



# Классы в С#

# Класс

- производный (агрегатный) тип данных, введённый программистом на основе уже существующих типов, характеризующийся объединением разнотипных данных и программного кода для их обработки

# Класс - тип данных

int

double

bool

...

класс1

класс2

# Состав класса

**Члены класса** – составляющие класс атомарные единицы

**Данные класса** – переменные внутри класса

**Методы класса** – программный код для обработки данных (не всегда)

# Определение класса

```
class ИмяКласса
```

```
{
```

```
    доступ тип имя;
```

```
    доступ тип имя(параметры)  
    оператор;
```

```
}
```

# Спецификаторы доступа

- **public** – доступен из любой точки программы
- **private** – доступен только внутри этого же класса
- **protected** – доступен внутри этого же класса и его наследников

# Простейший пример

```
class Human
{
    public string FName, LName;
    public int BirthYear;
}
```

# Дубль два: класс – тип данных!

```
int = 5; // Так не бывает!
```

```
int Radius;
```

```
Radius = 5;
```

```
Human Vasya;
```

```
Vasya = ????
```



# Объекты

Переменная, у которой тип данных какой-либо класс – объект (экземпляр класса)

В C# объекты доступны только косвенно – через ссылки

# Создание ссылки и объекта

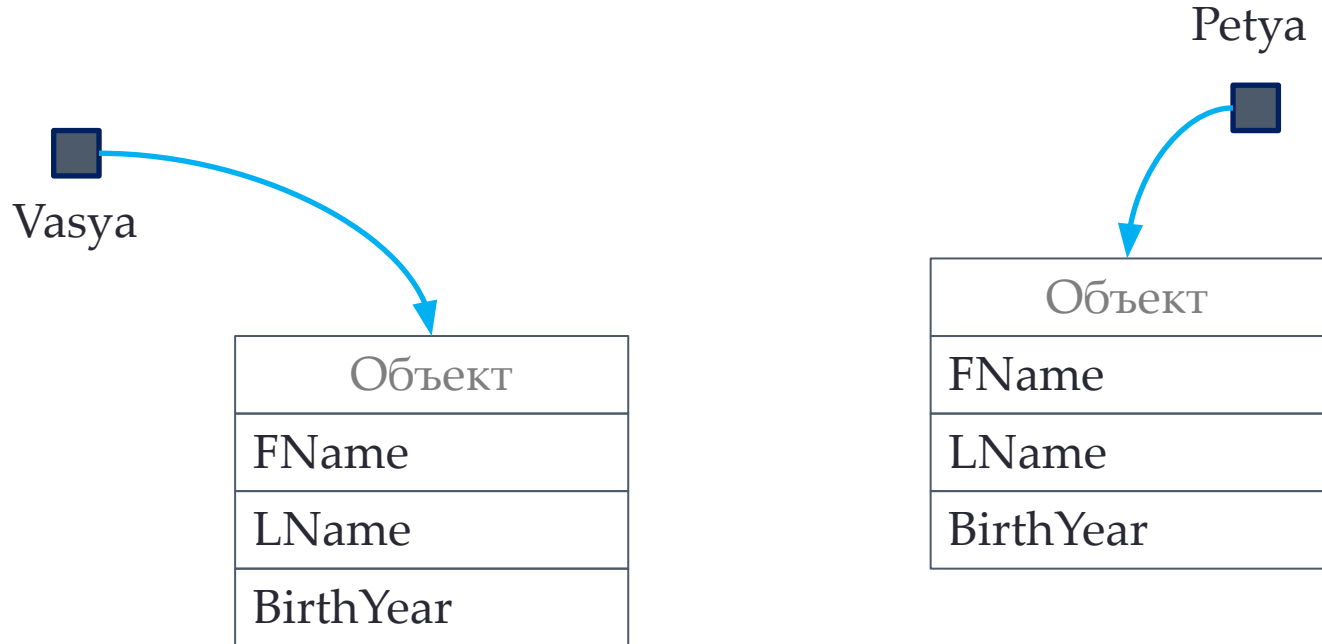
// Всегда в два этапа!

Human Vasya; // ссылка

Vasya = **new** Human(); // **новый объект**  
// ссылке присваивается адрес

Human Vasya = **new** Human();

# Представление в памяти



# Заполнение данных объекта

```
Human Vasya = new Human();  
Vasya.FName = "Василий";  
Vasya.LName = "Иванов";  
Vasya.BirthYear = 2001;
```

# Заполнение данных объекта

```
Human Petya = new Human();
```

```
Petya.FName = "Петров";
```

```
Petya.LName = "Пётр";
```

```
Petya.BirthYear = 2003;
```

# Данные класса == переменные

```
Radius = 5;
```

```
Petya.BirthYear = 2003;
```

```
Console.WriteLine( Radius );
```

```
Console.WriteLine( Petya.BirthYear );
```



Продолжение следует...