution  
 attribute), 44  
 current\_players (pbp-  
 action\_type (pbpstats.resources.enhanced\_pbp.live.jump\_ball.LiveJumpBall  
 attribute), 43 stats.resources.enhanced\_pbp.start\_of\_period.StartOfPeriod  
 attribute), 58  
 current\_players (pbp-  
 action\_type (pbpstats.resources.enhanced\_pbp.live.rebound.LiveRebound  
 attribute), 43 stats.resources.enhanced\_pbp.substitution.Substitution  
 attribute), 58  
 current\_players (pbp-  
 action\_type (pbpstats.resources.enhanced\_pbp.live.replay.LiveReplay  
 attribute), 43 stats.resources.enhanced\_pbp.substitution.Substitution  
 attribute), 58  
 current\_players (pbp-  
 action\_type (pbpstats.resources.enhanced\_pbp.live.start\_of\_period.LiveStartOfPeriod  
 attribute), 43 stats.resources.enhanced\_pbp.substitution.Substitution  
 attribute), 58  
 data (pbpstats.data\_loader.data\_nba.boxscore.loader.DataNbaBoxscoreLoader  
 attribute), 10  
 data (pbpstats.data\_loader.data\_nba.pbp.loader.DataNbaPbpLoader  
 attribute), 12  
 data (pbpstats.data\_loader.data\_nba.schedule.loader.DataNbaScheduleLoader  
 attribute), 14  
 data (pbpstats.data\_loader.live.boxscore.loader.LiveBoxscoreLoader  
 attribute), 16  
 data (pbpstats.data\_loader.live.pbp.loader.LivePbpLoader  
 attribute), 17  
 data (pbpstats.data\_loader.stats\_nba.base.StatsNbaLoaderBase  
 attribute), 20  
 data (pbpstats.resources.boxscore.boxscore.Boxscore  
 attribute), 32  
 data (pbpstats.resources.boxscore.data\_nba\_boxscore\_item.DataNbaBoxscoreItem  
 attribute), 33  
 data (pbpstats.resources.boxscore.live\_boxscore\_item.LiveBoxscoreItem  
 attribute), 33  
 data (pbpstats.resources.boxscore.stats\_nba\_boxscore\_item.StatsNbaBoxscoreItem  
 attribute), 33

## B

base\_stats (pbpstats.resources.enhanced\_pbp.enhanced\_pbp\_item.EnhancedPbpItem  
 attribute), 52  
 Boxscore (class in pbpstats.resources.boxscore.boxscore), 32

## C

Client (class in pbpstats.client), 64  
 count\_as\_possession (pbp-  
 stats.resources.enhanced\_pbp.enhanced\_pbp\_item.EnhancedPbpItem  
 attribute), 52

data (pbpstats.resources.enhanced\_pbp.data\_nba.enhanced\_pbp\_item.DataNbaEnhancedPbpItem  
 attribute), 35 data\_provider (pbp-  
 data (pbpstats.resources.enhanced\_pbp.enhanced\_pbp.EnhancedPbpItem stats.data\_loader.stats\_nba.boxscore.loader.StatsNbaBoxscoreLo  
 attribute), 52 attribute), 21  
 data (pbpstats.resources.enhanced\_pbp.live.enhanced\_pbp\_item.LiveEnhancedPbpItem (pbp-  
 attribute), 40 stats.data\_loader.stats\_nba.enhanced\_pbp.loader.StatsNbaEnhancedPbpItem  
 data (pbpstats.resources.enhanced\_pbp.stats\_nba.enhanced\_pbp\_item.StatsNbaEnhancedPbpItem  
 attribute), 46 data\_provider (pbp-  
 data (pbpstats.resources.games.data\_nba\_game\_item.DataNbaGameItem stats.data\_loader.stats\_nba.league\_game\_log.loader.StatsNbaLeagueGameLog  
 attribute), 60 attribute), 25  
 data (pbpstats.resources.games.games.Games attribute), 60 data\_provider (pbp-  
 attribute), 60 stats.data\_loader.stats\_nba.pbp.loader.StatsNbaPbpLoader  
 data (pbpstats.resources.games.stats\_nba\_game\_item.StatsNbaGameItem attribute), 23  
 attribute), 60 data\_provider (pbp-  
 data (pbpstats.resources.pbp.data\_nba\_pbp\_item.DataNbaPbpItem stats.data\_loader.stats\_nba.possessions.loader.StatsNbaPossessions  
 attribute), 61 attribute), 24  
 data (pbpstats.resources.pbp.live\_pbp\_item.LivePbpItem data\_provider (pbp-  
 attribute), 61 stats.data\_loader.stats\_nba.scoreboard.loader.StatsNbaScoreboard  
 data (pbpstats.resources.pbp.pbp.Pbp attribute), 61 attribute), 26  
 data (pbpstats.resources.pbp.stats\_nba\_pbp\_item.StatsNbaPbpItem data\_provider (pbp-  
 attribute), 61 stats.data\_loader.stats\_nba.shots.loader.StatsNbaShotsLoader  
 data (pbpstats.resources.possessions.possession.Possession attribute), 27  
 attribute), 62 data\_provider (pbp-  
 data (pbpstats.resources.possessions.possessions.Possessions stats.data\_loader.stats\_nba.summary.loader.StatsNbaSummaryLoader  
 attribute), 63 attribute), 28  
 data (pbpstats.resources.shots.shots.Shots attribute), 63 DataEjection (class in pbp-  
 data (pbpstats.resources.shots.stats\_nba\_shot.StatsNbaShot stats.resources.enhanced\_pbp.data\_nba.ejection),  
 attribute), 63 34  
 data\_provider (pbp- DataEndOfPeriod (class in pbp-  
 stats.data\_loader.data\_nba.boxscore.loader.DataNbaBoxscoreLoader stats.resources.enhanced\_pbp.data\_nba.end\_of\_period),  
 attribute), 10 34  
 data\_provider (pbp- DataEnhancedPbpItem (class in pbp-  
 stats.data\_loader.data\_nba.enhanced\_pbp.loader.DataNbaEnhancedPbpLoader stats.resources.enhanced\_pbp.data\_nba.enhanced\_pbp\_item),  
 attribute), 11 34  
 data\_provider (pbp- DataFieldGoal (class in pbp-  
 stats.data\_loader.data\_nba.pbp.loader.DataNbaPbpLoader stats.resources.enhanced\_pbp.data\_nba.field\_goal),  
 attribute), 12 35  
 data\_provider (pbp- DataFoul (class in pbp-  
 stats.data\_loader.data\_nba.possessions.loader.DataNbaPossessionsLoader stats.resources.enhanced\_pbp.data\_nba.foul),  
 attribute), 13 35  
 data\_provider (pbp- DataFreeThrow (class in pbp-  
 stats.data\_loader.data\_nba.schedule.loader.DataNbaScheduleLoader stats.resources.enhanced\_pbp.data\_nba.free\_throw),  
 attribute), 14 36  
 data\_provider (pbp- DataJumpBall (class in pbp-  
 stats.data\_loader.live.boxscore.loader.LiveBoxscoreLoader stats.resources.enhanced\_pbp.data\_nba.jump\_ball),  
 attribute), 16 37  
 data\_provider (pbp- DataLoaderFactory (class in pbp-  
 stats.data\_loader.live.enhanced\_pbp.loader.LiveEnhancedPbpLoader stats.data\_loader.factory), 30  
 attribute), 17 DataNbaBoxscoreFileLoader (class in pbp-  
 data\_provider (pbp- stats.data\_loader.data\_nba.boxscore.file),  
 stats.data\_loader.live.pbp.loader.LivePbpLoader 10  
 attribute), 18 DataNbaBoxscoreItem (class in pbp-  
 data\_provider (pbp- stats.resources.boxscore.data\_nba\_boxscore\_item),  
 stats.data\_loader.live.possessions.loader.LivePossessionLoader 11

