ESMAD | TSIW | POO
Exercise Sheet nº 7
Classes

Open Visual Studio Code and solve the following exercises:

1. **Class definition (files ex1.html/ex1.js)**

   Dice are small polyhedra engraved with specific instructions. The most classic data is the cube (six faces) carved with numbers from one to six.

   a. Create a **Dice** class to represent a dice that will allow us to roll it and take values ranging from 1 to 6, a six-sided dice. The class must have a set of private properties such as faceValue (value of each face) and quantity (number of polyhedron faces).
   b. Define a no-argument constructor that creates an object of the **Dice** class with six faces (cube).
   c. Define the set and get methods for the **faceValue** private property that will store the current face value of the dice.
   d. Define the following methods:
      a. A method called **getQuantityFaces** that returns a number representing the number of faces in the dice.
      b. A **roll** method that will "roll" the dice and store the result in the **faceValue** property. The drawn value must be in the range of 1 up to the number of faces of the dice.
   e. Find a way to roll the dice 100 times and present a frequency table similar to the one below.

Face	1	2	3	4	5	6
Frequency	10	18	17	21	15	19

2. **Class definition (files ex2.html/ex2.js)**

   Define a **Car** class to represent a car.

   a. A car comprises a brand, license plate, color, current and maximum fuel tank (in liters), consumption (at 100km), and fuel type. Set all properties to **private**.
   b. Create a constructor that receives all these parameters and initializes the respective properties.
   c. Define access (**get**) and modifier (**set**) methods for all properties with particular attention to the following:
      a. **color**: get method only
      b. **current fuel tank**: value greater than 0 and less than the maximum deposit
      c. **consumption**: value greater than 0
      d. **fuel type**: Gasoline or Diesel
   d. Add three cars (brand, plate, color, current fuel, maximum fuel, consumption, fuel type):
      a. “Ford”, ”91-GH-15”, “Green”, 40, 60, 5, “Diesel”
      b. “Opel”, ”23-AB-23”, “White”, 50, 55, 6.5, “Gasoline”
      c. “Nissan”, ”12-CX-45”, “Black”, 22, 70, 4.5, “Diesel”
   e. Create the following methods:
      a. Method to fill the tank. The number of liters supplied must be used as a parameter.
      b. A method that receives the number of kilometers traveled and that changes the number of liters considering the base consumption of the vehicle.
   f. Add the objects to an array of **cars.**
   g. Create the following functions:
      a. A function that returns the number of cars of a given brand passed as a parameter.
      b. A function that, given a type of fuel, returns the sum of liters of cars that have that type of fuel.

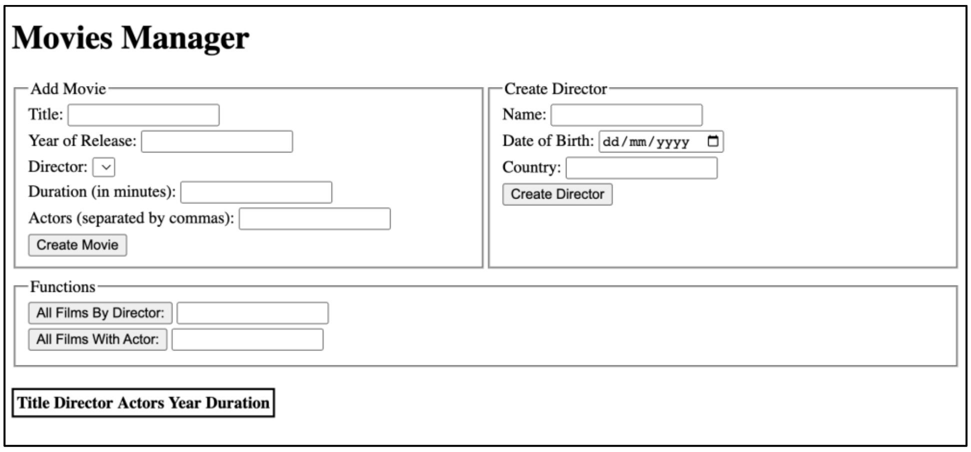
3. **App making use of classes (files ex3.html/ex3.js)**

Make an app that manages cinema films and their directors.

- A film is characterized by a title, year of release, director, duration (in minutes), and actors (text box with names separated by commas).
- A director is characterized by a name, date of birth and country. Assume that a film can only have one director.

a. Create a Movie class and a director class with the above-mentioned properties.
b. Create the UI with the following elements:
    a. Form to create a movie.
    b. Form to create a director (will feed a select in the main movie form).
    c. Table for rendering the films (title, director, actors, year and duration).
c. Create functions that allow you to get:
    a. All films by a given director (for simplicity console log all movies).
    b. All the titles of the films where a particular actor/actress enters (for simplicity console log all movies).

You must create elements in the UI that allow you to invoke the above functions.



Movies Manager

Add Movie
Title:
Year of Release:
Director:
Duration (in minutes):
Actors (separated by commas):
Create Movie

Create Director
Name:
Date of Birth: dd/mm/yyyy
Country:
Create Director

Functions
All Films By Director:
All Films With Actor:

Title	Director	Actors	Year	Duration