

```

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta name="robots" content="noindex">

```



```

<style>
  #container {
    display: flex;
    flex-wrap: wrap;
    align: center;
  }
  #container > div {
    background-color: gray;
    font-size: 20px;
    margin: 20px;
    padding: 20px;
    width: 200px;
    text-align: center;
  }
</style>

```

```

<body>
  .....
  <div id="container">
    <div>Project 0</a></div>
    <div>Project 1</div>
    <div>Project 2</div>
    <div>Project 3</div>
  </div>
</body>
<html>

```



<https://dot.net.by/esperanto/>

```

<!-- Google Font -->
  <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@300;500;700&display=swap"
  rel="stylesheet">

  <style>
    body {
      font-family: 'Poppins', sans-serif;

      <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.min.css"
      rel="stylesheet">

      <div class="buttons">
        <a href="project1.html" class="btn btn-primary btn-project">Project 1</a>

        @media (min-width: 768px) {
          .profile-image {
            width: 200px;
            height: 200px;
          }
          .name {
            font-size: 2.5rem;
          }
        }
      }

```

to enlarge photos on large screens from 150 to 200px

The same goes for fonts

get good stuff / library
Mark Otto

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) // - default option

PO.1 25.2.25

```
asp.net.by/es6/05.htm Array.find(myFunction)
fibonacci = [1, 1, 2, 3, 5, 8]
index 0 1 2 3 4
var first = fibonacci.find(myFunction);
function myFunction(value, index, array) {
  return value > 4;
}
```

asp.net.by/es6/05.htm Array.findIndex()

var fibonacci = [1, 1, 2, 3, 5, 8];

fibonacci[4] // The first digit > 4 - has index 4

var first = fibonacci.findIndex(myFunction);

function myFunction(value, index, array) {

return value > 3;



asp.net.by/es6/07.htm Классы

class ACat { constructor(n) {

this.name = n; //property

}

}

mycat = new ACat("Барсик");



```
class ACat {
  constructor(n) {
    this.name = n;
  }
}
```

return 'Меня зовут ' + this.name;

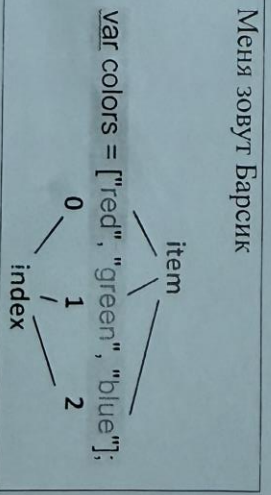
}

mycat = new ACat("Барсик");

document.write(mycat.Name());

40.1 3X 2025
+0.1 14X 2025

```
asp.net.by/es6/08.htm
asp.net.by/es6/09.htm
Array.forEach()
var colors = ["red", "green", "blue"];
```



colors.forEach(myFunction);

function myFunction(item, index) {

item ... index

}

Function One () {
// Do something

Function Two (call_One) {
// Do something else

call_One()

}

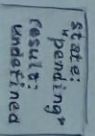
Two(One); // code is being executed

https://asp.net.by/callback

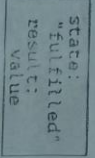
Callback illustrated

Promise

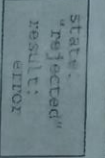
new Promise (executor)



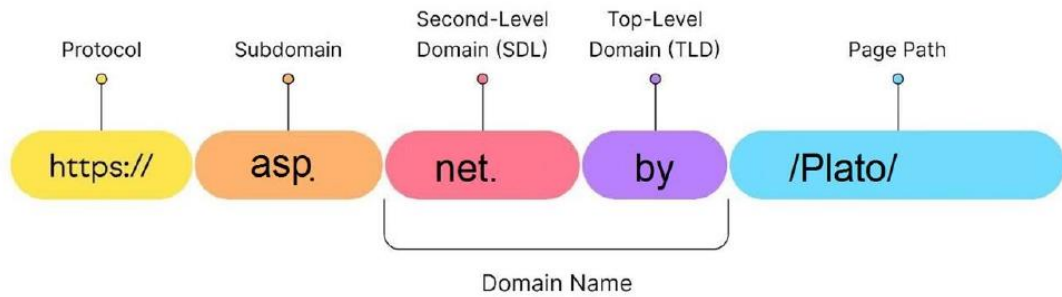
resolve(value)



reject(error)