DataNbaBoxscoreLoader (class in <i>pbpstats.data_loader.data_nba.boxscore.loader</i> ), 9	15	DataRebound (class in <i>pbpstats.resources.enhanced_pbp.data_nba.rebound</i> ), 37
DataNbaBoxscoreWebLoader (class in <i>pbpstats.data_loader.data_nba.boxscore.web</i> ), 10	37	DataReplay (class in <i>pbpstats.resources.enhanced_pbp.data_nba.replay</i> ), 37
DataNbaEnhancedPbpFactory (class in <i>pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp_factory</i> ), 34	37	EndOfPeriod (class in <i>pbpstats.resources.enhanced_pbp.data_nba.start_of_period</i> ), 38
DataNbaEnhancedPbpFileLoader (class in <i>pbpstats.data_loader.data_nba.enhanced_pbp.file</i> ), 11	38	DataSubstitution (class in <i>pbpstats.resources.enhanced_pbp.data_nba.substitution</i> ), 38
DataNbaEnhancedPbpLoader (class in <i>pbpstats.data_loader.data_nba.enhanced_pbp.loader</i> ), 11	38	DataTimeout (class in <i>pbpstats.resources.enhanced_pbp.data_nba.timeout</i> ), 38
DataNbaEnhancedPbpWebLoader (class in <i>pbpstats.data_loader.data_nba.enhanced_pbp.web</i> ), 11	38	DataTurnover (class in <i>pbpstats.resources.enhanced_pbp.data_nba.turnover</i> ), 38
DataNbaFileLoader (class in <i>pbpstats.data_loader.data_nba.file_loader</i> ), 15	38	DataViolation (class in <i>pbpstats.resources.enhanced_pbp.data_nba.violation</i> ), 39
DataNbaGameItem (class in <i>pbpstats.resources.games.data_nba_game_item</i> ), 60	39	Day (class in <i>pbpstats.objects.day</i> ), 31
DataNbaPbpFileLoader (class in <i>pbpstats.data_loader.data_nba.pbp.file</i> ), 12	dedupe_events_row_set () (pbpstats.data_loader.stats_nba.base.StatsNbaLoaderBase static method), 20	
DataNbaPbpItem (class in <i>pbpstats.resources.pbp.data_nba_pbp_item</i> ), 60	distance ( <i>pbpstats.resources.enhanced_pbp.field_goal.FieldGoal</i> attribute), 53	
DataNbaPbpLoader (class in <i>pbpstats.data_loader.data_nba.pbp.loader</i> ), 12	<b>E</b>	
DataNbaPbpWebLoader (class in <i>pbpstats.data_loader.data_nba.pbp.web</i> ), 12	Ejection (class in <i>pbpstats.resources.enhanced_pbp.ejection</i> ), 51	
DataNbaPossessionFileLoader (class in <i>pbpstats.data_loader.data_nba.possessions.file</i> ), 13	end_time ( <i>pbpstats.resources.possessions.possession.Possession</i> attribute), 62	
DataNbaPossessionLoader (class in <i>pbpstats.data_loader.data_nba.possessions.loader</i> ), 13	EndOfPeriod (class in <i>pbpstats.resources.enhanced_pbp.end_of_period</i> ), 52	
DataNbaPossessionWebLoader (class in <i>pbpstats.data_loader.data_nba.possessions.web</i> ), 13	EnhancedPbp (class in <i>pbpstats.resources.enhanced_pbp.enhanced_pbp</i> ), 52	
DataNbaScheduleFileLoader (class in <i>pbpstats.data_loader.data_nba.schedule.file</i> ), 14	EnhancedPbpItem (class in <i>pbpstats.resources.enhanced_pbp.enhanced_pbp_item</i> ), 52	
DataNbaScheduleLoader (class in <i>pbpstats.data_loader.data_nba.schedule.loader</i> ), 14	event_for_efficiency_stats ( <i>pbpstats.resources.enhanced_pbp.free_throw.FreeThrow</i> attribute), 55	
DataNbaScheduleWebLoader (class in <i>pbpstats.data_loader.data_nba.schedule.web</i> ), 14	event_stats ( <i>pbpstats.resources.enhanced_pbp.data_nba.enhanced_pbp</i> attribute), 35	
DataNbaWebLoader (class in <i>pbpstats.data_loader.data_nba.web_loader</i> ),	event_stats ( <i>pbpstats.resources.enhanced_pbp.ejection.Ejection</i> attribute), 51	
	event_stats ( <i>pbpstats.resources.enhanced_pbp.end_of_period.EndOfPeriod</i> attribute), 52	
	event_stats ( <i>pbpstats.resources.enhanced_pbp.enhanced_pbp_item</i> attribute), 53	

event\_stats (pbpstats.resources.enhanced\_pbp.field\_goal.FieldGoal (attribute), 53

event\_stats (pbpstats.resources.enhanced\_pbp.foul.FoulEvent\_type (pbpstats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul (attribute), 54

event\_stats (pbpstats.resources.enhanced\_pbp.free\_throw.FreeThrowType (pbpstats.resources.enhanced\_pbp.stats\_nba.free\_throw.StatsFreeThrow (attribute), 55

event\_stats (pbpstats.resources.enhanced\_pbp.jump\_ball.JumpBallType (pbpstats.resources.enhanced\_pbp.stats\_nba.jump\_ball.StatsJumpBall (attribute), 56

event\_stats (pbpstats.resources.enhanced\_pbp.live.enhanced\_pbp.LivePbpEnhancedPbp (attribute), 40

event\_stats (pbpstats.resources.enhanced\_pbp.rebound.ReboundType (pbpstats.resources.enhanced\_pbp.stats\_nba.replay.StatsRebound (attribute), 56

event\_stats (pbpstats.resources.enhanced\_pbp.replay.ReplayType (pbpstats.resources.enhanced\_pbp.stats\_nba.start\_of\_period.StatsStartOfPeriod (attribute), 57

event\_stats (pbpstats.resources.enhanced\_pbp.start\_of\_period.StartOfPeriodType (pbpstats.resources.enhanced\_pbp.stats\_nba.substitution.StatsSubstitution (attribute), 58

event\_stats (pbpstats.resources.enhanced\_pbp.substitution.SubstitutionType (pbpstats.resources.enhanced\_pbp.stats\_nba.timeout.StatsTimeout (attribute), 58

event\_stats (pbpstats.resources.enhanced\_pbp.timeout.TimeoutType (pbpstats.resources.enhanced\_pbp.stats\_nba.turnover.StatsTurnover (attribute), 58

event\_stats (pbpstats.resources.enhanced\_pbp.turnover.TurnoverType (pbpstats.resources.enhanced\_pbp.stats\_nba.violation.StatsViolation (attribute), 59

event\_stats (pbpstats.resources.enhanced\_pbp.violation.ViolationOrderError, 56

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.ejection.DataEjection (attribute), 34

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.end\_of\_period.EndOfPeriod (attribute), 34

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.field\_goal.FieldGoal (attribute), 35

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.foul.DataFoul (attribute), 35

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.free\_throw.DataFreeThrow (attribute), 36

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.jump\_ball.DataJumpBall (attribute), 37

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.rebound.DataRebound (attribute), 37

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.replay.DataReplay (attribute), 37

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.start\_of\_period.StartOfPeriod (attribute), 38

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.substitution.DataSubstitution (attribute), 38

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.timeout.DataTimeout (attribute), 38

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.turnover.DataTurnover (attribute), 38

event\_type (pbpstats.resources.enhanced\_pbp.data\_nba.violation.DataViolation (attribute), 39

event\_type (pbpstats.resources.enhanced\_pbp.stats\_nba.ejection.StatsEjection (attribute), 45

event\_type (pbpstats.resources.enhanced\_pbp.stats\_nba.end\_of\_period.StatsEndOfPeriod (attribute), 46

G

Game (class in pbpstats.objects.game), 31

Games (class in pbpstats.resources.games.games), 60

get\_all\_events\_at\_current\_time() (pbp- stats.resources.enhanced\_pbp.enhanced\_pbp\_item.EnhancedPbpItem method), 53  
 get\_data\_loader() (pbp- stats.data\_loader.factory.DataLoaderFactory method), 30  
 get\_event\_class() (pbp- stats.resources.enhanced\_pbp.data\_nba.enhanced\_pbp\_factory.DataNbaEnhancedPbpFactory method), 34  
 get\_event\_class() (pbp- stats.resources.enhanced\_pbp.live.enhanced\_pbp\_factory.LiveEnhancedPbpFactory method), 40  
 get\_event\_class() (pbp- stats.resources.enhanced\_pbp.stats\_nba.enhanced\_pbp\_factory.StatsNbaEnhancedPbpFactory method), 46  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.data\_nba.enhanced\_pbp\_item.DataNbaEnhancedPbpItem method), 35  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.enhanced\_pbp\_item.EnhancedPbpItem method), 53  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.live.enhanced\_pbp\_item.LiveEnhancedPbpItem method), 41  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.start\_of\_period.StartOfPeriod method), 58  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.stats\_nba.enhanced\_pbp\_item.StatsNbaEnhancedPbpItem method), 46  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.stats\_nba.field\_goal.StatsFieldGoal method), 47  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.stats\_nba.free\_throw.StatsFreeThrow method), 48  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.stats\_nba.jump\_ball.StatsJumpBall method), 49  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.stats\_nba.rebound.StatsRebound method), 49  
 get\_offense\_team\_id() (pbp- stats.resources.enhanced\_pbp.stats\_nba.turnover.StatsTurnover method), 51  
 get\_period\_starters() (pbp- stats.resources.enhanced\_pbp.data\_nba.start\_of\_period.DataStartOfPeriod method), 38  
 get\_period\_starters() (pbp- stats.resources.enhanced\_pbp.live.start\_of\_period.LiveStartOfPeriod method), 44  
 get\_period\_starters() (pbp- stats.resources.enhanced\_pbp.start\_of\_period.StartOfPeriod class method), 58  
 get\_period\_starters() (pbp- stats.resources.enhanced\_pbp.stats\_nba.start\_of\_period.StatsStartOfPeriod method), 50  
 get\_team\_ids() (pbp- stats.resources.possessions.possession.Possession method), 62  
 get\_team\_starting\_with\_ball() (pbp- stats.data\_nba.enhanced\_pbp\_factory.DataNbaEnhancedPbpFactory method), 58  
 incoming\_player\_id (pbp- stats.resources.enhanced\_pbp.data\_nba.substitution.DataSubstitution attribute), 46  
 incoming\_player\_id (pbp- stats.resources.enhanced\_pbp.live.substitution.LiveSubstitution attribute), 46  
 incoming\_player\_id (pbp- stats.resources.enhanced\_pbp.stats\_nba.substitution.StatsSubstitution attribute), 50  
 incoming\_player\_id (pbp- stats.resources.enhanced\_pbp.substitution.Substitution attribute), 46  
 InvalidNumberOfStartersException, 57  
 is\_3\_second\_violation (pbp- stats.resources.enhanced\_pbp.data\_nba.turnover.DataTurnover attribute), 39  
 is\_3\_second\_violation (pbp- stats.resources.enhanced\_pbp.live.turnover.LiveTurnover attribute), 44  
 is\_3\_second\_violation (pbp- stats.resources.enhanced\_pbp.stats\_nba.turnover.StatsTurnover attribute), 51  
 is\_3\_second\_violation() (pbp- stats.resources.enhanced\_pbp.turnover.Turnover class method), 59  
 is\_and1 (pbpstats.resources.enhanced\_pbp.field\_goal.FieldGoal attribute), 53  
 is\_assisted (pbpstats.resources.enhanced\_pbp.field\_goal.FieldGoal attribute), 53  
 is\_away\_from\_play\_foul (pbp- stats.resources.enhanced\_pbp.data\_nba.foul.DataFoul attribute), 35  
 is\_away\_from\_play\_foul (pbp- stats.resources.enhanced\_pbp.foul.Foul attribute), 54  
 is\_away\_from\_play\_foul (pbp- stats.resources.enhanced\_pbp.live.foul.LiveFoul attribute), 41  
 is\_away\_from\_play\_foul (pbp- stats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul attribute), 47  
 is\_away\_from\_play\_ft (pbp- stats.resources.enhanced\_pbp.free\_throw.FreeThrow



