
Pyeuromil Documentation

Release beta

Acpirience

Dec 06, 2021

Contents

1	pyeuromil	3
1.1	Usage	3
1.2	Installation	5
1.3	Documentation	5
1.4	Compatibility	5
1.5	Licence	5
1.6	Authors	5
	Python Module Index	9
	Index	11

[genindex](#) | [modindex](#)

CHAPTER 1

pyeuromil

build passing

A Python library to check and analyse Euromillions results
(Last result: 2021/12/03)

1.1 Usage

```
import pyeuromil as em
from datetime import date

# List of draw dates of September 2018
for result in em.euro_draw_dates(date(2018, 9, 1), date(2018, 9, 30)):
    print(result)
```

```
2018-09-28
2018-09-25
2018-09-21
2018-09-18
2018-09-14
2018-09-11
2018-09-07
2018-09-04
```

```
import pyeuromil as em
from datetime import date

# first week of october 2018 results
result_list = em.euro_results(date(2018, 10, 1), date(2018, 10, 7))
for result in result_list:
    print(result)
```

```
Result(date=datetime.date(2018, 10, 5), numbers=[8, 16, 24, 26, 35], stars=[3, 11])
Result(date=datetime.date(2018, 10, 2), numbers=[7, 17, 29, 37, 45], stars=[3, 11])
```

```
from pyeuromil import Plays, Grid
from datetime import date
import pprint

pp = pprint.PrettyPrinter(indent=2)

plays = Plays()
game = Grid([15, 18, 32, 35, 40],[1, 7], star_plus=True)
plays.append(game, start=date(2018,9,28), end=date(2018,10,30), tuesday=True, ↵
↵friday=True)

for play in plays:
    pp.pprint(Plays.play_summary(play))
```

```
[ { 'date': datetime.date(2018, 10, 23),
    'numbers': [32],
    'ranking': 0,
    'ranking_star_plus': 0,
    'stars': []},
  { 'date': datetime.date(2018, 10, 19),
    'numbers': [],
    'ranking': 0,
    'ranking_star_plus': 0,
    'stars': []},
  { 'date': datetime.date(2018, 10, 16),
    'numbers': [15, 40],
    'ranking': 12,
    'ranking_star_plus': 9,
    'stars': [1]},
  { 'date': datetime.date(2018, 10, 12),
    'numbers': [],
    'ranking': 0,
    'ranking_star_plus': 0,
    'stars': []},
  { 'date': datetime.date(2018, 10, 9),
    'numbers': [],
    'ranking': 0,
    'ranking_star_plus': 0,
    'stars': []},
  { 'date': datetime.date(2018, 10, 5),
    'numbers': [35],
    'ranking': 0,
    'ranking_star_plus': 0,
    'stars': []},
  { 'date': datetime.date(2018, 10, 2),
```

(continues on next page)

(continued from previous page)

```
'numbers': [],
'ranking': 0,
'ranking_star_plus': 0,
'stars': [],
{ 'date': datetime.date(2018, 9, 28),
  'numbers': [],
  'ranking': 0,
  'ranking_star_plus': 0,
  'stars': []}]
```

1.2 Installation

Install and update using [pip](#):

```
pip install pyeuromil
```

1.3 Documentation

Documentation is available at [ReadTheDocs](#)

1.4 Compatibility

python >= 3.6

1.5 Licence

MIT License

1.6 Authors

pyeuromil was written by [Acpirience](#).

1.6.1 pyeuromil

pyeuromil package

pyeuromil - A python library to check and analyse Euromillions results

Submodules

pyeuromil.euromil module

Euromil: give the Euromillions results and historical draw dates

`pyeuromil.euromil.euro_draw_dates` (*start_date=None, end_date=None*)
return the list of Euromillions draws between a *start_date* and an *end_date*

Parameters

- **start_date** (*date*) – start date
- **end_date** (*date*) – end_date

Returns list of date

Return type list of date

`pyeuromil.euromil.euro_results` (*start_date=None, end_date=None*)
return the list of Euromillions result between a *start_date* and an *end_date*

