



# Принципы объектно-ориентированного программирования

# Поколения языков программирования

- Развитию программирования поспособствовала идея Джона Фон Ньюмана (англ. John Von Neumann), опубликованная в 1945 году, в которой он описал компьютер, где в памяти вместе с данными хранится и сама программа.

# Поколения языков программирования

- Языком программирования первого поколения считается машинный код. **Машинный код** состоит из инструкций, которые компьютер (процессор) может выполнить (а также данных, которые принадлежат этим инструкциям)

# Поколения языков программирования

- Языками второго поколения считаются языки Ассемблера. При программировании на языке Ассемблера инструкции представлены человеку в понятной форме.

# Поколения языков программирования

- Языки ассемблера и машинный код считаются языками низкого уровня.

# Поколения языков программирования

- Языками программирования третьего поколения называются уже языки высокого уровня. Такие языки программирования не очень связаны с аппаратными средствами.

# Языки третьего поколения

- FORTRAN (The IBM Mathematical FORmula TRANslating System) - язык программирования, разработанный в 1950-тых годах для математических вычислений и в научных целях.
- COBOL (COmmon Business Oriented Language) - Объектно-ориентированный язык программирования, созданный в 1959 году, в основном для написания программ удовлетворяющих бизнес нуждам.
- BASIC (Beginner's All-purpose Symbolic Instruction Code) - Язык, разработанный в 1963 году, который поначалу создавался для того, чтобы инженеры могли производить на компьютерах различные симуляции.

# Языки третьего поколения

- C - имя происходит от того, что этот язык в основном базировался на языке с именем B. Создавался для написания операционных систем (многие UNIX совместимые операционные системы написаны на этом языке), уже долгое время является одним из самых популярных языков программирования.
- C++ - объектно-ориентированный C.
- Java - язык программирования, разработанный на основе C++.
- Visual Basic, Delphi, Python, C# - все являются языками третьего поколения. Многие языки программирования третьего поколения моложе (новее), чем некоторые языки четвёртого и пятого поколений.

# Поколения языков программирования

- Языки программирования четвертого поколения разработаны с целью упростить их изучение и использование. Эти языки, как правило, непроцедурные сосредоточены на единственном применении. Примером языка программирования четвертого поколения может служить SQL (Structured Query Language). Тут больше говорится «что делать» и меньше - «как делать».

# Поколения языков программирования

- Языки пятого поколения созданы для разработки систем искусственного интеллекта и для решения связанных с этой темой проблем.

# Базовые типы языков программирования

- К главным базовым типам относят: **процедурные, функциональные и объектно-ориентированные языки программирования.**

# Базовые типы языков программирования

- В процедурных языках программирования на языке программирования описываются действия и порядок их выполнения, а также эти действия разбиваются на группы (подпрограммы).

# Базовые типы языков программирования

- В функциональных языках программирования всё решение описывается при помощи функций.

# Базовые типы языков программирования

- В объектно-ориентированных языках программирования решение проблемы производится при помощи функций и структур данных, описанных в классах (англ. Class). Из каждого класса можно создать объект, у которого будет набор свойств и/или методов.

# Объектно-ориентированное программирование

- Объектно-ориентированное программирование (ООП) - это результат естественной эволюции более ранних методологий программирования. Потребность в ООП связана со стремительным усложнением приложений и отсюда как следствие недостаточной надежностью программ и выразительными способностями языков программирования.

# Объектно-ориентированное программирование

- ООП - это моделирование объектов посредством иерархически связанных классов.

# Объектно-ориентированное программирование

- Переход от традиционного программирования к ООП на начальном этапе характерен тем, что под объектами в программе подразумеваются конкретные физические объекты. В этом случае легче дается понимание различных действий над ними.

# Объектно-ориентированное программирование

- **Графический примитив**

- Нарисовать
- Стереть
- Получить координаты
- ...

- **Круг**

- Нарисовать через центра и радиус
- Получить координаты центра
- ...

- **Квадрат**

- Нарисовать через диагональ и точку
- Нарисовать через правую верхнюю левую нижнюю точку
- ...

- **Параллелограмм**

- Нарисовать через правую верхнюю, левую нижнюю и величину угла
- ...

# Принципы ООП

- ООП
  - Абстрагирование
  - Наследование
  - Полиморфизм
  - Инкапсуляция

# Принципы ООП

- **Абстрагирование** — это способ выделить набор значимых характеристик объекта, исключая из рассмотрения незначимые. Соответственно, абстракция — это набор всех таких характеристик.

# Принципы ООП

- **Инкапсуляция** — это свойство системы, позволяющее объединить данные и методы, работающие с ними в классе, и скрыть детали реализации от пользователя.

# Принципы ООП

- **Наследование** — это свойство системы, позволяющее описать новый класс на основе уже существующего с частично или полностью заимствуемой функциональностью. Класс, от которого производится наследование, называется базовым, родительским или суперклассом. Новый класс — потомком, наследником или производным классом.

# Принципы ООП

- **Полиморфизм** — это свойство системы использовать объекты с одинаковым интерфейсом без информации о типе и внутренней структуре объекта.

# Базовые понятия ООП

- **Класс** является описываемой на языке терминологии исходного кода моделью ещё не существующей сущности (объекта). Фактически он описывает устройство объекта, являясь своего рода чертежом.

# Базовые понятия ООП

- **Объект** - сущность в адресном пространстве вычислительной системы, появляющаяся при создании экземпляра класса или копирования прототипа.

# Базовые понятия ООП

- **Поля и свойства** - представляют сведения, содержащиеся в объекте. Поля подобны переменным, так как они могут быть прочитаны или изменены напрямую. Обычно они имеют схожие названия. Однако поле пишется с маленькой буквы и доступ к нему закрыт, свойство пишется же, наоборот, с большой буквы и имеет свои методы акселераторы `get`, `set`.

# Базовые понятия ООП

- **Методы - действие, которое выполняет объект, называется методом. Методы всегда называются при помощи глаголов, которые их характеризуют.**

# Базовые понятия ООП

```
class Test
{
    private string message;
    ссылка 0
    public string Message {
        set { message = value; }
        get { return message; }
    }
    ссылка 0
    public void PrintMessage()
    {
        Console.WriteLine(message);
    }
}
```

Пол  
е  
Свойств  
о

Мето  
д

# Базовые понятия ООП

**Клас    Объект**

```
с    т
Test objTest = new Test();
objTest.Message = "Hello word!";
objTest.PrintMessage();
```

**Действия с  
объектом**

# Задания

В тетрадях придумать поля и методы для сущности человек в таких реалиях как:

- 1) Введение учета успеваемости;
- (ФИО, предметы, оценки, получить оценку)
- 2) Амбулаторная карта больного;
- (ФИО, история болезни, жалобы, самочувствие, получить препарат)
- 3) Отдел кадров предприятия;
- (ФИО, образование, должность, оклад, дата приема на работу, принять/уволить с работы)

# Задания

- Придумать цепочку из 3 классов. Которые будут наследовать один от одного.
- Пример:
- Животное ->Млекопитающее->Лев

# Задания

- Развить цепочку дописав методы, которые нужно переопределить в наследуемом классе.