



ПЕРЕМЕННЫЕ

ЗНАЧЕНИЕ

```
val x = "value"
```

- Именованное значение, вычисляется сразу

ФУНКЦИЯ БЕЗ ПАРАМЕТРОВ

```
def x = "value"
```

- Метод без параметров - вычисляется каждый раз

ЛЕНИВЫЕ ЗНАЧЕНИЯ

```
lazy val x = "value"
```

- Ленивое значение - вычисляется при первом требовании

ПЕРЕМЕННЫЕ

```
var x = "value"
```

```
x = "new value"
```

- Настоящая переменная - может менять значение

ИСПОЛЬЗОВАНИЕ

```
val x = 1
```

```
val y = 2
```

```
val z = x + y
```

ПЕРЕМЕННЫЕ

```
val z = x + y
```

```
def x = 1
```

```
lazy val y = 2
```

УКАЗАНИЕ ТИПА

```
val x: Int = 1
```

```
val y: String = "value"
```

```
val z: Double = x
```

- Можно и часто нужно указывать тип

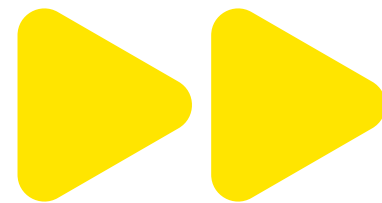
ОБЛАСТЬ ВИДИМОСТИ

```
val x = "Outer"  
{  
    val x = "Inner"  
  
    println(x)  
}
```

ОБЛАСТЬ ВИДИМОСТИ

```
{  
    val x = "Inner"  
  
    println(x)  
}  
println(x) //Ошибка!!!
```

Мы изучили области видимости



**В следующем разделе
изучим пространства имён**