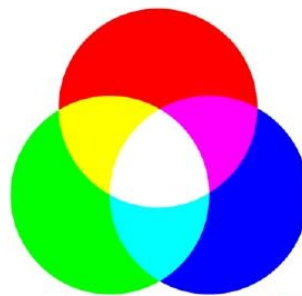
https://asp.net.by/es6/13.promise.htm



R- Red


G- Green

B – Blue

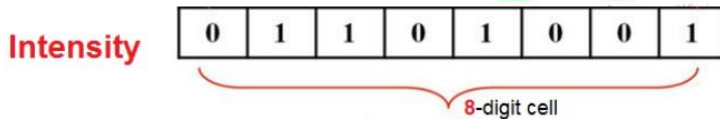


#ff0000 → 

R G B

#00A2E8 → 

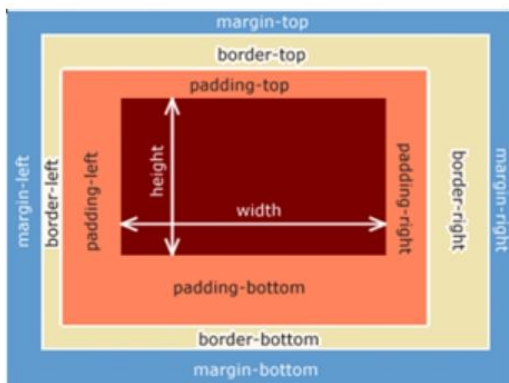
R G B



rgb(255,0,0), rgb(0,0,255), rgb(0,162,232), rgb(31,78,121)

rgb(100%,0%,0%), rgb(0%,0%,100%)

Box model



```
<style>
div
{
  color:white;
  background-color:brown;
  height: 250px; width: 100px;
  padding: 20px 11px 10px 15px;
  border: 15px solid yellow;
  margin: 120px 10px 15px 20px;
}
</style>
```

```
<div><b>Box Model</b><br>
```

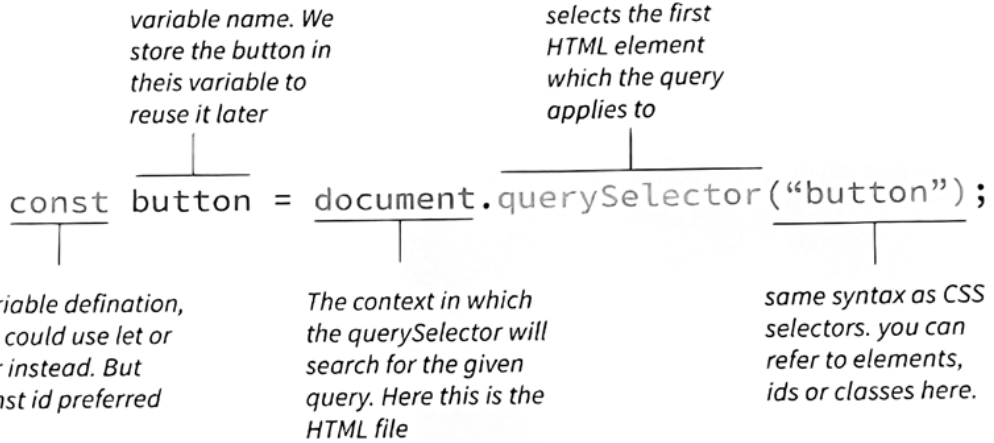
```
<style>
img {
  position: absolute;
  left: 2px;
  top: 10px;
  z-index: -1; // behind the text }
</style>
```



ES6: event-driven programming

querySelector

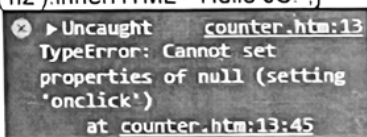
First we need to access the button element in our JavaScript code.



