

**МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ
КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Ордена Трудового Красного Знамени федеральное государственное бюджетное
образовательное учреждение высшего образования**

«Московский технический университет связи и информатики»

Кафедра “Математическая кибернетика и информационные технологии”

ОТЧЕТ

по лабораторной работе №6

по дисциплине «Введение в информационные технологии»

Тема: «Работа с классами ч.2»

Выполнил: студент группы БВТ2505

Коротков Артём Сергеевич

Проверил: Павликов. А.Е.

Москва, 2025

Цель работы:

Получить практический опыт работы с ООП в Python. Использование инкапсуляции, наследования.

Задание:

Задание 1:

Защита данных пользователя

1. Создайте класс `UserAccount`, который представляет аккаунт пользователя с атрибутами: имя пользователя (`username`), электронная почта (`email`) и приватный атрибут пароль (`password`).
2. Используйте конструктор `__init__` для инициализации этих атрибутов.
3. Реализуйте метод `set_password(new_password)`, который позволяет безопасно изменить пароль аккаунта.
4. Реализуйте метод `check_password(password)`, который проверяет, соответствует ли введённый пароль текущему паролю аккаунта и возвращает `True` или `False`.
5. Создайте объект класса `UserAccount`, попробуйте изменить пароль и проверить его с помощью методов `set_password` и `check_password`.

Задание 2:

Полиморфизм и наследование

1. Определите базовый класс `Vehicle` с атрибутами: `make` (марка) и `model` (модель), а также методом `get_info()`, который возвращает информацию о транспортном средстве.
2. Создайте класс `Car`, наследующий от `Vehicle`, и добавьте в него атрибут `fuel_type` (тип топлива). Переопределите метод `get_info()` таким образом, чтобы он включал информацию о типе топлива.

Скриншоты выполнения:

Задача 1:

```
Задайте новый пароль: qwerty
Введите пароль для проверки: qwerty
Пароль верный
Username: Valya, Email: dot@gmail.com Password: qwerty
```

Задача 2:

```
Базовый класс Vehicle:
Марка: lamborghini, Модель: Countach
Марка: Honda, Модель: Cr-v, Тип топлива: бензин
```

Исходный код программы:

Задача 1:

```
class UserAccount:
```

```

def __init__(self, username, email, password):
    self.username = username
    self.email = email
    self.__password = password
def set_password(self, new_password):
    self.__password = new_password
    return f"Пароль успешно изменен"
def check_password(self, password):
    return password == self.__password
def get_account_info(self):
    return f"Username: {self.username}, Email: {self.email} Password: {self.__password}"
user1 = UserAccount("Valya", "dot@gmail.com", "qwerty123")
password_input = input("Задайте новый пароль: ")
user1.set_password(password_input)
user_input = input("Введите пароль для проверки: ")
if user1.check_password(user_input):
    print("Пароль верный")
else:
    print("Неверный пароль")
print(user1.get_account_info())
print(user1.__password)

```

Задача 2:

```

class Vehicle:
    def init(self, make, model):
        self.make = make
        self.model = model
    def get_info(self):
        return f"Марка: {self.make}, Модель: {self.model}"
class Car(Vehicle):
    def init(self, make, model, fuel_type):
        super().init(make, model)

```

```
        self.fuel_type = fuel_type
    def get_info(self):
        base_info = super().get_info()
        return f"{base_info}, Тип топлива: {self.fuel_type}"
vehicle = Vehicle("lamborghini", "Countach")
car1 = Car("Honda", "Cr-v", "бензин")
car2 = Car("Tesla", "Cybertruck", "электричество")
car3 = Car("Ford", "F-250", "дизель")
print("Базовый класс Vehicle:")
print(vehicle.get_info())
print(car1.get_info())
```

Заключение

В ходе выполнения лабораторной работы были успешно решены следующие задачи:

1. Получен практический опыт работы с ООП в Python.
2. Изучены такие принципы как инкапсуляция и наследование.