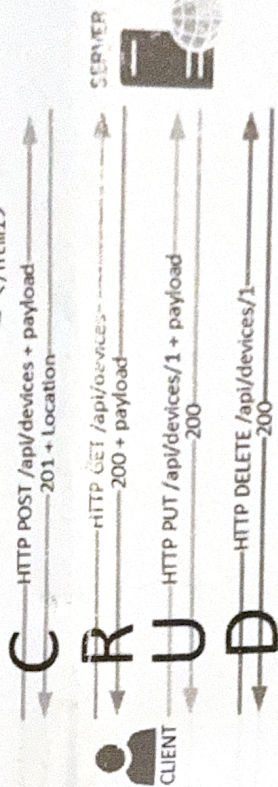
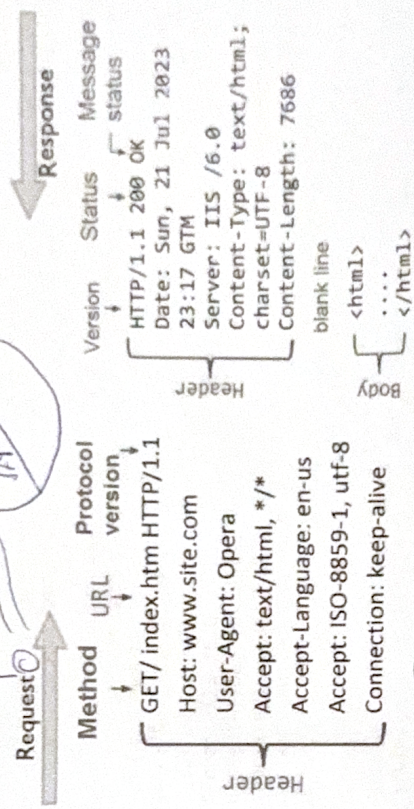
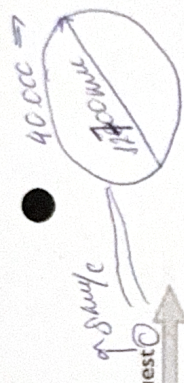


2 1024

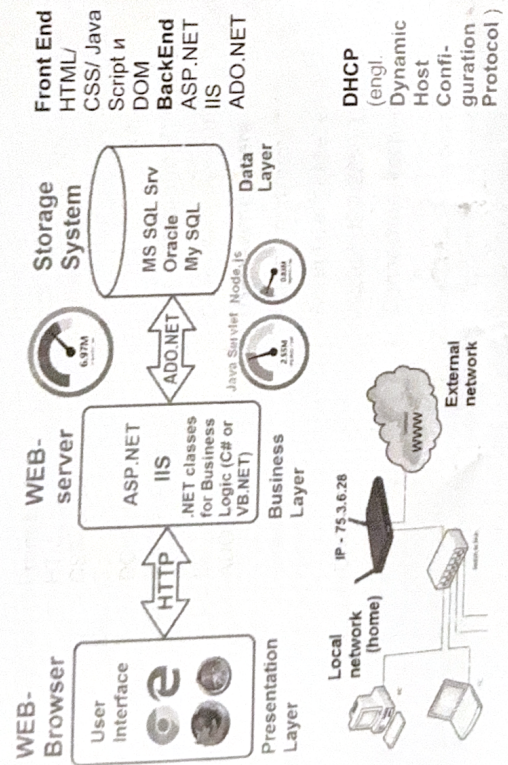
+ 0.2 to Fin Grade

601 25.9.24



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

	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



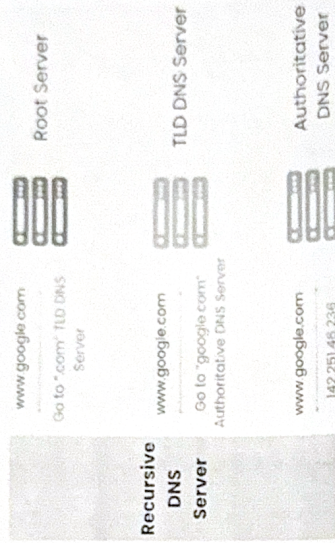
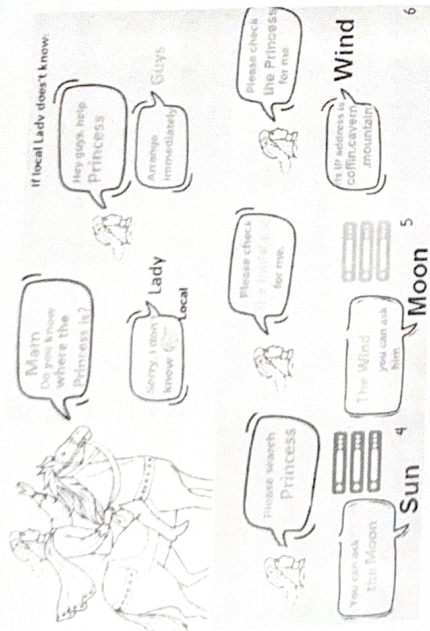
tracert - tracerout (in Apple) www.geotraceroute.com

