Introduction in .Net Core -> Lab 3 (Florin Olariu && Dan Nastasa)

Prerequisites:

a) Create a Blank solution
b) Add a Class Library (.NET Core)
c) Add a Unit Test Project (.NET Core)
d) Add dependency between Test Project and Class Library

Note: The exercise is meant to learn how to build domain models and it we help us to:

- Learn how to design classes
- How to apply aggregation
- How to use/apply encapsulation
- How to use/apply inheritance
- How the Repository Pattern works
- How to manipulate collections

Exercise:

1. Create a class called **Car** and expose the following properties:
    a. **Id**
    Color
    HorsePower
    NrOfDoors
    Consumption(liters per km)
    TrunkCapacity
    b. Create a class called **Motorbike** and expose the properties:
    Id
    Color
    HorsePower
    Consumption(liters per km)
    Model
    c. Expose the following behaviour, for both classes:
    ComputeFuelConsumption(int distance)
    StartEngine(virtual or abstract?) => should return **"Starting" + Car or "Starting" + Motorbike.**
    d. Using the inheritance principle extract a base class called **Vehicle**.
    e. Explain the option for the StartEngine method
    f. Create unit tests to have 100% code coverage for all the classes.
2. Create Record class and expose the following:
    a. **Id**
    Title
    StartDate
    EndDate
    Value
    b. Create Class **RecordRepository**
    c. Populate a List of records via constructor(minimum 3 records)
    d. Expose and test the following behaviour

i. **GetRecordByTitle**(string recordTitle)
ii. **FindFutureRecords**()
iii. **AddRecord**(Record record)
iv. **GetRecordByPosition**(int position)
v. **RemoveRecordsWithTitle**(string recordTitle)
vi. **GetExpiredRecords**()
vii. Create unit tests to have 100% code coverage.

Note:

1. All exercises are mandatory.
2. You will receive your points at the end of the lab.