attribute), 55  
 is\_away\_from\_play\_ft (pbp- stats.resources.enhanced\_pbp.live.free\_throw.LiveFreeThrow attribute), 42  
 is\_bad\_pass (pbpstats.resources.enhanced\_pbp.data\_nba.turnover.DataTurnover attribute), 39  
 is\_bad\_pass (pbpstats.resources.enhanced\_pbp.live.turnover.LiveTurnover attribute), 44  
 is\_bad\_pass (pbpstats.resources.enhanced\_pbp.stats\_nba.turnover.StatsTurnover attribute), 51  
 is\_bad\_pass () (pbp- stats.resources.enhanced\_pbp.turnover.Turnover class method), 59  
 is\_bad\_pass\_out\_of\_bounds (pbp- stats.resources.enhanced\_pbp.data\_nba.turnover.DataTurnover attribute), 39  
 is\_bad\_pass\_out\_of\_bounds (pbp- stats.resources.enhanced\_pbp.live.turnover.LiveTurnover attribute), 45  
 is\_bad\_pass\_out\_of\_bounds (pbp- stats.resources.enhanced\_pbp.stats\_nba.turnover.StatsTurnover attribute), 51  
 is\_bad\_pass\_out\_of\_bounds () (pbp- stats.resources.enhanced\_pbp.turnover.Turnover class method), 59  
 is\_blocked (pbpstats.resources.enhanced\_pbp.field\_goal.FieldGoal attribute), 53  
 is\_buzzer\_beater\_placeholder (pbp- stats.resources.enhanced\_pbp.rebound.Rebound attribute), 56  
 is\_buzzer\_beater\_rebound\_at\_shot\_time (pbpstats.resources.enhanced\_pbp.rebound.Rebound attribute), 56  
 is\_charge (pbpstats.resources.enhanced\_pbp.data\_nba.foul.DataFoul attribute), 35  
 is\_charge (pbpstats.resources.enhanced\_pbp.foul.Foul attribute), 54  
 is\_charge (pbpstats.resources.enhanced\_pbp.live.foul.LiveFoul attribute), 41  
 is\_charge (pbpstats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul attribute), 47  
 is\_clear\_path\_foul (pbp- stats.resources.enhanced\_pbp.data\_nba.foul.DataFoul attribute), 35  
 is\_clear\_path\_foul (pbp- stats.resources.enhanced\_pbp.foul.Foul attribute), 54  
 is\_clear\_path\_foul (pbp- stats.resources.enhanced\_pbp.live.foul.LiveFoul attribute), 41  
 is\_clear\_path\_foul (pbp- stats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul attribute), 47  
 is\_corner\_3 (pbpstats.resources.enhanced\_pbp.field\_goal.FieldGoal attribute), 53  
 is\_defensive\_3\_seconds (pbp- stats.resources.enhanced\_pbp.data\_nba.foul.DataFoul attribute), 36  
 is\_double\_foul (pbp- stats.resources.enhanced\_pbp.foul.Foul attribute), 54  
 is\_defensive\_3\_seconds (pbp- stats.resources.enhanced\_pbp.live.foul.LiveFoul attribute), 41  
 is\_defensive\_3\_seconds (pbp- stats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul attribute), 47  
 is\_delay\_of\_game (pbp- stats.resources.enhanced\_pbp.data\_nba.foul.DataFoul attribute), 36  
 is\_delay\_of\_game (pbp- stats.resources.enhanced\_pbp.data\_nba.violation.DataViolation attribute), 39  
 is\_delay\_of\_game (pbp- stats.resources.enhanced\_pbp.foul.Foul attribute), 54  
 is\_delay\_of\_game (pbp- stats.resources.enhanced\_pbp.live.foul.LiveFoul attribute), 41  
 is\_delay\_of\_game (pbp- stats.resources.enhanced\_pbp.live.violation.LiveViolation attribute), 45  
 is\_delay\_of\_game (pbp- stats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul attribute), 47  
 is\_delay\_of\_game (pbp- stats.resources.enhanced\_pbp.stats\_nba.violation.StatsViolation attribute), 51  
 is\_double\_foul (pbp- stats.resources.enhanced\_pbp.foul.Foul attribute), 54  
 is\_double\_foul (pbp- stats.resources.enhanced\_pbp.data\_nba.foul.DataFoul attribute), 36  
 is\_double\_foul (pbp- stats.resources.enhanced\_pbp.foul.Foul attribute), 54  
 is\_double\_foul (pbp- stats.resources.enhanced\_pbp.live.foul.LiveFoul attribute), 41  
 is\_double\_foul (pbp- stats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul attribute), 47  
 is\_double\_lane\_violation (pbp- stats.resources.enhanced\_pbp.data\_nba.violation.DataViolation attribute), 39  
 is\_double\_lane\_violation (pbp- stats.resources.enhanced\_pbp.live.violation.LiveViolation attribute), 39

