

Лабораторная работа № 1.

Тема: Основы ООП.

Цель: закрепить теоретические знания и применить их на практике.

Задание:

Вариант №1.

Задание 1. Класс и объект.

Создайте класс Car с полями Brand (марка автомобиля), Model (модель) и Year (год выпуска). Добавьте метод DisplayInfo(), который выводит информацию об автомобиле в консоль. Создайте объект класса Car и вызовите метод DisplayInfo().

Задание 2. Поля, свойства и модификаторы доступа.

Создайте класс BankAccount с приватными полями accountNumber (номер счета) и balance (баланс). Реализуйте свойства для доступа к этим полям:

Для accountNumber создайте только геттер (только для чтения).

Для balance создайте геттер и сеттер, но убедитесь, что баланс не может быть отрицательным.

Добавьте метод Deposit(decimal amount) для пополнения счета и метод Withdraw(decimal amount) для снятия средств. Создайте объект класса и протестируйте его работу.

Задание 3. Конструкторы.

Создайте класс Person с полями Name, Age и Email. Реализуйте два конструктора: Конструктор по умолчанию, который инициализирует пустые значения.

Конструктор с параметрами для инициализации всех полей.

Добавьте метод PrintInfo(), который выводит информацию о человеке.

Протестируйте оба конструктора.

Задание 4. Инкапсуляция.

Создайте класс Employee с закрытыми полями salary (зарплата) и bonus (премия).

Реализуйте метод CalculateTotalSalary(), который возвращает сумму зарплаты и премии. Добавьте свойство для установки премии, но убедитесь, что она не может быть отрицательной. Создайте объект класса и выведите общую зарплату.

Задание 5. Наследование и полиморфизм.

Создайте базовый класс Shape с методом Draw(), который выводит "Рисуем фигуру". Создайте два производных класса Circle и Rectangle, которые переопределяют метод Draw() для вывода "Рисуем круг" и "Рисуем прямоугольник" соответственно.

Переопределите метод CalculateArea() для каждого класса.

Вариант 2

Задание 1. Класс и объект.

Создайте класс Book с полями Title (название книги), Author (автор) и Pages (количество страниц). Добавьте метод GetSummary(), который возвращает строку с

краткой информацией о книге. Создайте объект класса Book и вызовите метод GetSummary().

Задание 2. Поля, свойства и модификаторы доступа

Создайте класс Student с закрытыми полями name (имя студента) и grades (список оценок). Реализуйте свойства:

Для name создайте геттер и сеттер.

Для grades создайте только геттер.

Добавьте метод AddGrade(int grade) для добавления оценки в список и метод GetAverageGrade(), который возвращает среднюю оценку. Убедитесь, что оценки находятся в диапазоне от 1 до 10. Протестируйте класс.

Задание 3. Конструкторы.

Создайте класс Product с полями Name, Price и Quantity. Реализуйте три конструктора:

Конструктор по умолчанию, который инициализирует пустые значения.

Конструктор с параметрами для инициализации Name и Price.

Конструктор с параметрами для инициализации всех полей.

Добавьте метод PrintDetails(), который выводит информацию о продукте.

Протестируйте все конструкторы.

Задание 4. Инкапсуляция.

Создайте класс Bank с закрытым полем totalFunds (общие средства банка).

Реализуйте методы:

DepositFunds(decimal amount) для пополнения средств.

WithdrawFunds(decimal amount) для снятия средств.

Убедитесь, что средства не могут быть сняты, если их недостаточно. Добавьте свойство для получения текущего баланса. Создайте объект класса и протестируйте его работу.

Задание 5. Наследование и полиморфизм.

Создайте класс Car, который будет содержать следующие приватные поля: Make, Model, и Year. Реализуйте конструктор для инициализации этих полей.

Добавьте свойства к классу Car, которые будут обеспечивать доступ к полям класса. Свойства должны иметь возможность чтения и записи.

Создайте класс ElectricCar, который наследуется от класса Car. Добавьте новое поле BatteryCapacity и реализуйте соответствующий конструктор.

Реализуйте метод DisplayInfo в обоих классах (Car и ElectricCar), который выводит информацию о машине. Используйте полиморфизм для вызова этого метода.

Отчет должен содержать (см. образец):

- номер и тему лабораторной работы;
- фамилию, номер группы студента и вариант задания;
- скриншоты окна Visual Studio с исходным кодом программ и комментариями;
- скриншоты с результатами выполнения программ;
- пояснения, если необходимо;
- выводы.

Отчеты в формате pdf отправлять на email: **colledge20education23@gmail.com**