172 . 16 . 254 . 1
 10101100.00010000.11111111.0.00000001
 8 bits

32 bits (4 bytes)

217.21.43.40 - sbmt.by, sbmt.bsu.by sb.bsu.by
 ping sb.bsu.by tracert sb.bsu.by https://geotraceroute.com/

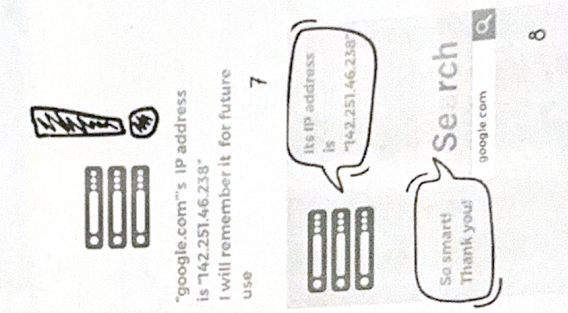
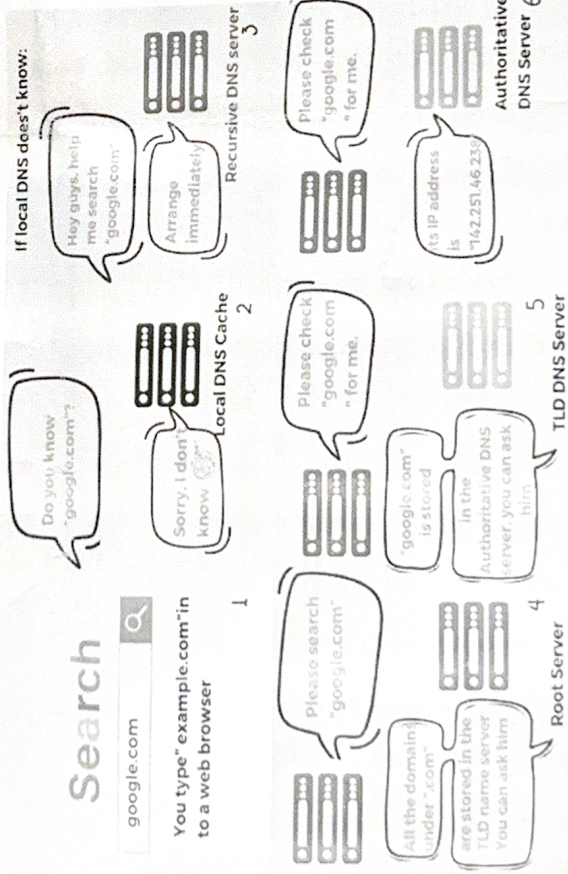
+375 (29) 254 07 92 - ANDREY O. YAROSHEVICH
 D: Vest my site
 krist.mibsk.by
 Libux Apache Tomcat



www.google.com

142.251.46.236

http://

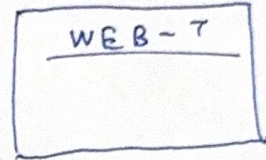


Папки - lower

yarik.runasp.net/WEB/



index.html



→ ao@dot.net.by

+0.1 2025.09.18

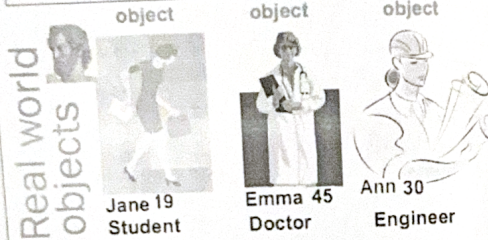
```
class ACat {
  string name;
  public ACat(n){
    this.name=n;
  }
}
ACat mycat= new ACat("Barsik");
```

```
class ACat {
  constructor(n) {
    this.name = n; //property
  }
}
mycat = new ACat("Barsik");
```

```
Say() {
  return "meow";
}
s = "My cat says <b>"
+ myCat.Say()+
"</b> !"
```

```
<script>
class APet {
  Say() {
    alert("No");
  }
}
class ACat extends APet {
  Say() {
    return "Miou";
  }
}
var myPet =
new ACat("Barsik");
S = "My pet says <b>"
+ myPet.Say()+
"</b> !"
</script>
```

