ES6:
asp.net.by/es6/1.htm, asp.net.by/es6/02.htm
```
{
 let
 const
}
```

asp.net.by/es6/03.htm **Arrow functions:**
x = (x, y) => y + ":" + x;
                              x("sbmt.by", "SBMT");

asp.net.by/es6/04.htm function myExam(y = 4) **–
параметр по умолчанию**

asp.net.by/es6/05.htm Array.**find**(myFunction)
asp.net.by/es6/06.htm Array.**findIndex**()
asp.net.by/es6/07.htm Классы
```
class ACat {

 constructor(n) {

   this.name = n;     //property

 }

}
```
mycat = new ACat("Барсик");

```
class ACat{
...................................
 Name()
   return 'Меня зовут ' + this.name;
 }
}
mycat = new ACat("Барсик");
document.write(mycat.Name());
```

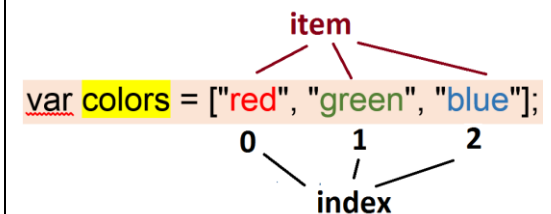| asp.net.by/es6/08.htm	Меня зовут Барсик
asp.net.by/es6/09.htm Array.**forEach**() var colors = ["red", "green", "blue"];	item var colors = ["red", "green", "blue"]; 0 1 2 index

```
colors.forEach(myFunction);
function myFunction(item, index) {
   item … index
}
```

Function One (){
    // Do something
}
Function Two (call_One){
    // Do something else
    call_One()
}
Two(One);   code is being executed

Callback illustrated

Promise
new Promise(executor)
state: "pending"
result: undefined
resolve(value) → state: "fulfilled" result: value
reject(error) → state: "rejected" result: error