**Exercise 9: Analyze Courses**

**Assignment Specification**

**Description**: This program will read a pre-processed outcome from Workday with SSE courses for Spring 22 and extract information about their content.

**The dataset**. REG\_CAP\_Spring22\_B.csv contains 94 rows composed by:

* Academic Units
* Section Status
* Section Listings
* Title
* Meeting Patterns
* Building/Room
* Campus Locations
* Instructor(s)/Teaching Assistant
* Credit Hour
* Enrollment Count
* Section Capacity
* Reserved Capacity
* Reserved Seats
* Waitlist Count
* Academic Period
* Start Date
* End Date
* Instructional Format
* Delivery Mode
* Public Notes
* Clustered Course Sections
* Academic Level
* Grading Bases
* Academic Period.1
* Course Definition
* Reference ID
* Level
* Program

**Input**: No user provided input. Data will be from the file REG\_CAP\_Spring22\_B.csv

**Output**:

1. Print the 5 courses with the highest number of students (from "Enrollment Count")
2. Print the 5 instructors (from "Instructor(s)/Teaching Assistant") with the highest number of students (from "Enrollment Count")
3. Compare the total number of students (from "Enrollment Count") for "UG", "G", "Corp" (from "Level"). Comparing means calculate the values and describe the results in the narrative part of the assignment
4. Compare the number of courses that run at full capacity (marked as "Closed") with those that were not ("Open"). Comparing means calculate the values and describe the results in the narrative part of the assignment
5. Create a pie chart with the distribution of students (from "Enrollment Count") per program (from "Program")
6. Create a pie chart with the distribution of students (from "Enrollment Count") per type of delivery (In-Person/Online, from "Delivery Mode")
7. Perform any other analysis that could make sense to better monitor the semester.

See details in the Procedure.

**Procedure**:

1. Read the input file
2. Perform the analysis required as "Output"
3. Submit your .py file and an interpretation report.

Please note:

* The code MUST be well commented for each functional group of lines (e.g.: a basic loop is a group; a function needs to be commented as entire function and inside it for its functional groups). Comments need to explain **clearly** what the statement is doing. **Lack or not enough comments will lead to points reduction. Non original comments will be considered null**
* The interpretation document **has to be original**. **Non original comments will be considered null.**