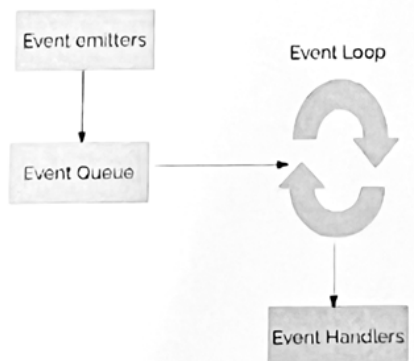
```
<html>
<head>
  <title>Hello</title>
  <script>
    function Hello(){
      document.querySelector("h2").innerHTML=
        "Hello JavaScript!";
    }
  </script>
</head>
<body>
<h2></h2>
<button onclick="Hello()">Click Me!</button>
</body>
</html>
```

```
document.addEventListener
('DOMContentLoaded',function(){
  document.querySelector('button').onclick
    = Hello; }
);
```

```
<html>
<head>
  <title>Hello</title>
</head>
<script>
  function Hello(){
    document.querySelector("h2").innerHTML="Hello JS!";
  }
</script>
<body>
<h2></h2>
<script>
  document.querySelector('button').onclick =
    Hello;
</script>
<button onclick="Hello()">Click Me!</button>
<script>
  document.querySelector('button').onclick = Hello;
</script>
</body>
</html>
```



```
document.querySelector('button').addEventListener('click',Hello);
```



```
<button onclick="Hello()">Click Here</button>
<script>
  let counter = 0;
  function Hello(){
    counter++;
    const heading=
      document.querySelector('h1');
    heading.innerHTML = counter;
  }
</script>
```

HTML forms

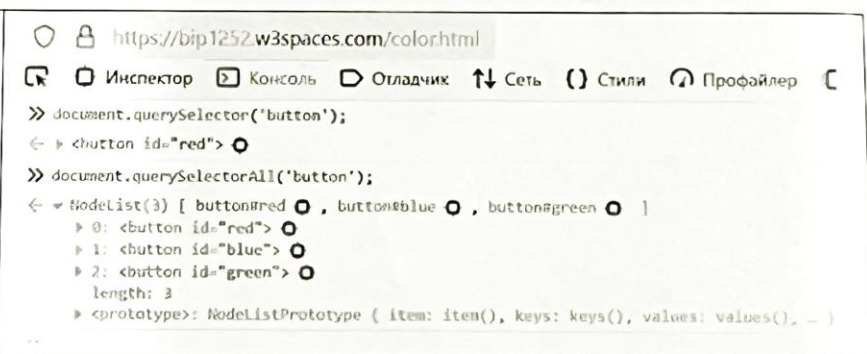
The placeholder - the thing the user sees filled into that input field originally

```
<form>
  automatically focus this input field
  <input autofocus id="name" placeholder="Name" type="text">
  <input type="submit">
</form>
```