attribute), 45  
 is\_double\_lane\_violation (pbp- stats.resources.enhanced\_pbp.stats\_nba.violation.StatsViolation attribute), 36  
 is\_double\_lane\_violation (pbp- stats.resources.enhanced\_pbp.live.free\_throw.LiveFreeThrow attribute), 42  
 is\_double\_lane\_violation() (pbp- stats.resources.enhanced\_pbp.violation.Violation is\_flagrant\_ft attribute), 48  
 is\_double\_technical (pbp- stats.resources.enhanced\_pbp.data\_nba.foul.DataFoul ft\_1\_of\_1 attribute), 36  
 is\_double\_technical (pbp- stats.resources.enhanced\_pbp.foul.Foul at is\_ft\_1\_of\_1 attribute), 36  
 is\_double\_technical (pbp- stats.resources.enhanced\_pbp.free\_throw.FreeThrow attribute), 55  
 is\_double\_technical (pbp- stats.resources.enhanced\_pbp.live.foul.LiveFoul is\_ft\_1\_of\_1 attribute), 41  
 is\_double\_technical (pbp- stats.resources.enhanced\_pbp.live.free\_throw.LiveFreeThrow attribute), 42  
 is\_double\_technical (pbp- stats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul ft\_1\_of\_1 attribute), 47  
 is\_double\_technical (pbp- stats.resources.enhanced\_pbp.stats\_nba.free\_throw.StatsFreeThrow attribute), 48  
 is\_end\_ft (pbpstats.resources.enhanced\_pbp.free\_throw.FreeThrow attribute), 55  
 is\_end\_ft (pbpstats.resources.enhanced\_pbp.free\_throw.FreeThrow is\_ft\_1\_of\_2 attribute), 36  
 is\_final (pbpstats.resources.games.data\_nba\_game\_item.DataNbaGameItem stats.resources.enhanced\_pbp.data\_nba.free\_throw.DataFreeThrow attribute), 60  
 is\_final (pbpstats.resources.games.stats\_nba\_game\_item.StatsNbaGameItem stats.resources.enhanced\_pbp.free\_throw.FreeThrow attribute), 60  
 is\_first\_ft (pbpstats.resources.enhanced\_pbp.free\_throw.FreeThrow attribute), 55  
 is\_first\_ft (pbpstats.resources.enhanced\_pbp.free\_throw.FreeThrow is\_ft\_1\_of\_2 attribute), 55  
 is\_flagrant (pbpstats.resources.enhanced\_pbp.foul.Foul stats.resources.enhanced\_pbp.live.free\_throw.LiveFreeThrow attribute), 54  
 is\_flagrant (pbpstats.resources.enhanced\_pbp.foul.Foul stats.resources.enhanced\_pbp.live.free\_throw.LiveFreeThrow attribute), 42  
 is\_flagrant1 (pbp- is\_ft\_1\_of\_2 attribute), 36  
 is\_flagrant1 (pbp- stats.resources.enhanced\_pbp.data\_nba.foul.DataFoul stats.resources.enhanced\_pbp.stats\_nba.free\_throw.StatsFreeThrow attribute), 48  
 is\_flagrant1 (pbp- is\_ft\_1\_of\_3 attribute), 36  
 is\_flagrant1 (pbp- stats.resources.enhanced\_pbp.foul.Foul at stats.resources.enhanced\_pbp.data\_nba.free\_throw.DataFreeThrow attribute), 54  
 is\_flagrant1 (pbp- is\_ft\_1\_of\_3 attribute), 36  
 is\_flagrant1 (pbp- stats.resources.enhanced\_pbp.live.foul.LiveFoul stats.resources.enhanced\_pbp.free\_throw.FreeThrow attribute), 41  
 is\_flagrant1 (pbp- is\_ft\_1\_of\_3 attribute), 55  
 is\_flagrant1 (pbp- stats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul stats.resources.enhanced\_pbp.live.free\_throw.LiveFreeThrow attribute), 42  
 is\_flagrant1 (pbp- is\_ft\_1\_of\_3 attribute), 42  
 is\_flagrant2 (pbp- is\_ft\_1\_of\_3 attribute), 36  
 is\_flagrant2 (pbp- stats.resources.enhanced\_pbp.data\_nba.foul.DataFoul stats.resources.enhanced\_pbp.stats\_nba.free\_throw.StatsFreeThrow attribute), 48  
 is\_flagrant2 (pbp- is\_ft\_1pt (pbpstats.resources.enhanced\_pbp.data\_nba.free\_throw.DataFreeThrow attribute), 36  
 is\_flagrant2 (pbp- is\_ft\_1pt (pbpstats.resources.enhanced\_pbp.free\_throw.FreeThrow attribute), 55  
 is\_flagrant2 (pbp- stats.resources.enhanced\_pbp.live.foul.LiveFoul is\_ft\_1pt (pbpstats.resources.enhanced\_pbp.live.free\_throw.LiveFreeThrow attribute), 42  
 is\_flagrant2 (pbp- is\_ft\_1pt (pbpstats.resources.enhanced\_pbp.stats\_nba.free\_throw.StatsFreeThrow attribute), 48  
 is\_flagrant2 (pbp- is\_ft\_1pt (pbpstats.resources.enhanced\_pbp.stats\_nba.free\_throw.StatsFreeThrow attribute), 47  
 is\_flagrant2 (pbp- is\_ft\_2\_of\_2 attribute), 36  
 is\_flagrant\_ft (pbp- stats.resources.enhanced\_pbp.data\_nba.free\_throw.DataFreeThrow attribute), 36

attribute), 36		stats.resources.enhanced_pbp.live.violation.LiveViolation
is_ft_2_of_2	(pbp- stats.resources.enhanced_pbp.free_throw.FreeThrow	attribute), 45
attribute), 55		is_goaltend_violation (pbp- stats.resources.enhanced_pbp.stats_nba.violation.StatsViolation
is_ft_2_of_2	(pbp- attribute), 51	
stats.resources.enhanced_pbp.live.free_throw.LiveFreeThrow	is_goaltend_violation () (pbp- stats.resources.enhanced_pbp.violation.Violation	
attribute), 42		
is_ft_2_of_2	(pbp- class method), 59	
stats.resources.enhanced_pbp.stats_nba.free_throw.StatsFreeThrow	stats.resources.enhanced_pbp.field_goal.FieldGoal	
attribute), 48		attribute), 53
is_ft_2_of_3	(pbp- is_inbound_foul (pbp- stats.resources.enhanced_pbp.data_nba.free_throw.DataFreeThrow	
attribute), 36		stats.resources.enhanced_pbp.data_nba.foul.DataFoul
is_ft_2_of_3	(pbp- is_inbound_foul (pbp- stats.resources.enhanced_pbp.free_throw.FreeThrow	
attribute), 55		stats.resources.enhanced_pbp.foul.Foul attribute), 54
is_ft_2_of_3	(pbp- is_inbound_foul (pbp- stats.resources.enhanced_pbp.live.free_throw.LiveFreeThrow	
attribute), 42		stats.resources.enhanced_pbp.live.foul.LiveFoul
is_ft_2_of_3	(pbp- is_inbound_foul (pbp- stats.resources.enhanced_pbp.stats_nba.free_throw.StatsFreeThrow	
attribute), 48		stats.resources.enhanced_pbp.stats_nba.foul.StatsFoul
is_ft_2pt (pbpstats.resources.enhanced_pbp.data_nba.free_throw.DataFreeThrow		attribute), 47
attribute), 36		stats.resources.enhanced_pbp.free_throw.FreeThrow
is_ft_2pt (pbpstats.resources.enhanced_pbp.free_throw.FreeThrow		attribute), 56
attribute), 55		is_jumpball_violation (pbp- stats.resources.enhanced_pbp.data_nba.violation.DataViolation
is_ft_2pt (pbpstats.resources.enhanced_pbp.live.free_throw.LiveFreeThrow		attribute), 39
attribute), 42		is_jumpball_violation (pbp- stats.resources.enhanced_pbp.live.violation.LiveViolation
is_ft_2pt (pbpstats.resources.enhanced_pbp.stats_nba.free_throw.StatsFreeThrow		attribute), 45
attribute), 48		stats.resources.enhanced_pbp.data_nba.violation.DataViolation
is_ft_3_of_3	(pbp- attribute), 51	
stats.resources.enhanced_pbp.data_nba.free_throw.DataFreeThrow		stats.resources.enhanced_pbp.stats_nba.violation.StatsViolation
attribute), 36		attribute), 51
is_ft_3_of_3	(pbp- is_jumpball_violation () (pbp- stats.resources.enhanced_pbp.violation.Violation	
attribute), 55		
is_ft_3_of_3	(pbp- class method), 59	
stats.resources.enhanced_pbp.live.free_throw.LiveFreeThrow	is_kicked_ball (pbp- stats.resources.enhanced_pbp.data_nba.turnover.DataTurnover	
attribute), 42		attribute), 39
is_ft_3_of_3	(pbp- attribute), 39	
stats.resources.enhanced_pbp.stats_nba.free_throw.StatsFreeThrow		stats.resources.enhanced_pbp.live.turnover.LiveTurnover
attribute), 48		
is_ft_3pt (pbpstats.resources.enhanced_pbp.data_nba.free_throw.DataFreeThrow		is_kicked_ball_violation (pbp- stats.resources.enhanced_pbp.stats_nba.turnover.StatsTurnover
attribute), 36		attribute), 51
is_ft_3pt (pbpstats.resources.enhanced_pbp.free_throw.FreeThrow		is_kicked_ball_violation () (pbp- stats.resources.enhanced_pbp.turnover.Turnover
attribute), 56		
is_ft_3pt (pbpstats.resources.enhanced_pbp.live.free_throw.LiveFreeThrow		attribute), 59
attribute), 42		is_kicked_ball_violation (pbp- stats.resources.enhanced_pbp.data_nba.violation.DataViolation
is_ft_3pt (pbpstats.resources.enhanced_pbp.stats_nba.free_throw.StatsFreeThrow		attribute), 39
attribute), 48		is_kicked_ball_violation (pbp- stats.resources.enhanced_pbp.live.violation.LiveViolation
is_goaltend_violation	(pbp- stats.resources.enhanced_pbp.data_nba.violation.DataViolation	
attribute), 39		
is_goaltend_violation	(pbp- stats.resources.enhanced_pbp.live.violation.LiveViolation	



