LABORATORY 8

REQUIREMENT

For the program you developed for Lab5-7 implement persistent storage for all entities using file-based repositories. Also implement a **settings.properties** file to configure your application.

Observations:

1. You must implement **two** additional repository sets: one using text files for storage, and one using binary files (e.g. using object serialization with Pickle).
2. The program must work the same way using ***in-memory*** repositories, **text file** repositories and **binary file** repositories.
3. The decision of which repositories are employed, as well as the location of the repository input files will be made in the program's settings.properties file. An example is below:

    a. settings.properties for loading from memory (input file are not required):

    repository = inmemory
    cars = ""
    clients = ""
    rentals = ""

    b. settings.properties for loaging from binary files, for someone who also created a GUI:

    repository = binaryfiles
    cars = "*cars.pickle*"
    clients = "*clients.pickle*"
    rentals = "*rentals.pickle*"
    ui = "*GUI*"

NB! If your Lab5-7 uses layered architecture properly, these are the only places where source code needs to change:

1. Repository layer – for implementing the required code.
2. AppStart – to load the properties file and start the required repositories.

BONUS POSSIBILITY (0.1P)

- In addition to the file-based implementations above, implement the repository layer to use JSON or XML files for storage (at your choice).
- Create a Settings class into which you load the data from the settings.properties file. Then, the AppStart module decides which modules are started by examining the Settings object. This further decouples the properties input file from the application. Deadline is **week 11.**

BONUS POSSIBILITY (0.2P)

- Implement an SQL-backed repository. Deadline is **week 12.**