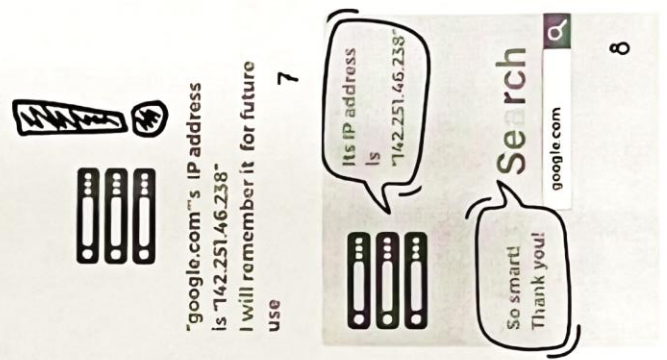
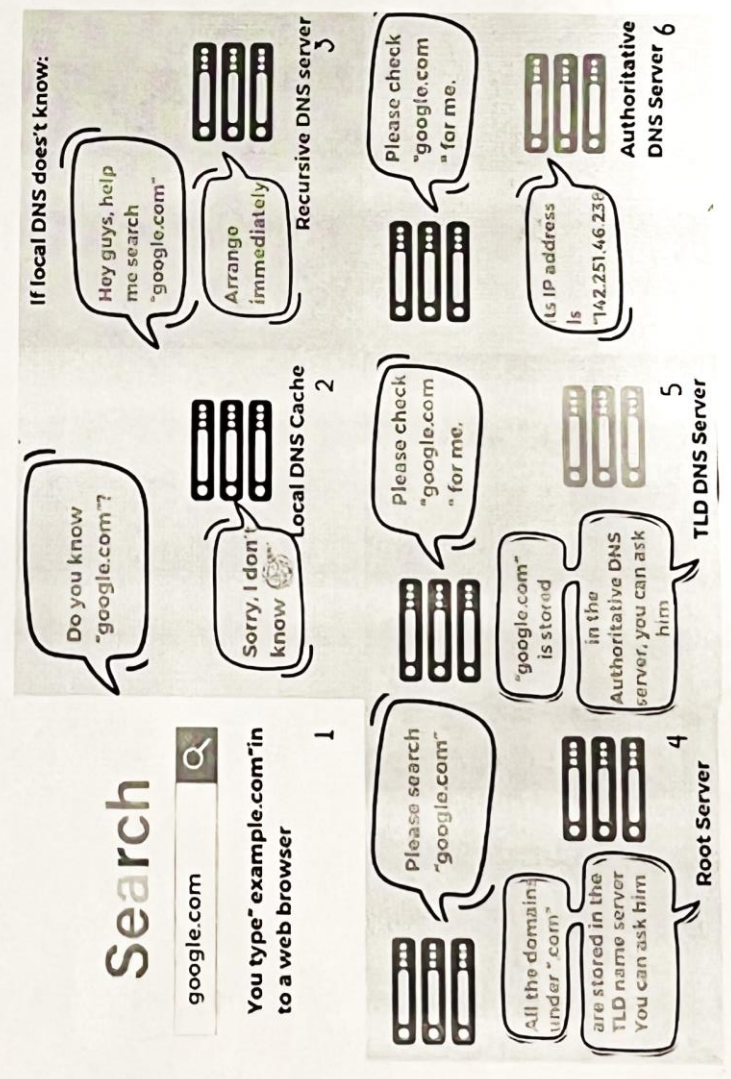
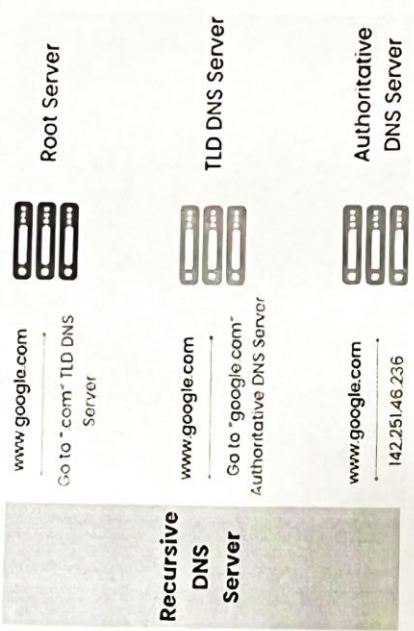
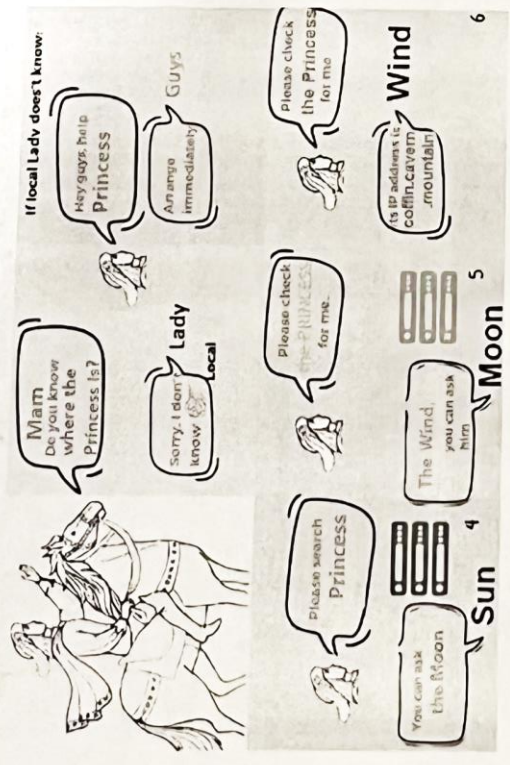