<b>Index</b>	<b>77</b>
--------------	-----------

<i>stats.resources.enhanced_pbp.live.foul.LiveFoul</i> <i>attribute</i> ), 41	<i>stats.resources.enhanced_pbp.stats_nba.foul.StatsFoul</i> <i>attribute</i> ), 48
<i>is_offensive_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.stats_nba.foul.StatsFoul</i> <i>attribute</i> ), 47	<i>is_placeholder</i> ( <i>pbp- stats.resources.enhanced_pbp.data_nba.rebound.DataRebound</i> <i>attribute</i> ), 37
<i>is_offensive_goaltending</i> ( <i>pbp- stats.resources.enhanced_pbp.data_nba.turnover.DataTurnover</i> <i>attribute</i> ), 39	<i>is_placeholder</i> ( <i>pbp- stats.resources.enhanced_pbp.live.rebound.LiveRebound</i> <i>attribute</i> ), 43
<i>is_offensive_goaltending</i> ( <i>pbp- stats.resources.enhanced_pbp.live.turnover.LiveTurnover</i> <i>attribute</i> ), 45	<i>is_placeholder</i> ( <i>pbp- stats.resources.enhanced_pbp.rebound.Rebound</i> <i>attribute</i> ), 57
<i>is_offensive_goaltending</i> ( <i>pbp- stats.resources.enhanced_pbp.stats_nba.turnover.StatsTurnover</i> <i>attribute</i> ), 51	<i>is_placeholder</i> ( <i>pbp- stats.resources.enhanced_pbp.stats_nba.rebound.StatsRebound</i> <i>attribute</i> ), 49
<i>is_offensive_goaltending()</i> ( <i>pbp- stats.resources.enhanced_pbp.turnover.Turnover</i> <i>class method</i> ), 59	<i>is_possession_ending_event</i> ( <i>pbp- stats.resources.enhanced_pbp.data_nba.enhanced_pbp_item.Data</i> <i>attribute</i> ), 35
<i>is_penalty_event()</i> ( <i>pbp- stats.resources.enhanced_pbp.enhanced_pbp_item.EnhancedPbpItem</i> <i>method</i> ), 53	<i>is_possession_ending_event</i> ( <i>pbp- stats.resources.enhanced_pbp.enhanced_pbp_item.EnhancedPbp</i> <i>attribute</i> ), 53
<i>is_personal_block_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.data_nba.foul.DataFoul</i> <i>attribute</i> ), 36	<i>is_possession_ending_event</i> ( <i>pbp- stats.resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem</i> <i>attribute</i> ), 41
<i>is_personal_block_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.foul.Foul</i> <i>at-</i> <i>tribute</i> ), 55	<i>is_possession_ending_event</i> ( <i>pbp- stats.resources.enhanced_pbp.stats_nba.enhanced_pbp_item.StatsEnhancedPbpItem</i> <i>attribute</i> ), 46
<i>is_personal_block_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.live.foul.LiveFoul</i> <i>attribute</i> ), 42	<i>is_putback</i> ( <i>pbpstats.resources.enhanced_pbp.field_goal.FieldGoal</i> <i>attribute</i> ), 54
<i>is_personal_block_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.stats_nba.foul.StatsFoul</i> <i>attribute</i> ), 47	<i>is_real_rebound</i> ( <i>pbp- stats.resources.enhanced_pbp.rebound.Rebound</i> <i>attribute</i> ), 57
<i>is_personal_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.data_nba.foul.DataFoul</i> <i>attribute</i> ), 36	<i>is_second_chance_event()</i> ( <i>pbp- stats.resources.enhanced_pbp.enhanced_pbp_item.EnhancedPbp</i> <i>method</i> ), 53
<i>is_personal_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.foul.Foul</i> <i>at-</i> <i>tribute</i> ), 55	<i>is_shooting_block_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.data_nba.foul.DataFoul</i> <i>attribute</i> ), 36
<i>is_personal_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.live.foul.LiveFoul</i> <i>attribute</i> ), 42	<i>is_shooting_block_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.foul.Foul</i> <i>at-</i> <i>tribute</i> ), 55
<i>is_personal_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.stats_nba.foul.StatsFoul</i> <i>attribute</i> ), 47	<i>is_shooting_block_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.live.foul.LiveFoul</i> <i>attribute</i> ), 42
<i>is_personal_take_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.data_nba.foul.DataFoul</i> <i>attribute</i> ), 36	<i>is_shooting_block_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.stats_nba.foul.StatsFoul</i> <i>attribute</i> ), 48
<i>is_personal_take_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.foul.Foul</i> <i>at-</i> <i>tribute</i> ), 55	<i>is_shooting_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.data_nba.foul.DataFoul</i> <i>attribute</i> ), 36
<i>is_personal_take_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.live.foul.LiveFoul</i> <i>attribute</i> ), 42	<i>is_shooting_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.foul.Foul</i> <i>at-</i> <i>tribute</i> ), 55
<i>is_personal_take_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.stats_nba.foul.StatsFoul</i> <i>attribute</i> ), 47	<i>is_shooting_foul</i> ( <i>pbp- stats.resources.enhanced_pbp.live.foul.LiveFoul</i> <i>attribute</i> ), 42

```

is_step_out_of_bounds() (pbp-
    stats.resources.enhanced_pbp.turnover.Turnover JumpBall (class in pbp-
        class method), 59 stats.resources.enhanced_pbp.jump_ball),
is_technical (pbp-
    stats.resources.enhanced_pbp.data_nba.foul.DataFoul 56
    attribute), 36

```

lineup\_stats (pbp- stats.data\_loader.live.pbp.web), 18  
 stats.resources.possessions.possessions.PossessionLivePossessionFileLoader (class in pbp- stats.data\_loader.live.possessions.file), 19  
 attribute), 63  
 LiveBoxscoreFileLoader (class in pbp- stats.data\_loader.live.boxscore.file), 16  
 LiveBoxscoreItem (class in pbp- stats.resources.boxscore.live\_boxscore\_item), 33  
 LiveBoxscoreLoader (class in pbp- stats.data\_loader.live.boxscore.loader), 15  
 LiveBoxscoreWebLoader (class in pbp- stats.data\_loader.live.boxscore.web), 16  
 LiveEjection (class in pbp- stats.resources.enhanced\_pbp.live.ejection), 39  
 LiveEndOfPeriod (class in pbp- stats.resources.enhanced\_pbp.live.end\_of\_period), 40  
 LiveEnhancedPbpFactory (class in pbp- stats.resources.enhanced\_pbp.live.enhanced\_pbp\_factory), 44  
 LiveEnhancedPbpFileLoader (class in pbp- stats.data\_loader.live.enhanced\_pbp.file), 17  
 LiveEnhancedPbpItem (class in pbp- stats.resources.enhanced\_pbp.live.enhanced\_pbp\_item), 40  
 LiveEnhancedPbpLoader (class in pbp- stats.data\_loader.live.enhanced\_pbp.loader), 16  
 LiveEnhancedPbpWebLoader (class in pbp- stats.data\_loader.live.enhanced\_pbp.web), 17  
 LiveFieldGoal (class in pbp- stats.resources.enhanced\_pbp.live.field\_goal), 41  
 LiveFileLoader (class in pbp- stats.data\_loader.live.file\_loader), 19  
 LiveFoul (class in pbp- stats.resources.enhanced\_pbp.live.foul), 41  
 LiveFreeThrow (class in pbp- stats.resources.enhanced\_pbp.live.free\_throw), 42  
 LiveJumpBall (class in pbp- stats.resources.enhanced\_pbp.live.jump\_ball), 43  
 LivePbpFileLoader (class in pbp- stats.data\_loader.live.pbp.file), 18  
 LivePbpItem (class in pbp- stats.resources.pbp.live\_pbp\_item), 61  
 LivePbpLoader (class in pbp- stats.data\_loader.live.pbp.loader), 17  
 LivePbpWebLoader (class in pbp- stats.data\_loader.live.pbp.web), 18  
 stats.data\_loader.live.possessions.loader), 18  
 LivePossessionWebLoader (class in pbp- stats.data\_loader.live.possessions.web), 19  
 LiveRebound (class in pbp- stats.resources.enhanced\_pbp.live.rebound), 43  
 LiveReplay (class in pbp- stats.resources.enhanced\_pbp.live.replay), 43  
 LiveStartOfPeriod (class in pbp- stats.resources.enhanced\_pbp.live.start\_of\_period), 43  
 LiveSubstitution (class in pbp- stats.resources.enhanced\_pbp.live.substitution), 44  
 LiveTimeout (class in pbp- stats.resources.enhanced\_pbp.live.timeout), 44  
 LiveTurnover (class in pbp- stats.resources.enhanced\_pbp.live.turnover), 44  
 LiveViolation (class in pbp- stats.resources.enhanced\_pbp.live.violation), 45  
 LiveWebLoader (class in pbp- stats.data\_loader.live.web\_loader), 19  
 load\_data () (pbpstats.data\_loader.data\_nba.boxscore.file.DataNbaBoxscoreFileLoader method), 10  
 load\_data () (pbpstats.data\_loader.data\_nba.boxscore.web.DataNbaBoxscoreWebLoader method), 10  
 load\_data () (pbpstats.data\_loader.data\_nba.pbp.file.DataNbaPbpFileLoader method), 12  
 load\_data () (pbpstats.data\_loader.data\_nba.pbp.web.DataNbaPbpWebLoader method), 12  
 load\_data () (pbpstats.data\_loader.data\_nba.schedule.file.DataNbaScheduleFileLoader method), 14  
 load\_data () (pbpstats.data\_loader.data\_nba.schedule.web.DataNbaScheduleWebLoader method), 14  
 load\_data () (pbpstats.data\_loader.live.boxscore.file.LiveBoxscoreFileLoader method), 16  
 load\_data () (pbpstats.data\_loader.live.boxscore.web.LiveBoxscoreWebLoader method), 16  
 load\_data () (pbpstats.data\_loader.live.pbp.file.LivePbpFileLoader method), 18  
 load\_data () (pbpstats.data\_loader.live.pbp.web.LivePbpWebLoader method), 18  
 load\_data () (pbpstats.data\_loader.stats\_nba.boxscore.file.StatsNbaBoxscoreFileLoader method), 21  
 load\_data () (pbpstats.data\_loader.stats\_nba.boxscore.web.StatsNbaBoxscoreWebLoader method), 21

