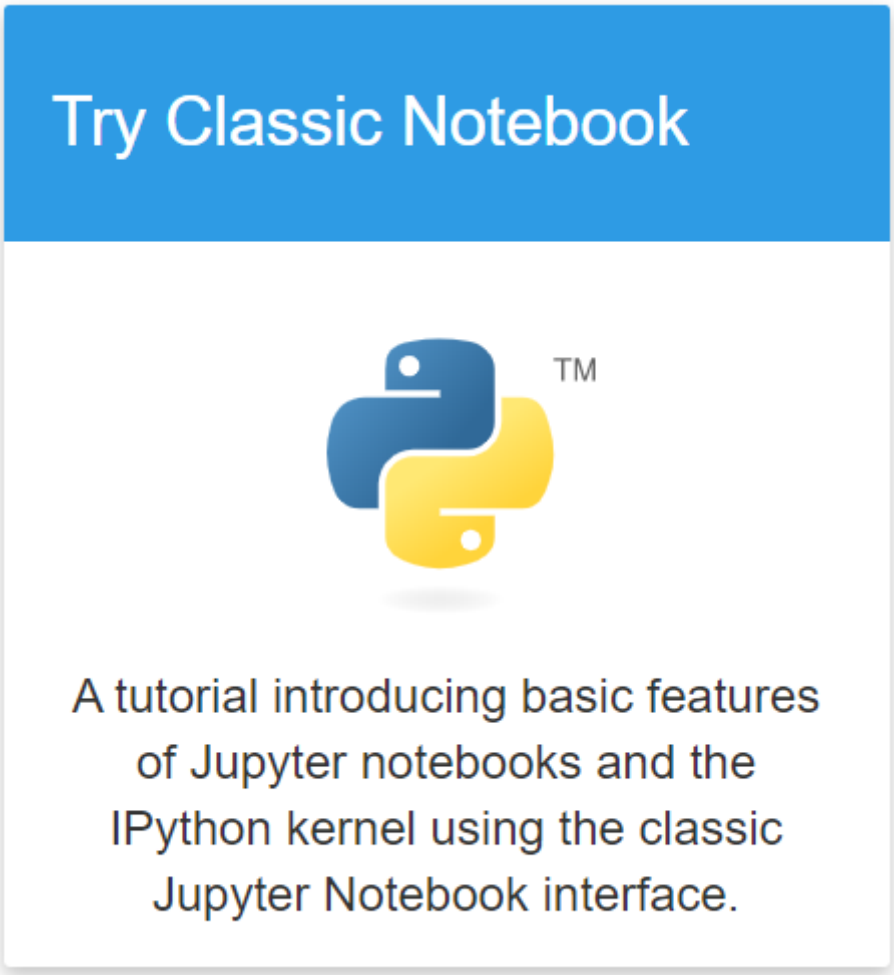
iPlacenta Jupyter Notebook

In the course of the next hour we will look into data types in Python and basic plotting. We will not force you to install anything and give you a web instance instead. The drawback is, that some of the functions do not work in the trial version.

Please go to jupyter.org/try (https://jupyter.org/try)

- Open Classic Notebook



This will direct you to a trial instance of jupyter notebook including some tutorials. We will ignore them for now. Feel free to look at them later though, they are a nice introduction. Let us open a new empty notebook.

- Click on the tab "File" then "Open new" and "Python 3" you should now have an empty notebook. Please do not forget to download it, so you do not lose your progress. You can also upload an existing notebook and open it.

Data Types

Every value in a program has a specific type.

- Integer (int): represents positive or negative whole numbers like 3 or -512.
- Floating point number (float): represents real numbers like 3.14159 or -2.5.
- Character string (usually called “string”, str): text.
  - Written in either single quotes or double quotes (as long as they match).
  - The quote marks aren’t printed when the string is displayed.

Use the built-in function type to find the type of a value.

In [ ]:

```
```

In [ ]:

```
```

Types control what operations (or methods can be performed on a given value.

- A value’s type determines what the program can do to it

In [ ]:

```
```

In [ ]:

```
```

You can use the + and * operators on strings

In [ ]:

```
```

Strings have a length - numbers do not

In [ ]:

```
```

In [ ]:

```
```

You must convert numbers to strings or vice versa when operating on them

In [ ]:

Convert a type into another type using int or str commands

In [ ]:

In [ ]:

Variables only change value when something is assigned to them

In [ ]:

Reading Tabular Data into DataFrames (df)

- import pandas library
- use pandas to load a csv data set
- get basic information about data set

In [ ]:

Use index_col to specify that a column's values should be used as row headings

In [ ]:

Use DataFrame.info to find out more about a dataframe

In [ ]:

Use DataFrame.T to transpose a dataframe

In [ ]:

```
```

Use DataFrame.describe to get summary statistics about data

In [ ]:

```
```

Use DataFrame.iloc[..., ...] to select values by their position

In [ ]:

```
```

In [ ]:

```
```

Use DataFrame.loc[..., ...] to select values by their label

In [ ]:

```
```

Use : on its own to select all columns or all rows

In [ ]:

```
```

In [ ]:

```
```

Select multiple columns or rows using DataFrame.loc and a named slice.

In [ ]:

```
```

In [ ]:

```
```

Intro to Plotting in Python

In [ ]:

In [ ]:

In [ ]:

We can also plot a pandas dataframe In this case we have to convert the column headings from a string to integer data type. We want to extract the last 4 characters of each column name. We can use the strip() (only works on strings).

In [ ]:

In [ ]:

In [ ]:

Many styles of plot are available.

In [ ]:

Plot dashed line

In [ ]:

Scatterplot

In [ ]:

add axis titles

In [ ]:

In [ ]:

```

```

In [ ]:

```

```