

Задание на Life Coding: Основы работы с классами: инкапсуляция, наследование, полиморфизм.

Задача 1:

Создайте базовый класс **Device** с полями:

- string Model;
- DeviceStatus Status; (DeviceStatus с значениями: On, Off)

Добавьте методы, которые переключают DeviceStatus и пишут сообщения "{Model} включен\выключен":

```
void TurnOn();
void TurnOff();
```

Создайте классы-наследники **Computer**, **Printer**. Переопределите методы TurnOn/TurnOff (Компьютер {Model} включен\выключен)

Задача 2:

Создайте базовый класс **Employee** с полями:

- string Name
- readonly string Department (отдел)
- decimal Salary

И методом PrintInfo()

Создайте классы-наследники **Manager** - добавляет поле int TeamSize (размер команды), **Developer** - добавляет поле string ProgrammingLanguage

Переопределите метод PrintInfo() в каждом наследнике таким образом, чтобы они возвращали информацию именно о данном классе.

"Сотрудник: {Name}, Отдел: {Department}, Зарплата: {Salary}"

"Менеджер: {Name}, Команда: {TeamSize} человек, Зарплата: {Salary}"

"Разработчик: {Name}, Язык программирования: {ProgrammingLanguage}, Зарплата: {Salary}"

Задача 3:

Создайте базовый класс **Vehicle** с полями:

- string Brand (марка)
- int MaxSpeed (максимальная скорость)
- int FuelLevel (уровень топлива)

Добавьте метод CheckStatus(), который выводит информацию транспорте.
"*{Brand}*: скорость до {MaxSpeed} км/ч, топливо {FuelLevel}%".

Создайте два класса-наследника:

Car - добавляет поле int Doors (количество дверей) и bool HasAirConditioning;

Motorcycle - добавляет поле bool HasSidecar (есть ли коляска) и string HelmetColor

Переопределите метод CheckStatus() в каждом классе-наследнике.

"*{Brand}*: {Doors} дверей, {"с кондиционером" \ "без кондиционера"}, топливо {FuelLevel}%"

"*{Brand}*: {"с коляской" \ "без коляски"}, шлем {HelmetColor}, топливо {FuelLevel}%"

Задача 4:

Создайте базовый класс **Animal** с полями:

- string Name
- AnimalType Type (тип животного с значениями: Mammal, Bird, Fish)
- int Age
- double Weight

Добавьте метод void Describe(), который выводит информацию о животном
"*{Name}*: {Type}, возраст {Age} лет, вес {Weight} кг".

Создайте классы-наследники:

Dog - добавляет поле string Breed(порода), bool IsTrained (обученный),

Cat - добавляет поле string FurColor (цвет меха)

Переопределите метод Describe().

"*{Breed}* {Name}: {"дрессированная" \ "недрессированная"}, возраст {Age} лет, вес {Weight} кг"
"*{FurColor}* кот {Name}: , возраст {Age} лет, вес {Weight} кг"