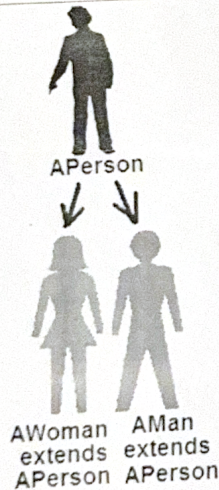
class To describe a Woman: name, age, job
Women can do: eat, drink, sleep, walk, ...



function WashDishes()

Men's version
wipe dry

Female version
wet with water



ira= new
AWoman()



ivan= new
AMan()

```
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];
```



```

<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>

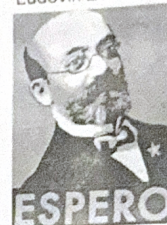
```

```

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

```

Ludovik Zamenhof



Project 0

Project 1

Project 2

Project 3

<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">

Project 1

@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

ES6:

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

Arrow functions:

```
asp.net.by/es6/03.htm
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
```

```
  }
```

```
}
```

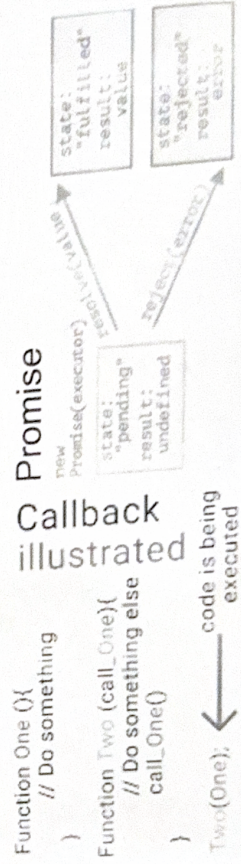
```
mycat = new ACat("Барсик");
```

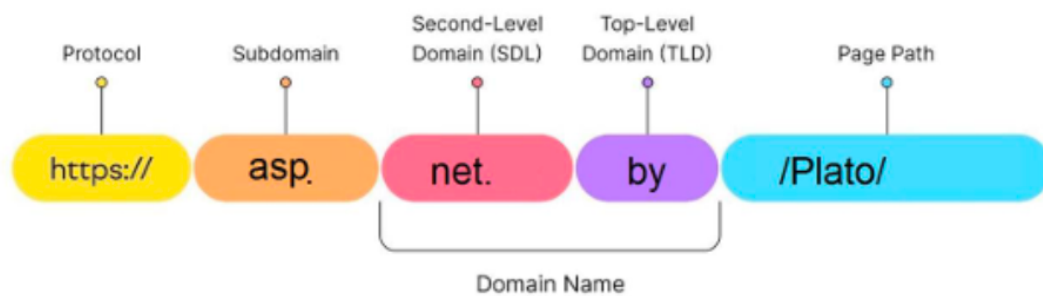
```
class ACat{
  .....
  Name()
  return 'Меня зовут' + this.name;
}

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() | item |
| var colors = ["red", "green", "blue"], | 0 1 2 |
| "green", "blue"; | index |

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





R- Red

G- Green

B – Blue



#ff0000

R G B

#00A2E8

R G B

Intensity

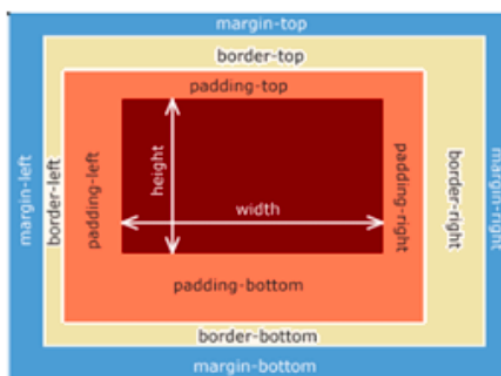
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

8-digit cell

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>Box Model

<style>

img {

position: absolute;

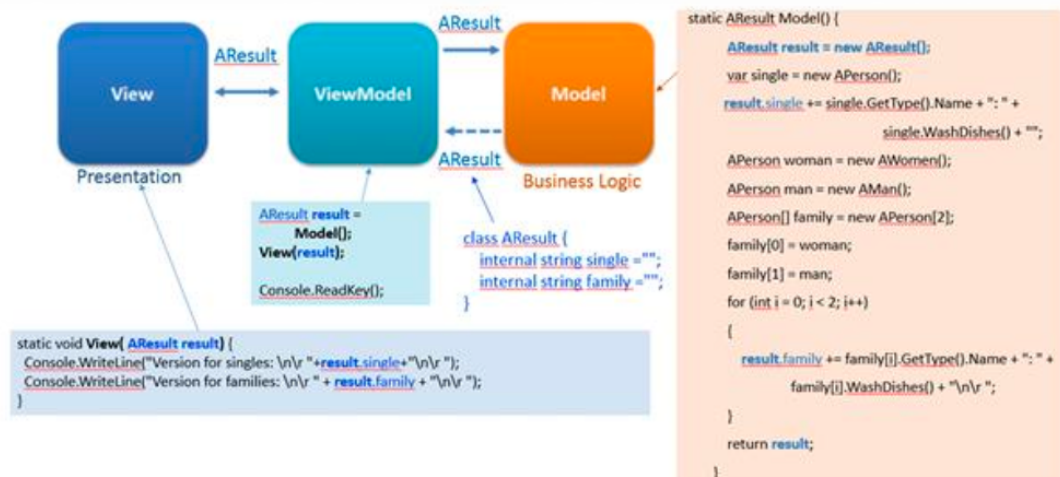
left: 2px;

top: 10px;

z-index: -1; // behind the text }

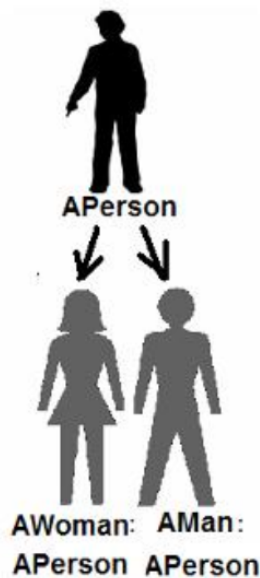
</style>





Men's version
wipe dry

Female version
wet with water



