

Закрепление материала

- Что такое класс?
- Что такое объект?
- Что такое экземпляр класса?
- Что такое ООП?
- Назовите основные парадигмы ООП.
- Что такое инкапсуляция?
- Что такое свойство?
- Какие виды свойств бывают?
- Что такое модификаторы доступа и где их используют?
- Какие типы конструкторов вы знаете?
- Зачем использовать конструкторы по умолчанию?
- Что такое partial класс?
- Что такое partial метод?
- Когда нужно применять частичные классы и методы?
- Что такое ООП?
- Какие основные парадигмы ООП вы знаете?
- Что такое инкапсуляция?
- Что такое ассоциация?
- Какие связи отношений между классами вы знаете?
- Чем поля, помеченные ключевым словом readonly, отличаются от констант?

Задания

Задание 1

Требуется:

Создать класс с именем `Address`.

В теле класса требуется создать поля: `index`, `country`, `city`, `street`, `house`, `apartment`. Для каждого поля, создать свойство с двумя методами доступа.

Создать экземпляр класса `Address`.

В поля экземпляра записать информацию о почтовом адресе.

Выведите на экран значения полей, описывающих адрес.

Задание 2

Требуется: Создать класс с именем `Rectangle`.

В теле класса создать два поля, описывающие длины сторон `double side1`, `side2`.

Создать пользовательский конструктор `Rectangle(double side1, double side2)`, в теле которого поля `side1` и `side2` инициализируются значениями аргументов.

Создать два метода, вычисляющие площадь прямоугольника - `double AreaCalculator()` и периметр прямоугольника - `double PerimeterCalculator()`.

Создать два свойства `double Area` и `double Perimeter` с одним методом доступа `get`.

Написать программу, которая принимает от пользователя длины двух сторон прямоугольника и выводит на экран периметр и площадь.

Задание 3

Требуется:

Создать класс `Book`. Создать классы `Title`, `Author` и `Content`, каждый из которых должен содержать одно строковое поле и метод `void Show()`.

Реализуйте возможность добавления в книгу названия книги, имени автора и содержания.

Выведите на экран разными цветами при помощи метода `Show()` название книги, имя автора и содержание.

Задание 4

Требуется:

Создать классы `Point` и `Figure`.

Класс `Point` должен содержать два целочисленных поля и одно строковое поле.

Создать три свойства с одним методом доступа `get`.

Создать пользовательский конструктор, в теле которого проинициализируйте поля значениями аргументов.

Класс `Figure` должен содержать конструкторы, которые принимают от 3-х до 5-ти аргументов типа `Point`.

Создать два метода: `double LengthSide(Point A, Point B)`, который рассчитывает длину стороны многоугольника; `void PerimeterCalculator()`, который рассчитывает периметр многоугольника.

Написать программу, которая выводит на экран название и периметр многоугольника.

Задание 5

Требуется:

Создать класс `User`, содержащий информацию о пользователе (логин, имя, фамилия, возраст, дата заполнения анкеты). Поле дата заполнения анкеты должно быть проинициализировано только один раз (при создании экземпляра данного класса) без возможности его дальнейшего изменения.

Реализуйте вывод на экран информации о пользователе.

Задание 6

Требуется:

Создать класс `Converter`.

В теле класса создать пользовательский конструктор, который принимает три вещественных аргумента, и инициализирует поля соответствующие курсу 3-х основных валют, по отношению к гривне - `public Converter(double usd, double eur, double rub)`.

Написать программу, которая будет выполнять конвертацию из гривны в одну из указанных валют, также программа должна производить конвертацию из указанных валют в гривну.

Задание 7

Требуется:

Создать класс `Employee`.

В теле класса создать пользовательский конструктор, который принимает два строковых аргумента, и инициализирует поля, соответствующие фамилии и имени сотрудника.

Создать метод рассчитывающий оклад сотрудника (в зависимости от должности и стажа) и налоговый сбор.

Написать программу, которая выводит на экран информацию о сотруднике (фамилия, имя, должность), оклад и налоговый сбор.

Задание 8

Требуется:

Создать класс `Invoice`.

В теле класса создать три поля `int` account, `string` customer, `string` provider которые должны быть проинициализированы один раз (при создании экземпляра данного класса) без возможности их дальнейшего изменения.

В теле класса создать два закрытых поля `string` article, `int` quantity

Создать метод расчета стоимости заказа с НДС и без НДС.

Написать программу, которая выводит на экран сумму оплаты заказанного товара с НДС или без НДС.