UNIVERSITATEA TEHNICĂ
A MOLDOVEI

FAF.OOP22.1

Laboratory work: #0

GIT & Intro to Classes

2023

Homework N° 0

Assignment

The assignment's goal is to make sure, you, have all the prerequisites necessary to start working on the upcoming laboratories: ***Git****,* ***Github account****,* ***Integrated Development Environment (IDE)***

Task: Follow the directions

Follow the instructions, but feel free to deviate if you know what you're doing. Good luck!

1. Install Git - Go to the git install page;
2. Create a github account in Gihub;
3. Create a class in a language and editor of your choice (Java, Python, C++, C#) that:
   - Stores information about the object (to represent an object's state - moving/standing, it's speed or color);
   - Has a method for informing the user of it's state (printing it's state to the console);

DEADLINE: End of the class.

**You don't have to use Git or Github in this lab.*

Documentation on the next page.

About Git

Git is a "**Version Control System**" (VCS) that records changes in files – which could be images, layouts, and most importantly, it records changes in *your* code files. Through "version control", Git enables you to branch out to different versions of your code or revert to any previously recorded state. It is incredibly useful and an essential concept for programmers to comprehend.

Git is the **tool** you're really going to use consistently.

References:

- first time Git configuration (name, email);
- Practical straight to the point Git use.

About Github

Github is not a direct extension of Git, they are separate solutions to solve different problems. However, git and github can totally be used together.

Github is a hosting server for code repositories, being the largest one and having one of the most collaborative environment - millions of open source projects (Git, Visual Studio Code, React source code) that use it for code review, tracking updates, features and even bugs (issues) and other things like discussions.

It is also used in small teams, with private repositories and by solo developers to host their repositories in the cloud, so they can work on their projects remotely from anywhere and any computer and keep track of their work.

References:

- Useful (not mandatory) guide to setup a local SSH key for when you will work with Github, and how to add it to your github account for passwordless git - github operations.

Creating a Class

Classes are a collection of functions and variables to describe an Object state and an Object's behavior.

When we are working with classes and objects functions become methods and variables become fields/attributes.

Here's a few simple examples of types of variables and their definition, method definition and class definition in Java, C++ and Python. With their help, you are to define your own classes.

JAVA

Types: int, long, float, double, boolean, char, String

Field definition example:

```
char grade = 'A';
boolean isNoon = false;
int[] = {9, 2, 19};
String group = "OOP"
```

Class definition example:

```
public class Car {
    float speed = 0;
    boolean isMoving = false;

    public void stop() {
        isMoving = false;
    }

    public void accelerate(float amount) {
        speed += amount;
        isMoving = speed != 0;
    }

    public void printSpeed() {
        System.out.println("Speed: " + speed);
    }

    public void printIsMoving() {
        if(isMoving) {
            System.out.println("The car is moving");
        } else {
            System.out.println("The car is not moving");
        }
    }
}
```

Types: int, long, float, double, bool, char, string
Variable definition example:

```
char grade = ’A’;
bool isNoon = false;
int arr[] = {9, 2, 19};
std::string group = "OOP";
```

Class definition example:

```
class Car {
private:
    float speed = 0;
    bool isMoving = false;

public:
    void stop() {
        isMoving = false;
    }

    void accelerate(float amount) {
        speed += amount;
        isMoving = speed != 0;
    }

    void printSpeed() {
        std::cout << "Speed: " << speed << std::endl;
    }

    void printIsMoving() {
        if (isMoving) {
            std::cout << "The car is moving" << std::endl;
        } else {
            std::cout << "The car is not moving" << std::endl;
        }
    }
};
```

Python

Types: int, float, bool, str
Variable definition example:

```python
grade = ’A’
isNoon = False
arr = [9, 2, 19]
group = "OOP"
```

Class definition example:

```python
class Car:
    def __init__(self):
        self.speed = 0
        self.isMoving = False

    def stop(self):
        self.isMoving = False

    def accelerate(self, amount):
        self.speed += amount
        self.isMoving = self.speed != 0

    def printSpeed(self):
        print("Speed:", self.speed)

    def printIsMoving(self):
        if self.isMoving:
            print("The car is moving")
        else:
            print("The car is not moving")
```