Parameters

- **start_date** (*date*) – start date
- **end_date** (*date*) – end_date

Returns list of EuroResult

Raises ValueError

`pyeuromil.euromil.euro_stats` (*start_date=None, end_date=None*)
return a count of numbers and stars between a *start_date* and an *end_date*

Parameters

- **start_date** (*date*) – start date
- **end_date** (*date*) – end_date

Returns a dictionary with the numbers of drows per numbers and stars

Return type dict

pyeuromil.euromil_play module

Euromy_play: stores Euromillions plays and give summary of wins

class `pyeuromil.euromil_play.Grid` (*numbers, stars, star_plus=False*)
Bases: object

contains a grid played at an Euromillions game

static `check_numbers` (*numbers, stars*)
check if parameters for the grid are corrects

Parameters

- **numbers** (*list of int*) – The list of numbers played
- **stars** (*list of int*) – The list of stars played

Returns check status and error message is status is ko

Return type bool, string

evaluate_grid (*result*)

returns the list of numbers and stars both in a Grid and a result

Parameters **result** (*EuroResult*) – The result we want to check the grid against

Returns the list of the numbers in the grid and the result and the list of the stars in the grid and the result

Return type list of int, list of int

class pyeuromil.euromil_play.**Plays**

Bases: object

Stores plays and is used to validate if plays are wins or loses

append (*grid*, *, *start=None*, *end=None*, *tuesday=False*, *friday=False*)

Adds a new grid and dates of play to Plays

Parameters

- **grid** – The grid played
- **start_date** (*date*) – start date
- **end_date** (*date*) – end_date
- **tuesday** (*date*) – Indicates in the grid is played on tuedays
- **friday** (*date*) – Indicates in the grid is played on fridays

Raises ValueError

static play_summary (*play*, *only_wins=False*)

returns the summary of numbers and win rankings for a play (ensemble of games)

Parameters

- **play** (*play*) – The play the summary will be based upon
- **only_wins** (*bool*) – show only winning games in summary

Returns dictionary with date, numbers, stars, ranking and ranking_star_plus keys

Return type dict

static ranking (*numbers*, *stars*)

returns ranking for a normal game and ranking for a Star Plus game Ranking is 0 if nothing was won

Parameters

- **numbers** (*list of int*) – The list of numbers played
- **stars** (*list of int*) – The list of stars played

Returns normal ranking and Star Plus ranking

Return type int, int

Raises ValueError

pyeuromil.euromil_utils module

constants and definition used by the classes

`pyeuromil.euromil_utils.EuroPlay`

A play of the Euromillions, contains a grid, a start date, an end date and if the grid was played on tuesdays of fridays

alias of `pyeuromil.euromil_utils.Play`

`pyeuromil.euromil_utils.EuroResult`

A result of the Euromillions, contains numbers, starts and the date of the draw

alias of `pyeuromil.euromil_utils.Result`

p

- `pyeuromil`, 5
- `pyeuromil.euromil`, 6
- `pyeuromil.euromil_play`, 6
- `pyeuromil.euromil_utils`, 7

A

`append()` (*pyeuromil.euromil_play.Plays method*), 7

C

`check_numbers()` (*pyeuromil.euromil_play.Grid static method*), 6

E

`euro_draw_dates()` (*in module pyeuromil.euromil*), 6

`euro_results()` (*in module pyeuromil.euromil*), 6

`euro_stats()` (*in module pyeuromil.euromil*), 6

`EuroPlay` (*in module pyeuromil.euromil_utils*), 7

`EuroResult` (*in module pyeuromil.euromil_utils*), 8

`evaluate_grid()` (*pyeuromil.euromil_play.Grid method*), 7

G

`Grid` (*class in pyeuromil.euromil_play*), 6

P

`play_summary()` (*pyeuromil.euromil_play.Plays static method*), 7

`Plays` (*class in pyeuromil.euromil_play*), 7

`pyeuromil` (*module*), 5

`pyeuromil.euromil` (*module*), 6

`pyeuromil.euromil_play` (*module*), 6

`pyeuromil.euromil_utils` (*module*), 7

R

`ranking()` (*pyeuromil.euromil_play.Plays static method*), 7