

ОБЪЯВЛЕНИЕ ПЕРЕМЕННЫХ

Создание и инициализация переменных

БЛОК: ОСНОВЫ
JAVA



План урока



Типы объявлений
переменных



Понятие
«инициализация»



Где может быть объявлена
переменная



Синтаксис
объявления



Переменные

- ✓ От самого названия «переменная» - изменяемое по ходу работы программы значение
 - ✓ Без переменных не было бы никакой программы на Java (или другом языке)
 - ✓ Понятие используется почти во всех языках программирования
 - ✓ В зависимости от места объявления – хранится в разных «местах» памяти
 - ✓ **Переменная в Java обязательно должна иметь «тип» и «имя»**
-

....

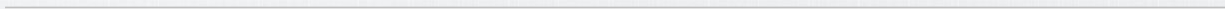
Объявление переменной без явной инициализации

Модификатор
доступа

Имя
переменной

```
private String name;
```

Тип
переменной



...

Объявление переменной с явной инициализацией

```
Camry camry = new Camry();
```

```
private int i = 1;
```



Где может быть объявлена переменная



В классе (вне метода) – может называться: «instance переменная», «свойство объекта», «поле»

- Описывает характеристику объекта или просто является внутренней переменной
- Уничтожается вместе с объектом
- Область видимости - видна во всех методах класса
- Чаще всего – имеет доступ через гетер/сетер (но может и не иметь их, если не нужен доступ извне)
- Используется модификатор доступа (даже если не указан явно)
- Инициализируется по-умолчанию (если значение не указано)



Внутри метода – локальная переменная

- Уничтожается сразу после выполнения метода
 - Область видимости – только внутри метода
 - Нельзя использовать модификатор доступа
 - Обязательно нужно инициализировать
-



Значения по-умолчанию

При объявлении переменной компилятор может присвоить ей значение по-умолчанию, если явно ничего не указано



Instance переменная: в зависимости от типа переменной - примитивный или объектный (ссылочный):

- `boolean = false;`
- `int = 0;`
- `String = null;`
- ...



Локальная переменная не инициализируется по-умолчанию!
Нужно явно указывать значение!

- Часто из-за этого возникает ошибка компиляции
-



Синтаксис объявления

- ✓ Каждая переменная в отдельной строке (частый вариант)

```
private int x;  
private int y;  
private int z;
```

- ✓ Если переменные одного типа – можно указывать их в одной строке через запятую

```
private int x,y,z;
```

В обоих случаях переменные можно инициализировать



Имена переменных и области видимости

- ✓ В одной области видимости нельзя создавать две переменные с одним именем
 - ✓ Локальная переменная может «замещать» переменную класса
 - ✓ Часто граница области видимости – скобки {
... }
-