

Практична робота №1

Тема: Система керування версіями Git.

Мета: Закріпити знання про базові поняття класу, поняття про конструктори і деструктори, принципи роботи із класами.

Хід роботи

1. Переглядаю лекцію курсу CS50, а також ознайомлююсь із додатковою літературою та теоретичними відомостями.
2. Проходжу основний рівень інтерактивного курсу Learn Git Branching.
3. Створюю новий консольний C++-проект.
4. Запускаю у папці проекту Git Bash. Ініціалізую у папці проекту git-репозиторій (git init).
5. Додаю у проєку клас згідно варіанту 8:

8	Car	id, модель, рік випуску, ціна, реєстраційний номер, vin-номер, кількість пасажирських місць, кількість дверей.
---	-----	--

6. Роблю коміт проєкта (git commit).
7. Додаю до класу конструктори (за умовчуванням, параметризований та копіювання) і деструктор.
8. Роблю коміт проєкту (git commit).
9. Оформляю звіт.

Скріншот виконання команди git log:

```
MINGW64:/b/College/Practice/Lab1/WorkWithGit/WorkWithGit
WorkWithGit.vcxproj.filters
WorkWithGit.vcxproj.user

no changes added to commit (use "git add" and/or "git commit -a")

Nazar@DESKTOP-FK60VO7 MINGW64 /b/College/Practice/Lab1/WorkWithGit/WorkWithGit (master)
$ git commit -am "Add constructors and destructors"
[master e3e122e] Add constructors and destructors
 2 files changed, 41 insertions(+)

Nazar@DESKTOP-FK60VO7 MINGW64 /b/College/Practice/Lab1/WorkWithGit/WorkWithGit (master)
$ git log
commit e3e122e09e941d2efd70aed56eb4e59455f167c4 (HEAD -> master)
Author: Lekantrop <98541354+Lekantrop-gd@users.noreply.github.com>
Date:   Fri Oct 27 12:19:53 2023 +0300

    Add constructors and destructors

commit 8251806d4645615a3f0938a005e6487551b212e2
Author: Lekantrop <98541354+Lekantrop-gd@users.noreply.github.com>
Date:   Fri Oct 27 12:13:20 2023 +0300

    Initial commit

Nazar@DESKTOP-FK60VO7 MINGW64 /b/College/Practice/Lab1/WorkWithGit/WorkWithGit (master)
$
```

Вихідний код:

Файл Car.h:

```
#pragma once
#include<iostream>
using namespace std;

class Car
{
private:
int id;
string model;
int price;
string registrationNumber;
string vinCode;
int numberOfSeats;
int numberOfDoors;

public:
Car() = default;
Car(int id, const string& model, int price, const string& registrationNumber,
const string& vinCode, int numberOfSeats, int numberOfDoors);
Car(const Car &car);
~Car();
};
```

Файл Car.cpp

```
#include "Car.h"

Car::Car(
int id,
const string& model,
int price,
const string& registrationNumber,
const string& vinCode,
int numberOfSeats,
int numberOfDoors)
:
id(id),
model(model),
price(price),
registrationNumber(registrationNumber),
vinCode(vinCode),
numberOfSeats(numberOfSeats),
numberOfDoors(numberOfDoors)
{
}

Car::Car(const Car& car)
{
```

```
this->id = car.id;  
this->model = car.model;  
this->price = car.price;  
this->registrationNumber = car.registrationNumber;  
this->vinCode = car.vinCode;  
this->numberOfSeats = car.numberOfSeats;  
this->numberOfDoors = car.numberOfDoors;  
}
```

```
Car::~~Car()  
{  
}
```

Висновок: під час виконання даної практичної роботи, закріплено знання про базові поняття класу, поняття про конструктори і деструктори, принципи роботи із класами.