

## IS324 - Spring 2022

### Lab – 5

### OOP & Logging

#### **LAB OBJECTIVES:**

- Understand the concept of objects.
- Develop python applications in the OOP paradigm by creating objects and methods.
- Logging and tracking events that happen when the python applications runs

#### **Exercise 1:**

The UML below represents the class Apartment, its attributes, and methods. Write a python program for this UML class.

Apartment
RoomsNum Parking Rented Price
__init__ (self, ro, pa, re, pr) __str__ (self): str

#### **Exercise 2:**

Open a new python file named with your student ID and import the class you have created in **Exercise 1** and Create 7 instances of Apartment class with the following information:

Instance 1 : (RoomsNum = 3, Parking = False, Rented = False, Price = 1000)

Instance 2 : (RoomsNum = 5, Parking = True, Rented = True, Price = 2000)

Instance 3 : (RoomsNum = 3, Parking = True, Rented = False, Price = 1200)

Instance 4 : (RoomsNum = 2, Parking = False, Rented = True, Price = 800)

Instance 5 : (RoomsNum = 4, Parking = True, Rented = False, Price = 1700)

Instance 6 : (RoomsNum = 3, Parking = False, Rented = False, Price = 1000)

Instance 7 : (RoomsNum = 4, Parking = False, Rented = True, Price = 1500)

### Exercise 3:

With the same python file you have created in **Exercise 2**, create an empty list and add all the previous instances to this list.

### Exercise 4:

Write the following functions in the same python file you have created in **Exercise 2**:

**A) num\_Available(self, list):** that accepts a list of apartments and returns how many apartments are available (not rented).

#### Testing:

*If I call this method with the list you created in **Exercise 3**, the output should be 4 as there are 4 apartments which are not rented yet.*

**B) num\_Apartment\_Rooms(self, list, rNum):** that accepts a list of apartments, rNum (int), and returns how many apartments that have this number of rooms.

#### Testing:

*If I call this method with the list you created in **Exercise 3** and rNum=4, the output should be 2 as there are 2 apartments which have 4 rooms.*

**C) apartment\_Search(self, list, park, price):** that accepts a list of apartments, park (boolean), price (int), and print out the info for all apartments that meets the parking condition and has a rental price less than the specified price.

#### Testing:

*If I call this method with the list you created in **Exercise 3** and parking=True, price=1400, the output should be: (RoomsNum = 3, Parking = True, Rented = False, Price = 1200).*

### Exercise 5:

Import and use logging module in your python file and let the log file named as (apartmentLog.log) and log out the events that will happen after the program runs showing at least two different levels of log messages in the log file.

Sample:

```
ERROR - 2022-03-01 01:19:19,307 - root - Apartment already rented
INFO - 2022-03-01 01:19:19,307 - root - Apartment rented successfully
```