

 brain

Search...

Submodules

`data_categories`

`data_fields`

`data_sets`

```

class Brain
    Brain()
    client
    instrument_type
    region
    delay
    universe
    get_instrument_types()
    get_regions()
    get_delays()
    get_universes()
    get_instruments()
    get_data_categories()
    get_data_sets()
    get_data_set()
    get_data_fields()
    get_data_field()
    get_data_field_details()
    get_data_frame()

    get_universes()
    get_instruments()
    get_data_categories()
    get_data_sets()
    get_data_set()
    get_data_fields()
    get_data_field()
    get_data_field_details()
    get_data_frame()

class Client
    Client()
    raise_on_unexpected_status
    with_headers()
    with_cookies()
    with_timeout()
    set_httpx_client()
    get_httpx_client()
    set_async_httpx_client()
    get_async_httpx_client()
    get_client()

```

brain

A client library for accessing BRAIN

```

@define
class Brain:

```

```

# Brain(
    Brain_instruments: Dict[tuple, List[str]] = None,
    *,
    client: Client = None,
    instrument_type: str = 'EQUITY',
    region: str = 'USA',
    delay: int = 1,
    universe: str = 'TOP3000'
)

```

Method generated by attrs for class Brain.

`client: Client`

`instrument_type: str`

`region: str`

```

@define
class Client:

```

A class for keeping track of data related to the API

The following are accepted as keyword arguments and will be used to construct httpx Clients internally:

```

``base_url``: The base URL for the API, all requests are made to a relative path to this URL
``cookies``: A dictionary of cookies to be sent with every request
``headers``: A dictionary of headers to be sent with every request
``timeout``: The maximum amount of a time a request can take. API functions will raise httpx.TimeoutException if this is exceeded.
``verify_ssl``: Whether or not to verify the SSL certificate of the API server. This should be True but can be set to False for testing purposes.
``follow_redirects``: Whether or not to follow redirects. Default value is False.
``httpx_args``: A dictionary of additional arguments to be passed to the ``httpx.Client`` constructor

```

Attributes: `raise_on_unexpected_status`: Whether or not to raise an errors.UnexpectedStatus if the API returns a status code that was not documented in the source OpenAPI document. Can also be provided as a keyword argument to the constructor.

```

Client(
    base_url: str
)

```

API Documentation

```
class DataCategory
    DataCategory()
        id
        name
        dataset_count
        field_count
        alpha_count
        user_count
        value_score
        region
        children
        additional_properties
        to_dict()
        from_dict()
        additional_keys
class DataField
    DataField()
        id
```

```
class DataField
    DataField()
        id
        description
        dataset
        category
        subcategory
        type
        region
        delay
        universe
        data
        coverage
        turnover
        user_count
        alpha_count
        pyramid_multiplier
        themes
        additional_properties
        to_dict()
        from_dict()
        additional_keys
```

brain.models

Contains all the data models used in inputs/outputs

class DataCategory:

Rejects unexpected fields, read only fields, and values of the wrong type

Attributes: id (str): name (str): dataset_count (int): field_count (int): alpha_count (int): user_count (int): value_score (float): region (List[str]): children (List['DataSubCategory']):

```
# DataCategory
    id: str,
    name: str,
    dataset_count: int,
    field_count: int,
    alpha_count: int,
    user_count: int,
    value_score: float,
    region: List[str],
    children: List[DataSubCategory]
)
```

class DataField:

Rejects unexpected fields, read only fields, and values of the wrong type

Attributes: id (str): description (str): dataset (DataSummary): Rejects unexpected fields, read only fields, and values of the wrong type category (DataSummary): Rejects unexpected fields, read only fields, and values of the wrong type subcategory (DataSummary): Rejects unexpected fields, read only fields, and values of the wrong type type (str): region (Union[Unset, str]): delay (Union[Unset, int]): universe (Union[Unset, str]): data (Union[Unset, List['DataFieldDetails']]): coverage (Union[Unset, float]): turnover (Union[Unset, float]): user_count (Union[Unset, int]): alpha_count (Union[Unset, int]): pyramid_multiplier (Union[Unset, float]): themes (Union[Unset, List['Theme']]):

```
DataField(
    id: str,
    description: str,
    dataset: DataSummary,
    category: DataSummary,
    subcategory: DataSummary,
    type: str,
    region: Union[brain.types.Unset, str] = <brain.types.Unset object>,
    delay: Union[brain.types.Unset, int] = <brain.types.Unset object>,
    universe: Union[brain.types.Unset, str] = <brain.types.Unset object>,
    data: Union[brain.types.Unset, List[DataFieldDetails]] = <brain.types.Unset object>,
    coverage: Union[brain.types.Unset, float] = <brain.types.Unset object>,
    turnover: Union[brain.types.Unset, float] = <brain.types.Unset object>,
    user_count: Union[brain.types.Unset, int] = <brain.types.Unset object>,
    alpha_count: Union[brain.types.Unset, int] = <brain.types.Unset object>,
    pyramid_multiplier: Union[brain.types.Unset, float] = <brain.types.Unset object>,
```

```
class DataFieldDetails
    DataFieldDetails()
    region
    delay
    universe
    coverage
    user_count
    alpha_count
    themes
    turnover
    pyramid_multiplier
    brainlabs_url
    additional_properties
    to_dict()
    from_dict()
    additional_keys
```

class DataFieldDetails:

Rejects unexpected fields, read only fields, and values of the wrong type

Attributes: region (str): delay (int): universe (str): coverage (float): user_count (int): alpha_count (int): themes (List[Theme]): turnover (Union[Unset, float]): pyramid_multiplier (Union[Unset, float]): brainlabs_url (Union[Unset, str]):

DataFieldDetails(

```
    region: str,
    delay: int,
    universe: str,
    coverage: float,
    user_count: int,
    alpha_count: int,
    themes: List[Theme],
    turnover: Union[brain.types.Unset, float] = <brain.types.Unset object>,
    pyramid_multiplier: Union[brain.types.Unset, float] = <brain.types.Unset object>,
    brainlabs_url: Union[brain.types.Unset, str] = <brain.types.Unset object>
)
```

```
class DataSet
    DataSet()
    id
    name
    description
    category
    subcategory
    research_papers
    region
    delay
    universe
    data
    coverage
    turnover
    value_score
    user_count
    alpha_count
    field_count
    pyramid_multiplier
    themes
    additional_properties
```

class DataSet:

Rejects unexpected fields, read only fields, and values of the wrong type

Attributes: id (str): name (str): description (str): category (DataSummary): Rejects unexpected fields, read only fields, and values of the wrong type subcategory (DataSummary): Rejects unexpected fields, read only fields, and values of the wrong type research_papers (List[ResearchPaper]): region (Union[Unset, str]): delay (Union[Unset, int]): universe (Union[Unset, str]): data (Union[Unset, List[DataSetDetails]]): coverage (Union[Unset, float]): turnover (Union[Unset, float]): value_score (Union[Unset, float]): user_count (Union[Unset, int]): alpha_count (Union[Unset, int]): field_count (Union[Unset, int]): pyramid_multiplier (Union[Unset, float]): themes (Union[Unset, List['Theme']]):

DataSet(

```
    id: str,
    name: str,
    description: str,
    category: DataSummary,
    subcategory: DataSummary,
    research_papers: List[ResearchPaper],
    region: Union[brain.types.Unset, str] = <brain.types.Unset object>,
    delay: Union[brain.types.Unset, int] = <brain.types.Unset object>,
    universe: Union[brain.types.Unset, str] = <brain.types.Unset object>,
    data: Union[brain.types.Unset, List[DataSetDetails]] = <brain.types.Unset object>,
    coverage: Union[brain.types.Unset, float] = <brain.types.Unset object>,
    turnover: Union[brain.types.Unset, float] = <brain.types.Unset object>,
    value_score: Union[brain.types.Unset, float] = <brain.types.Unset object>,
    user_count: Union[brain.types.Unset, int] = <brain.types.Unset object>,
```

```
class DataSetDetails
    DataSetDetails()
    region
    delay
    universe
    value_score
    field_count
    themes
    coverage
    turnover
    user_count
    alpha_count
    pyramid_multiplier
    additional_properties
    to_dict()
    from_dict()
    additional_keys
```

class DataSetDetails:

Rejects unexpected fields, read only fields, and values of the wrong type

Attributes: region (str): delay (int): universe (str): value_score (float): field_count (int): themes (List[Theme]): coverage (Union[Unset, float]): turnover (Union[Unset, float]): user_count (Union[Unset, int]): alpha_count (Union[Unset, int]): pyramid_multiplier (Union[Unset, float]):

```
DataSetDetails(
    region: str,
    delay: int,
    universe: str,
    value_score: float,
    field_count: int,
    themes: List[Theme],
    coverage: Union[brain.types.Unset, float] = <brain.types.Unset object>,
    turnover: Union[brain.types.Unset, float] = <brain.types.Unset object>,
    user_count: Union[brain.types.Unset, int] = <brain.types.Unset object>,
    alpha_count: Union[brain.types.Unset, int] = <brain.types.Unset object>,
    pyramid_multiplier: Union[brain.types.Unset, float] = <brain.types.Unset object>
)
```

```
class DataSubCategory
    DataSubCategory()
    id
    name
    dataset_count
    field_count
    alpha_count
    user_count
    value_score
    region
    additional_properties
    to_dict()
    from_dict()
    additional_keys
```

class DataSubCategory:

Rejects unexpected fields, read only fields, and values of the wrong type

Attributes: id (str): name (str): dataset_count (int): field_count (int): alpha_count (int): user_count (int): value_score (float): region (List[str]):

```
DataSubCategory(
    id: str,
    name: str,
    dataset_count: int,
    field_count: int,
    alpha_count: int,
    user_count: int,
    value_score: float,
    region: List[str]
)
```

```
class DataSummary
    DataSummary()
    id
    name
    additional_properties
    to_dict()
    from_dict()
    additional_keys
```

class DataSummary:

Rejects unexpected fields, read only fields, and values of the wrong type

Attributes: id (str): name (str):

```
DataSummary(id: str, name: str)
```

Method generated by attrs for class DataSummary.

```
class PaginatedDataFieldList
    PaginatedDataFieldList()
    count
    next_
    previous
    results
    additional_properties
    to_dict()
    from_dict()
    additional_keys
```

```
class PaginatedDataSetList
    PaginatedDataSetList()
    count
    next_
    previous
    results
    additional_properties
    to_dict()
    from_dict()
    additional_keys
```

```
class ResearchPaper
    ResearchPaper()
    type
    title
    url
    additional_properties
    to_dict()
    from_dict()
    additional_keys
```

```
class Theme
    Theme()
    id
    name
    multiplier
    additional_properties
    to_dict()
    from_dict()
    additional_keys
```

class PaginatedDataFieldList:

Attributes: count (Union[Unset, int]): Example: 123. next_(Union[None, Unset, str]): previous (Union[None, Unset, str]): results (Union[Unset, List['DataField']]):

PaginatedDataFieldList

```
count: Union[brain.types.Unset, int] = <brain.types.Unset object>,
next_: Union[NoneType, brain.types.Unset, str] = <brain.types.Unset object>,
previous: Union[NoneType, brain.types.Unset, str] = <brain.types.Unset object>,
results: Union[brain.types.Unset, List[DataField]] = <brain.types.Unset object>
)
```

Method generated by attrs for class PaginatedDataFieldList.

class PaginatedDataSetList:

Attributes: count (Union[Unset, int]): Example: 123. next_(Union[None, Unset, str]): previous (Union[None, Unset, str]): results (Union[Unset, List['DataSet']]):

PaginatedDataSetList

```
count: Union[brain.types.Unset, int] = <brain.types.Unset object>,
next_: Union[NoneType, brain.types.Unset, str] = <brain.types.Unset object>,
previous: Union[NoneType, brain.types.Unset, str] = <brain.types.Unset object>,
results: Union[brain.types.Unset, List[DataSet]] = <brain.types.Unset object>
)
```

Method generated by attrs for class PaginatedDataSetList.

class ResearchPaper:

Rejects unexpected fields, read only fields, and values of the wrong type

Attributes: type (str): title (str): url (str):

ResearchPaper(type: str, title: str, url: str)

Method generated by attrs for class ResearchPaper.

type: str

class Theme:

Rejects unexpected fields, read only fields, and values of the wrong type

Attributes: id (str): name (str): multiplier (float):

Theme(id: str, name: str, multiplier: float)

Method generated by attrs for class Theme.

id: str

name: str

The screenshot shows a web-based API documentation interface for the 'brain' library. At the top left is a logo with the text 'brain'. Below it is a search bar with placeholder text 'Search...'. The main content area is titled 'API Documentation' and lists several methods: 'plot_descriptive_stats()', 'plot_instruments_values()', 'plot_values_distribution()', 'plot_groups_average_values()', and 'plot_non_na_counts()'. At the bottom left of this section, it says 'built with pdoc' next to the pdoc logo.

brain.visualizations

A client library for accessing BRAIN

```
def plot_descriptive_stats(df: pandas.core.frame.DataFrame, log_y: bool = False) -> None:
```

Calculate descriptive statistics for each row (excluding 'count') and visualize them as a line plot using Seaborn.

Parameters:

- df: pd.DataFrame, the input DataFrame containing time-series data.
- log_y: bool, whether to use a log scale for the y-axis (default: False).

Returns:

- None, displays the plot.

```
[2]: brain = Brain(region="EUR", delay=1, universe="TOP2500")
print(f"Instrument Type = {brain.instrument_type}, Region = {brain.region}, Universe = {brain.universe}, Delay = {brain.delay}")
Instrument Type = EQUITY, Region = EUR, Universe = TOP2500, Delay = 1
[3]: fields = brain.get_data_fields()
[10]: fields.count
[10]: 10000
```