**L**

- `load_data()` (`pbpstats.data_loader.stats_nba.league_game_log.file.StatsNbaLeagueGameLogFileLoader` (pbp-  
method), 26 attribute), 42
- `load_data()` (`pbpstats.data_loader.stats_nba.league_game_log.web.StatsNbaLeagueGameLogWebLoader` (pbp-  
method), 26 stats.resources.enhanced\_pbp.stats\_nba.foul.StatsFoul
- `load_data()` (`pbpstats.data_loader.stats_nba.pbp.file.StatsNbaPbpFileLoader` (pbp-  
method), 23 attribute), 48
- `load_data()` (`pbpstats.data_loader.stats_nba.pbp.web.StatsNbaPbpWebLoader` (pbp-  
method), 23 offense\_team\_id attribute), 42
- `load_data()` (`pbpstats.data_loader.stats_nba.scoreboard.file.StatsNbaScoreboardFileLoader` (pbp-  
method), 27 stats.resources.possessions.possession.Possession  
attribute), 62
- `load_data()` (`pbpstats.data_loader.stats_nba.scoreboard.web.StatsNbaScoreboardWebLoader` (pbp-  
method), 27 opponent\_stats attribute), 62
- `load_data()` (`pbpstats.data_loader.stats_nba.shots.file.StatsNbaShotsFileLoader` (pbp-  
method), 28 stats.resources.possessions.possessions.Possessions  
attribute), 63
- `load_data()` (`pbpstats.data_loader.stats_nba.shots.web.StatsNbaShotsWebLoader` (pbp-  
method), 28 oreb (pbpstats.resources.enhanced\_pbp.data\_nba.rebound.DataRebound  
attribute), 34
- `load_data()` (`pbpstats.data_loader.stats_nba.summary.file.StatsNbaSummaryFileLoader` (pbp-  
method), 29 oreb (pbpstats.resources.enhanced\_pbp.live.rebound.LiveRebound  
attribute), 45
- `load_data()` (`pbpstats.data_loader.stats_nba.summary.web.StatsNbaSummaryWebLoader` (pbp-  
method), 29 oreb (pbpstats.resources.enhanced\_pbp.rebound.Rebound  
attribute), 45
- `load_data()` (`pbpstats.data_loader.stats_nba.summary.web.StatsNbaSummaryWebLoader` (pbp-  
method), 29 oreb (pbpstats.resources.enhanced\_pbp.stats\_nba.rebound.StatsRebound  
attribute), 49

**M**

- `make_list_of_dicts()` (pbp- outgoing\_player\_id (pbp-  
stats.data\_loader.stats\_nba.base.StatsNbaLoaderBase attribute), 38
- `make_list_of_dicts()` (pbp- outgoing\_player\_id (pbp-  
stats.data\_loader.stats\_nba.shots.loader.StatsNbaShotsLoader attribute), 44
- `missed_shot` (pbpstats.resources.enhanced\_pbp.data\_nba.rebound.DataRebound (pbp-  
attribute), 37 stats.resources.enhanced\_pbp.substitution.StatsSubstitution  
attribute), 50
- `missed_shot` (pbpstats.resources.enhanced\_pbp.live.rebound.LiveRebound (pbp-  
attribute), 43 outgoing\_player\_id attribute), 50
- `missed_shot` (pbpstats.resources.enhanced\_pbp.rebound.Rebound (pbp-  
attribute), 57 stats.resources.enhanced\_pbp.substitution.Substitution  
attribute), 58
- `missed_shot` (pbpstats.resources.enhanced\_pbp.stats\_nba.rebound.StatsRebound (pbp-  
attribute), 49 overturn\_ruling attribute), 37

**N**

- `NbaEnhancedPbpLoader` (class in pbp- overturn\_ruling (pbp-  
stats.data\_loader.nba\_enhanced\_pbp\_loader), 30 stats.resources.enhanced\_pbp.live.replay.LiveReplay  
attribute), 43
- `NbaPossessionLoader` (class in pbp- overturn\_ruling (pbp-  
stats.data\_loader.nba\_possession\_loader), 30 stats.resources.enhanced\_pbp.replay.Replay  
attribute), 57
- `num_ft_for_trip` (pbp- overturn\_ruling (pbp-  
stats.resources.enhanced\_pbp.free\_throw.FreeThrow attribute), 49

**P**

- `number_of_fta_for_foul` (pbp- parent\_object (pbp-  
stats.resources.enhanced\_pbp.data\_nba.foul.DataFoul stats.data\_loader.data\_nba.boxscore.loader.DataNbaBoxscoreLo  
attribute), 36 attribute), 10
- `number_of_fta_for_foul` (pbp- parent\_object (pbp-  
stats.resources.enhanced\_pbp.foul.Foul at- stats.data\_loader.data\_nba.enhanced\_pbp.loader.DataNbaEnhan  
tribute), 55 attribute), 11



parent\_object (pbp- (module), 11  
stats.data\_loader.data\_nba.pbp.loader.DataNbaPbpLoader (module), 10  
attribute), 12

parent\_object (pbp- pbpstats.data\_loader.data\_nba.enhanced\_pbp.web  
stats.data\_loader.data\_nba.possessions.loader.DataNbaPossessionsLoader (module), 13  
attribute), 13 pbpstats.data\_loader.data\_nba.file\_loader

parent\_object (pbp- (module), 15  
stats.data\_loader.data\_nba.schedule.loader.DataNbaScheduleLoader (module), 12  
attribute), 14 pbpstats.data\_loader.data\_nba.pbp.file

parent\_object (pbp- pbpstats.data\_loader.data\_nba.pbp.loader  
stats.data\_loader.live.boxscore.loader.LiveBoxscoreLoader (module), 11  
attribute), 16 pbpstats.data\_loader.data\_nba.pbp.web

parent\_object (pbp- (module), 12  
stats.data\_loader.live.enhanced\_pbp.loader.LiveEnhancedPbpLoader (module), 13  
attribute), 17 pbpstats.data\_loader.data\_nba.possessions.file

parent\_object (pbp- pbpstats.data\_loader.data\_nba.possessions.loader  
stats.data\_loader.live.pbp.loader.LivePbpLoader (module), 12  
attribute), 18 pbpstats.data\_loader.data\_nba.possessions.web

parent\_object (pbp- (module), 13  
stats.data\_loader.live.possessions.loader.LivePossessionsLoader (module), 14  
attribute), 19 pbpstats.data\_loader.data\_nba.schedule.file

parent\_object (pbp- pbpstats.data\_loader.data\_nba.schedule.loader  
stats.data\_loader.stats\_nba.boxscore.loader.StatsNbaBoxscoreLoader (module), 13  
attribute), 21 pbpstats.data\_loader.data\_nba.schedule.web

parent\_object (pbp- (module), 14  
stats.data\_loader.stats\_nba.enhanced\_pbp.loader.StatsNbaEnhancedPbpLoader (module), 15  
attribute), 22 pbpstats.data\_loader.data\_nba.web\_loader

parent\_object (pbp- pbpstats.data\_loader.factory (module), 29  
stats.data\_loader.stats\_nba.league\_game\_log.loader.StatsNbaLeagueGameLogLoader (module), 16  
attribute), 25 pbpstats.data\_loader.live.boxscore.file

parent\_object (pbp- pbpstats.data\_loader.live.boxscore.loader  
stats.data\_loader.stats\_nba.pbp.loader.StatsNbaPbpLoader (module), 15  
attribute), 23 pbpstats.data\_loader.live.boxscore.web

parent\_object (pbp- (module), 16  
stats.data\_loader.stats\_nba.possessions.loader.StatsNbaPossessionsLoader (module), 17  
attribute), 24 pbpstats.data\_loader.live.enhanced\_pbp.file

parent\_object (pbp- pbpstats.data\_loader.live.enhanced\_pbp.loader  
stats.data\_loader.stats\_nba.scoreboard.loader.StatsNbaScoreboardLoader (module), 18  
attribute), 26 pbpstats.data\_loader.live.enhanced\_pbp.web

parent\_object (pbp- (module), 17  
stats.data\_loader.stats\_nba.shots.loader.StatsNbaShotsLoader (module), 19  
attribute), 27 pbpstats.data\_loader.live.file\_loader

parent\_object (pbp- pbpstats.data\_loader.live.pbp.file (mod-  
stats.data\_loader.stats\_nba.summary.loader.StatsNbaSummaryLoader (module), 17  
attribute), 28 pbpstats.data\_loader.live.pbp.loader

Pbp (class in pbpstats.resources.pbp.pbp), 61 (module), 17

pbpstats.client (module), 64 pbpstats.data\_loader.live.pbp.web (mod-  
ule), 18

pbpstats.data\_loader.data\_nba.boxscore.file (module), 10 pbpstats.data\_loader.live.possessions.file

pbpstats.data\_loader.data\_nba.boxscore.loader (module), 19 pbpstats.data\_loader.live.possessions.loader

(module), 9 pbpstats.data\_loader.live.possessions.web

pbpstats.data\_loader.data\_nba.boxscore.web (module), 18 pbpstats.data\_loader.live.possessions.web

