**Course: Introduction to Data Science (DS2006) - Laboratory 12**

**Student:**

**Create a new file named** [**LogAnalyser.py**](http://loganalyser.py)**.**

**Task 1: C**reate the structure of a class named LogAnalyser with an \_\_init\_\_ method like the one shown in Figure 1.

**import pandas as pd**

**class LogAnalyser:**

**def \_\_init\_\_(self):**

**self.data = []**

**self.player\_names = []**

**self.df = pd.DataFrame()**

**Figure 1 . Code Snippet.**

**Task 2:** When you look at Figure 1, how many lines of code are new to you (considering what we have seen so far)? List each one of them here.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Continue on the next next page BUT wait until you do these two activities before moving forward!**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Alright! The two new line of codes are:**

**import pandas as pd**

**self.df = pd.DataFrame()**

**Even if you do not know what they are exactly (yet), you should be able by now to know what is going on with the code without doing anything about it.**

**Task 3:** Explain what is happening here: **import pandas as pd**

**Task 4:** Explain what is happening here: **self.df = pd.DataFrame()**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Continue on the next next page BUT wait until you do these two activities before moving forward!**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Alrighty, based on what we have seen so far, your answers should be something like:**

**\*** It is importing a package named **pandas** and making an alias to use as **pd** in the code.

\* It is creating an attribute named df that is an instance/object of the class **pd.DataFrame**.

A Pandas DataFrame can be used as a 2-dimensional labeled data structure with columns of potentially different types. You can think of it like a spreadsheet. DataFrames are the most commonly used Pandas object and they are able to:

* **Handle tabular data:** Perfect for data that can be organized into rows and columns.
* **Perform Data manipulation:** Pandas provides powerful tools for filtering, cleaning, transforming, and analyzing data.
* **Perform Data analysis:** Many statistical and analytical operations can be performed directly on DataFrames.
* **Play nice with others:** i.e. there is a good integration with other libraries.

**Task 5:** In order to better understand the basic functionalities of Pandas Data Frames we will create a file named [results.txt](http://results.txt.py) with the exact content shown in Figure 2. Note that Round 1: … needs to be in the first line of the file.

Round 1: a rolled 3, b rolled 1, c rolled 6

Round 2: a rolled 6, b rolled 1, c rolled 3

Round 3: a rolled 2, b rolled 2, c rolled 6

Round 4: a rolled 6, b rolled 6, c rolled 1

Round 5: a rolled 4, b rolled 5, c rolled 1

Round 6: a rolled 6, b rolled 4, c rolled 3

Figure 2 - Example of a Battle of Dices log.

**Task 6:** Now we need to import that data from Battle of Dices into a pandas Dataframe. It would be a good idea to do that as a method in the LogAnalyser class. Create a method (you can choose the name) that will receive an open parameter with the filename that has the log information we want to open. You can use the code from Figure 3 inside your method.

with open(filename, "r") as f:

lines = f.readlines()

for line in lines:

# Example line: "Round 1: a rolled 1, b rolled 2, c rolled 3"

try:

round\_part, rolls\_part = line.strip().split(":", 1)

except ValueError:

continue # skip malformed lines

rolls\_dict = {}

for part in rolls\_part.split(","):

part = part.strip() # remove spaces

if " rolled " in part:

name, \_, roll = part.partition(" rolled ")

rolls\_dict[name] = int(roll)

if name not in self.player\_names:

self.player\_names.append(name)

self.data.append(rolls\_dict)

# Convert list of dicts to DataFrame

self.df = pd.DataFrame(self.data, columns=self.player\_names)

self.df.index += 1 # rounds starting from 1

self.df.index.name = "Round"

Figure 3 - Code snippet for Opening a Battle of Dices log.

**Task 7:** Before we go into more detail about what is happening in that code (we will come back to details in it in our next lecture), let's try to run it. For that you will need to:

1. Write the code to create an object of the class LogAnalyser.

2. Call the method you created to load files passing as a parameter the file we created in **task 5**, that is “**results.txt**”.

3. Print the attribute df.

The output should be similar to the image shown in Figure 4.

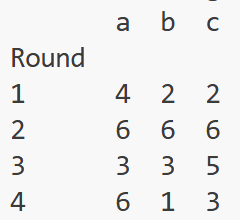


Figure 4 - Example of printing the DataFrame created from results.txt

**mylog = LogAnalyser()**

**mylog.load\_bod\_file("results.txt")**

Figure 5 - Code snippet

Now lets try some common and useful commands using pandas that can be used to better explore the data. Assuming that you have a code sequence similar to the one shown in Figure 5 (to create an object of the LogAnalyser class and to call the method that loads the results.txt file), what happens when you use:

**Task 8: print(mylog.df.head(2))**

**Task 9: print(mylog.df.info())**

**Task 10: print(mylog.df.describe())**

**Task 11: print(mylog.df.shape)**