



ЗВІТ
з лабораторної роботи №2
з дисципліни 'Кросплатформні засоби програмування'

Виконала:
Ст.гр. КІ-303
Онишкевич Тетяна
Перевірів:
Майдан М.В

Львів - 2023

КЛАСИ ТА ПАКЕТИ

Завдання:

1. Написати та налагодити програму на мові Java, що реалізує у вигляді класу предметну область згідно варіанту. Програма має задовольняти наступним вимогам:
 - програма має розміщуватися в пакеті `Група.Прізвище.Lab2`;
 - клас має містити мінімум 3 поля, що є об'єктами класів, які описують складові частини предметної області;
 - клас має містити кілька конструкторів та мінімум 10 методів;
 - для тестування і демонстрації роботи розробленого класу розробити клас-драйвер;
 - методи класу мають вести протокол своєї діяльності, що записується у файл;
 - розробити механізм коректного завершення роботи з файлом (не надіятися на метод `finalize()`);
 - програма має володіти коментарями, які дозволять автоматично згенерувати документацію до розробленого пакету.
2. Автоматично згенерувати документацію до розробленої програми.
3. Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.
4. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації та завантажити його у ВНС.
5. Дати відповідь на контрольні запитання.

Варіант:

14. Телевізор

Код програми:

```
class TV:
    def __init__(self, brand: str, price: float,
screen_size: int):
        self.COLOUR = ("neutral", "warm", "white",
"cool", "native")
        self.STATES = ("OFF", "ON")
        self.BASE_CHANNEL = 0
        self.BASE_VOLUME = 50

        self.brand = brand
        self.price = price
        self.screen_size = screen_size

        self.is_on = True
        self.volume = self.BASE_VOLUME
        self.scheme = self.COLOUR[0]
        self.current_channel = self.BASE_CHANNEL
```

```

        def tv_state(self) -> None:
            print(f"Turning off the {self.brand} TV...")
            if self.is_on else print(f"Turning on the {self.brand} TV...")

            self.is_on = not self.is_on

        def set_colour_scheme(self, scheme: str) -> None:
            self.scheme = scheme if scheme in self.COLOUR
        else self.COLOUR[0]

        def chanel_switch(self, channel: int) -> None:
            self.current_channel = channel if self.is_on
        else self.BASE_CHANNEL

            print(f"Changed to channel
{self.current_channel}")

        def volume_control(self, volume: int) -> None:
            self.volume = volume if volume in range(100)
        else self.BASE_VOLUME

            print(f"Changed on {self.volume} dB")

        def __str__(self) -> str:
            return (f"Model: {self.brand}\n"
                    f"Price: {self.price} $\n"
                    f"Screen size: {self.screen_size}
Inch \n"

                    f"Power: {self.STATES[self.is_on]}\n"
                    f"Channel: {self.current_channel}\n"
                    f"Volume: {self.volume}")

    def main():
        tv = TV("Samsung", 129.99, 55)
        tv.volume_control(90)
        tv.set_colour_scheme("warm")
        tv.chanel_switch(5)
        print(tv)

    if __name__ == "__main__":
        main()

```

[Посилання на Github](#)

Результат:

```
Changed on 90 dB
Changed to channel 5
Model: Samsung
Price: 129.99 $
Screen size: 55 Inch
Power: ON
Channel: 5
Volume: 90

Process finished with exit code 0
|
```