(module), 10

pbpstats.data\_loader.data\_nba.enhanced\_pbp.file (module), 19

---

pbpstats.data_loader.live.web_loader (module), 19	pbpstats.data_loader.stats_nba.summary.loader (module), 28
pbpstats.data_loader.nba_enhanced_pbp_loader (module), 30	pbpstats.data_loader.stats_nba.summary.web (module), 29
pbpstats.data_loader.nba_possession_loader (module), 30	pbpstats.data_loader.stats_nba.web_loader (module), 29
pbpstats.data_loader.stats_nba.base (module), 20	pbpstats.objects.day (module), 30
pbpstats.data_loader.stats_nba.boxscore.file (module), 21	pbpstats.objects.game (module), 31
pbpstats.data_loader.stats_nba.boxscore.loader (module), 20	pbpstats.objects.season (module), 31
pbpstats.data_loader.stats_nba.boxscore.web (module), 21	pbpstats.resources.boxscore.boxscore (module), 32
pbpstats.data_loader.stats_nba.enhanced_pbp.file (module), 22	pbpstats.resources.boxscore.data_nba_boxscore_item (module), 33
pbpstats.data_loader.stats_nba.enhanced_pbp.loader (module), 21	pbpstats.resources.boxscore.live_boxscore_item (module), 33
pbpstats.data_loader.stats_nba.enhanced_pbp.web (module), 22	pbpstats.resources.boxscore.stats_nba_boxscore_item (module), 33
pbpstats.data_loader.stats_nba.file_loader (module), 29	pbpstats.resources.enhanced_pbp.data_nba.ejection (module), 34
pbpstats.data_loader.stats_nba.league_game_log (module), 25	pbpstats.resources.enhanced_pbp.data_nba.end_of_per (module), 34
pbpstats.data_loader.stats_nba.league_game_log (module), 25	pbpstats.resources.enhanced_pbp.data_nba.enhanced_p (module), 34
pbpstats.data_loader.stats_nba.league_game_log (module), 26	pbpstats.resources.enhanced_pbp.data_nba.enhanced_p (module), 35
pbpstats.data_loader.stats_nba.pbp.file (module), 23	pbpstats.resources.enhanced_pbp.data_nba.field_goa (module), 35
pbpstats.data_loader.stats_nba.pbp.loader (module), 23	pbpstats.resources.enhanced_pbp.data_nba.foul (module), 35
pbpstats.data_loader.stats_nba.pbp.web (module), 23	pbpstats.resources.enhanced_pbp.data_nba.free_throv (module), 36
pbpstats.data_loader.stats_nba.possessions.file (module), 24	pbpstats.resources.enhanced_pbp.data_nba.jump_ball (module), 37
pbpstats.data_loader.stats_nba.possessions.loader (module), 24	pbpstats.resources.enhanced_pbp.data_nba.rebound (module), 37
pbpstats.data_loader.stats_nba.possessions.web (module), 24	pbpstats.resources.enhanced_pbp.data_nba.replay (module), 37
pbpstats.data_loader.stats_nba.scoreboard.file (module), 26	pbpstats.resources.enhanced_pbp.data_nba.start_of_p (module), 38
pbpstats.data_loader.stats_nba.scoreboard.loader (module), 26	pbpstats.resources.enhanced_pbp.data_nba.substitut (module), 38
pbpstats.data_loader.stats_nba.scoreboard.web (module), 27	pbpstats.resources.enhanced_pbp.data_nba.timeout (module), 38
pbpstats.data_loader.stats_nba.shots.file (module), 28	pbpstats.resources.enhanced_pbp.data_nba.turnover (module), 39
pbpstats.data_loader.stats_nba.shots.loader (module), 27	pbpstats.resources.enhanced_pbp.data_nba.violation (module), 51
pbpstats.data_loader.stats_nba.shots.web (module), 28	pbpstats.resources.enhanced_pbp.ejection (module), 52
pbpstats.data_loader.stats_nba.summary.file (module), 29	pbpstats.resources.enhanced_pbp.end_of_period (module), 52
	pbpstats.resources.enhanced_pbp.enhanced_pbp (module), 52
	pbpstats.resources.enhanced_pbp.enhanced_pbp_item (module), 52





<code>stats.resources.boxscore.boxscore.Boxscore</code> <code>attribute</code> ), 32	<code>resource (pbpstats.data_loader.data_nba.enhanced_pbp.loader.DataNbaPbpLoader</code> <code>attribute</code> ), 11
<code>player_stats</code> ( <code>pbp-</code> <code>stats.resources.possessions.possessions.Possessions</code> <code>attribute</code> ), 63	<code>resource (pbpstats.data_loader.data_nba.pbp.loader.DataNbaPbpLoader</code> <code>attribute</code> ), 12
<code>player_team_map</code> ( <code>pbp-</code> <code>stats.resources.boxscore.boxscore.Boxscore</code> <code>attribute</code> ), 32	<code>resource (pbpstats.data_loader.data_nba.possessions.loader.DataNbaPos</code> <code>attribute</code> ), 13
<code>Possession</code> ( <code>class</code> <code>in</code> <code>pbp-</code> <code>stats.resources.possessions.possession</code> ), 62	<code>resource (pbpstats.data_loader.data_nba.schedule.loader.DataNbaSched</code> <code>attribute</code> ), 14
<code>possession_has_timeout</code> ( <code>pbp-</code> <code>stats.resources.possessions.possession.Possession</code> <code>attribute</code> ), 62	<code>resource (pbpstats.data_loader.live.boxscore.loader.LiveBoxscoreLoader</code> <code>attribute</code> ), 16
<code>possession_start_type</code> ( <code>pbp-</code> <code>stats.resources.possessions.possession.Possession</code> <code>attribute</code> ), 62	<code>resource (pbpstats.data_loader.live.enhanced_pbp.loader.LiveEnhanced</code> <code>attribute</code> ), 17
<code>possession_stats</code> ( <code>pbp-</code> <code>stats.resources.possessions.possession.Possession</code> <code>attribute</code> ), 62	<code>resource (pbpstats.data_loader.live.pbp.loader.LivePbpLoader</code> <code>attribute</code> ), 18
<code>Possessions</code> ( <code>class</code> <code>in</code> <code>pbp-</code> <code>stats.resources.possessions.possessions</code> ), 63	<code>resource (pbpstats.data_loader.live.possessions.loader.LivePossessionLo</code> <code>attribute</code> ), 19
<code>previous_possession_end_rebound_player_id</code> ( <code>pbpstats.resources.possessions.possession.Possession</code> <code>attribute</code> ), 62	<code>resource (pbpstats.data_loader.stats_nba.boxscore.loader.StatsNbaBoxs</code> <code>attribute</code> ), 21
<code>previous_possession_end_shooter_player_id</code> ( <code>pbpstats.resources.possessions.possession.Possession</code> <code>attribute</code> ), 62	<code>resource (pbpstats.data_loader.stats_nba.enhanced_pbp.loader.StatsNba</code> <code>attribute</code> ), 22
<code>previous_possession_end_steal_player_id</code> ( <code>pbpstats.resources.possessions.possession.Possession</code> <code>attribute</code> ), 62	<code>resource (pbpstats.data_loader.stats_nba.league_game_log.loader.Stats</code> <code>attribute</code> ), 25
<code>previous_possession_end_turnover_player_id</code> ( <code>pbpstats.resources.possessions.possession.Possession</code> <code>attribute</code> ), 62	<code>resource (pbpstats.data_loader.stats_nba.pbp.loader.StatsNbaPbpLoader</code> <code>attribute</code> ), 23
<code>previous_possession_ending_event</code> ( <code>pbp-</code> <code>stats.resources.possessions.possession.Possession</code> <code>attribute</code> ), 62	<code>resource (pbpstats.data_loader.stats_nba.possessions.loader.StatsNbaPo</code> <code>attribute</code> ), 24
<code>previous_possession_has_timeout</code> ( <code>pbp-</code> <code>stats.resources.possessions.possession.Possession</code> <code>attribute</code> ), 62	<code>resource (pbpstats.data_loader.stats_nba.scoreboard.loader.StatsNbaSc</code> <code>attribute</code> ), 26
<code>Rebound</code> ( <code>class</code> <code>in</code> <code>pbp-</code> <code>stats.resources.enhanced_pbp.rebound</code> ), 56	<code>resource (pbpstats.data_loader.stats_nba.shots.loader.StatsNbaShotsLo</code> <code>attribute</code> ), 28
<code>rebound</code> ( <code>pbpstats.resources.enhanced_pbp.field_goal.FieldGoal</code> <code>attribute</code> ), 54	<code>resource (pbpstats.data_loader.stats_nba.summary.loader.StatsNbaSum</code> <code>attribute</code> ), 29
<code>rebounds</code> ( <code>pbpstats.resources.enhanced_pbp.enhanced_pbp.EnhancedPbp</code> <code>attribute</code> ), 52	<code>resource (pbpstats.data_loader.stats_nba.replay.DataReplay</code> <code>attribute</code> ), 37
<code>Replay</code> ( <code>class</code> <code>in</code> <code>pbp-</code> <code>stats.resources.enhanced_pbp.replay</code> ), 57	<code>ruling_stands</code> ( <code>pbp-</code> <code>stats.resources.enhanced_pbp.data_nba.replay.DataReplay</code> <code>attribute</code> ), 37
<code>resource (pbpstats.data_loader.data_nba.boxscore.loader.DataNbaBoxscore)</code> <code>attribute</code> ), 10	<code>ruling_stands</code> ( <code>pbp-</code> <code>stats.resources.enhanced_pbp.live.replay.LiveReplay</code> <code>attribute</code> ), 43
	<code>ruling_stands</code> ( <code>pbp-</code> <code>stats.resources.enhanced_pbp.replay.Replay</code> <code>attribute</code> ), 57
	<code>ruling_stands</code> ( <code>pbp-</code> <code>stats.resources.enhanced_pbp.stats_nba.replay.StatsReplay</code> <code>attribute</code> ), 49
	<code>score_margin</code> ( <code>pbp-</code> <code>stats.resources.enhanced_pbp.enhanced_pbp_item.EnhancedPbp</code> <code>attribute</code> ), 53
	<code>Season</code> ( <code>class</code> <code>in</code> <code>pbpstats.objects.season</code> ), 32
	<code>season</code> ( <code>pbpstats.data_loader.stats_nba.base.StatsNbaLoaderBase</code> <code>attribute</code> ), 20
	<code>season_type</code> ( <code>pbpstats.data_loader.stats_nba.base.StatsNbaLoaderBase</code> <code>attribute</code> ), 20