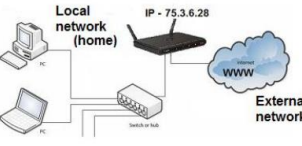
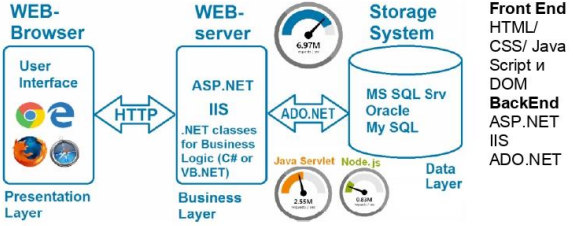
input field where the user can type in some text

```
<script>
  document.addEventListener('DOMContentLoaded', function()
{document.querySelector('form').onsubmit = function() {
  let name = document.querySelector('#name').value;
    alert(`Hello, ${name}!`);
  });
});
</script>
```

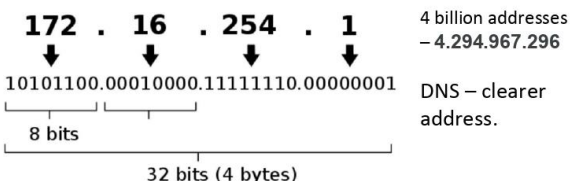
<pre><button id="red">Red </button> <button id="blue">Blue </button> <button id="green">Green </button></pre>	
---	---

```
document.addEventListener('DOMContentLoaded',function(){
document.querySelector('button').onclick = Hello; }
);
document.querySelector('#red').onclick = function() {
  document.querySelector('#hello').style.color = 'red'; };
```