```

class APerson {
    virtual public string WashDishes() {
        return "dry and wet";
    }
}

```



```

class AMan: APerson {
    override public
    string WashDishes()
    {
        return "dry";
    }
}

```

```

class AWoman: APerson {
    override public
    string WashDishes()
    {
        return "wet";
    }
}

```




Сорока-Белобока кашу варила,
Деток кормила:
- Этому дала, - Этому дала,



querySelectorAll() → [, ]
NodeList

[, ].forEach();

```
>> document.querySelectorAll('button');
← ▼ NodeList(3) [ button#red, button#blue, button#green ]
  ▶ 0: <button id="red">
  ▶ 1: <button id="blue">
  ▶ 2: <button id="green">
    length: 3
  ▶ <prototype>: NodeListPrototype { item: item(), keys: keys(), values: values(), ... }
```

```
document.querySelectorAll('button').forEach(function(button) {
  button.onclick = function() {
    document.querySelector('#hello').style.color =
      button.dataset.color;
  };
});
```

Red Blue Green

When the page is done loading

```
<script>
  document.addEventListener('DOMContentLoaded', function() {
    document.querySelectorAll('.color-change').forEach(
      function(button) {
        button.onclick = function() {
          document.querySelector('#hello').style.color = button.dataset.color;
        };
      }
    );
  });
</script>
```

I'm going to document.querySelectorAll, looking for all of the buttons

looked for things of the particular class

HTML

```
<button class="color-change" data-color="red">Red</button>
<button class="color-change" data-color="blue">Blue</button>
<button class="color-change" data-color="green">Green</button>
```

<button class="color-change" data-color="red">Red</button>

function that takes as input, the button

It's first going to pass in as input, the first button

then the second button

then the third button

```
function(button) {
  button.onclick = function() {
```

```
document.querySelector('#hello').
  style.color =
    button.dataset.color;
};
```

```
<head>
  <script>
    document.addEventListener('DOMContentLoaded', () => {
      document.querySelector('#color-change').onchange = function() {
        document.querySelector('#hello').style.color = this.value;
      };
    });
  </script>
</head>
<body>
  <h1 id="hello">Hello!</h1>
  <select id="color-change">
    <option value="red">Red</option>
    <option value="blue">Blue</option>
    <option value="green">Green</option>
  </select>
</body>
```



```
function() {...}
```

```
function(but) {
  but.onclick =
    function() {...}
}
```

```
() => {...}
```

```
but => {
  but.onclick =
    () => {...}
}
```