seconds_remaining	(pbp- StatsFoul (class in pbp- stats.resources.enhanced_pbp.data_nba.enhanced_pbp_item.EnhancedPbpItem), 35	47
seconds_remaining	(pbp- StatsFreeThrow (class in pbp- stats.resources.enhanced_pbp.enhanced_pbp_item.EnhancedPbpItem), 53	48
seconds_remaining	(pbp- StatsJumpBall (class in pbp- stats.resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem), 41	49
seconds_remaining	(pbp- StatsNbaBoxscoreFileLoader (class in pbp- stats.resources.enhanced_pbp.stats_nba.enhanced_pbp_item.EnhancedPbpItem), 47	21
seconds_since_previous_event	(pbp- StatsNbaBoxscoreItem (class in pbp- stats.resources.boxscore.stats_nba_boxscore_item), 53	21
self_reb	(pbpstats.resources.enhanced_pbp.rebound.Rebound (class in pbp- stats.data_loader.stats_nba.boxscore.loader), 57	21
shot_data	(pbpstats.resources.enhanced_pbp.field_goal.FieldGoal (class in pbp- stats.data_loader.stats_nba.boxscore.web), 54	21
shot_type	(pbpstats.resources.enhanced_pbp.field_goal.FieldGoal (class in pbp- stats.data_loader.stats_nba.enhanced_pbp_factory), 54	46
shot_type	(pbpstats.resources.enhanced_pbp.free_throw.FreeThrow (class in pbp- stats.data_loader.stats_nba.enhanced_pbp_loader), 56	22
shot_value	(pbpstats.resources.enhanced_pbp.data_nba.field_goal_data_nba.FieldGoalDataNbaFileLoader (class in pbp- stats.data_loader.stats_nba.enhanced_pbp.file), 35	22
shot_value	(pbpstats.resources.enhanced_pbp.field_goal.FieldGoal (class in pbp- stats.data_loader.stats_nba.enhanced_pbp_loader), 54	22
shot_value	(pbpstats.resources.enhanced_pbp.free_throw.FreeThrow (class in pbp- stats.data_loader.stats_nba.enhanced_pbp_loader), 56	22
shot_value	(pbpstats.resources.enhanced_pbp.live.field_goal.LiveFieldGoal (class in pbp- stats.data_loader.stats_nba.enhanced_pbp.web), 41	23
shot_value	(pbpstats.resources.enhanced_pbp.stats_nba.field_goal.StatsFieldGoal (class in pbp- stats.data_loader.stats_nba.file_loader), 47	29
Shots	(class in pbpstats.resources.shots.shots), 63	60
start_score_margin	(pbp- StatsNbaGameItem (class in pbp- stats.resources.games.stats_nba_game_item), 62	60
start_time	(pbpstats.resources.possessions.possession.Possession (class in pbp- stats.data_loader.stats_nba.league_game_log.file), 62	25
StartOfPeriod	(class in pbp- StatsNbaLeagueGameLogFileLoader (class in pbp- stats.data_loader.stats_nba.league_game_log.loader), 57	25
StatsEjection	(class in pbp- StatsNbaLeagueGameLogWebLoader (class in pbp- stats.data_loader.stats_nba.league_game_log.web), 45	26
StatsEndOfPeriod	(class in pbp- StatsNbaPbpFileLoader (class in pbp- stats.data_loader.stats_nba.pbp.file), 46	20
StatsEnhancedPbpItem	(class in pbp- StatsNbaPbpItem (class in pbp- stats.data_loader.stats_nba.base), 46	23
StatsFieldGoal	(class in pbp- StatsNbaPbpItem (class in pbp- stats.data_loader.stats_nba.pbp.file), 47	

<i>stats.resources.pbp.stats_nba_pbp_item</i> ), 61	<i>StatsSubstitution</i> (class in <i>pbp- stats.resources.enhanced_pbp.stats_nba.substitution</i> ), 50
<i>StatsNbaPbpLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.pbp.loader</i> ), 23	<i>StatsTimeout</i> (class in <i>pbp- stats.resources.enhanced_pbp.stats_nba.timeout</i> ), 50
<i>StatsNbaPbpWebLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.pbp.web</i> ), 23	<i>StatsTurnover</i> (class in <i>pbp- stats.resources.enhanced_pbp.stats_nba.turnover</i> ), 50
<i>StatsNbaPossessionFileLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.possessions.file</i> ), 24	<i>StatsViolation</i> (class in <i>pbp- stats.resources.enhanced_pbp.stats_nba.violation</i> ), 51
<i>StatsNbaPossessionLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.possessions.loader</i> ), 24	<i>stripped_descriptor</i> ( <i>pbp- stats.resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem</i> attribute), 41
<i>StatsNbaPossessionWebLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.possessions.web</i> ), 24	<i>stripped_sub_type</i> ( <i>pbp- stats.resources.enhanced_pbp.live.enhanced_pbp_item.LiveEnhancedPbpItem</i> attribute), 41
<i>StatsNbaScoreboardFileLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.scoreboard.file</i> ), 26	<i>sub_type</i> ( <i>pbpstats.resources.enhanced_pbp.live.end_of_period.LiveEndOfPeriod</i> attribute), 40
<i>StatsNbaScoreboardLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.scoreboard.loader</i> ), 26	<i>sub_type</i> ( <i>pbpstats.resources.enhanced_pbp.live.start_of_period.LiveStartOfPeriod</i> attribute), 44
<i>StatsNbaScoreboardWebLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.scoreboard.web</i> ), 27	<i>Substitution</i> (class in <i>pbp- stats.resources.enhanced_pbp.substitution</i> ), 58
<i>StatsNbaShot</i> (class in <i>pbp- stats.resources.shots.stats_nba_shot</i> ), 63	<i>support_ruling</i> ( <i>pbp- stats.resources.enhanced_pbp.data_nba.replay.DataReplay</i> attribute), 37
<i>StatsNbaShotsFileLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.shots.file</i> ), 28	<i>support_ruling</i> ( <i>pbp- stats.resources.enhanced_pbp.live.replay.LiveReplay</i> attribute), 43
<i>StatsNbaShotsLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.shots.loader</i> ), 27	<i>support_ruling</i> ( <i>pbp- stats.resources.enhanced_pbp.replay.Replay</i> attribute), 57
<i>StatsNbaShotsWebLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.shots.web</i> ), 28	<i>support_ruling</i> ( <i>pbp- stats.resources.enhanced_pbp.stats_nba.replay.StatsReplay</i> attribute), 49
<i>StatsNbaSummaryFileLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.summary.file</i> ), 29	<b>T</b>
<i>StatsNbaSummaryLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.summary.loader</i> ), 28	<i>team_items</i> ( <i>pbpstats.resources.boxscore.boxscore.Boxscore</i> attribute), 32
<i>StatsNbaSummaryWebLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.summary.web</i> ), 29	<i>team_stats</i> ( <i>pbpstats.resources.possessions.possessions.Possessions</i> attribute), 63
<i>StatsNbaWebLoader</i> (class in <i>pbp- stats.data_loader.stats_nba.web_loader</i> ), 29	<i>TeamHasBackToBackPossessionsException</i> , 24
<i>StatsRebound</i> (class in <i>pbp- stats.resources.enhanced_pbp.stats_nba.rebound</i> ), 49	<i>Timeout</i> (class in <i>pbp- stats.resources.enhanced_pbp.timeout</i> ), 58
<i>StatsReplay</i> (class in <i>pbp- stats.resources.enhanced_pbp.stats_nba.replay</i> ), 49	<i>total_seconds</i> ( <i>pbp- stats.resources.boxscore.live_boxscore_item.LiveBoxscoreItem</i> attribute), 33
<i>StatsStartOfPeriod</i> (class in <i>pbp- stats.resources.enhanced_pbp.stats_nba.start_of_period</i> ), 50	<i>Turnover</i> (class in <i>pbp- stats.resources.enhanced_pbp.turnover</i> ), 59

turnovers (*pbpstats.resources.enhanced\_pbp.enhanced\_pbp.EnhancedPbp*  
*attribute*), [52](#)

## V

video\_url (*pbpstats.resources.enhanced\_pbp.stats\_nba.enhanced\_pbp\_item.StatsEnhancedPbpItem*  
*attribute*), [47](#)

Violation (class in *pbp-*  
*stats.resources.enhanced\_pbp.violation*),  
[59](#)

## W

winning\_team (*pbp-*  
*stats.resources.enhanced\_pbp.jump\_ball.JumpBall*  
*attribute*), [56](#)