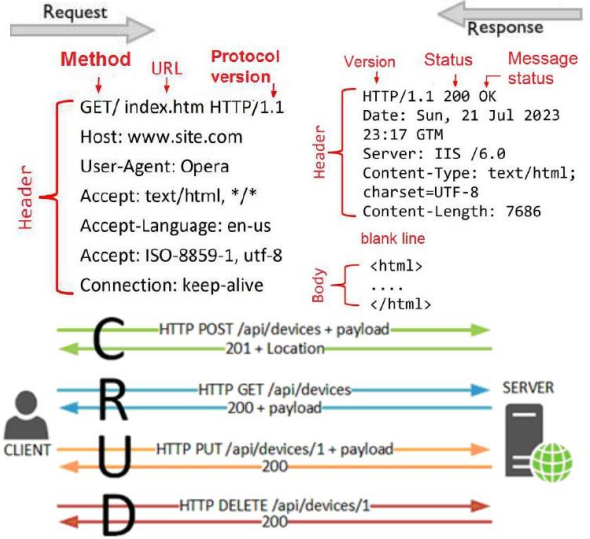




tracert – tracerout (in Apple) www.geotraceroute.com

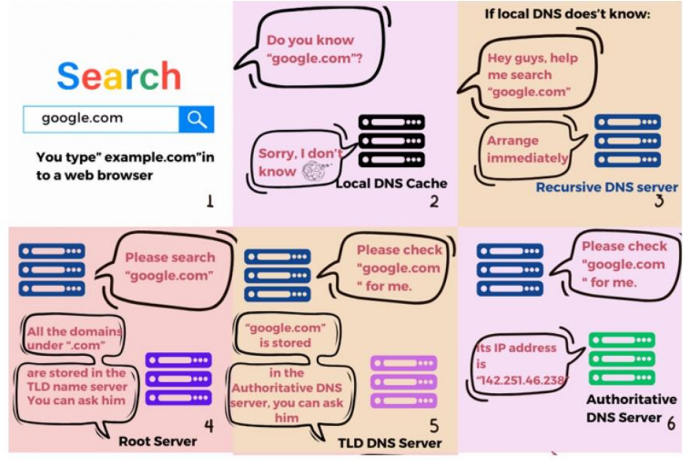
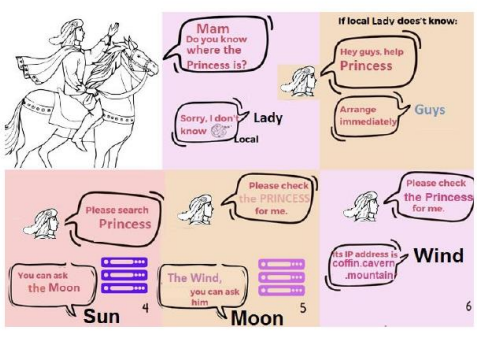






217.21.43.40 – sbmt.by, sbmt.bsu.by, sb.bsu.by
 ping sb.bsu.by, tracert sb.bsu.by, <https://geotraceroute.com/>
 +375 (29) 254 07 92 - ANDREY O. YAROSHEVICH

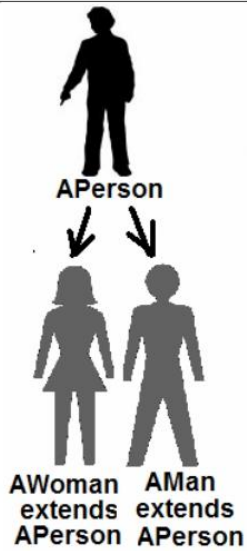


- GET — getting a resource
- POST — resource creation
- PUT — resource update
- DELETE — deleting a resource

Methods	Request		Response	
	URL	Request body	Status	Response body
1 GET	Yes	No	Yes	Yes
2 PUT	Yes	Yes	Yes	No
3 POST	Yes	Yes	Yes	Yes
4 DELETE	Yes	No	Yes	No



<pre>class ACat { string name; public ACat(n){ this.name=n; } } ACat mycat= new ACat("Barsik");</pre>	<pre>class ACat { constructor(n) { this.name = n; //property } } mycat = new ACat("Barsik");</pre>	<pre>Say(){ return "meou"; } s = "My cat says " + myCat.Say()+ "!"</pre>	<pre><script> class APet { Say() { alert("No"); } } class ACat extends APet { Say() { return "Miou"; } } var myPet = new ACat("Barsik"); S = "My pet says " + myPet.Say()+ " !" </script></pre>
<p>class To describe a Woman: name, age, job Women can do: eat, drink, sleep, walk, ...</p>			
<p>Real world objects</p>  <p>object</p> <p>June 19 Student</p>	<p>object</p>  <p>Emma 45 Doctor</p>	<p>object</p>  <p>Ann 30 Engineer</p>	<p>Polymorphism in Biology</p>  <p>Queen Drone Worker</p>




function WashDishes()

Men's version
wipe dry



Female version
wet with water



 <p>ira= new AWoman()</p>	 <p>ivan= new AMan()</p>
--	---

```
class AMan {
  WashDishes() {
    return 'wipe dry';
  }
}
```

```
class AWomen {
  WashDishes() {
    return 'wet';
  }
}
```

```
var family = [new AWomen(), new AMan()];
for(i=0;i<2;i++){ alert(" "+ family[i].WashDishes());} //dry wet
```

```
women = new AWomen(); man = new AMan(); var family = [women,man];
```