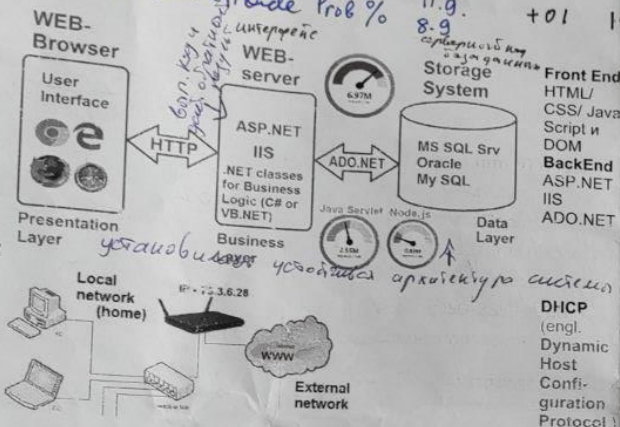


+0.3 to Fin Exam Grade
+0.2 to Fin Grade Prob %

+0.1 15.9.25
+01 19.9.25

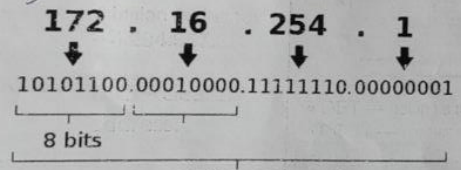
+0.1 2025.09.24

2nd version
1st programming
3rd presentation
4th conclusion
5th conclusion and chapter cycle



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

адрес сайта и 8 байтов



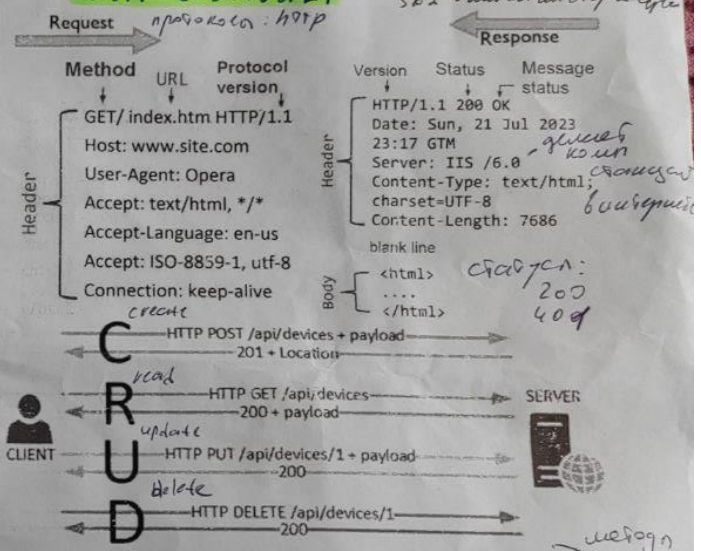
4 billion addresses
- 4.294.967.296
DNS - clearer address.

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
Встретимся впереди для разговора

CRUD-операции
Таблица

Блок - набор из куска
Мегалекс - набор из куска



- 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

ITS Interns
Inf Server
Windows
Linux Apache Tomcat

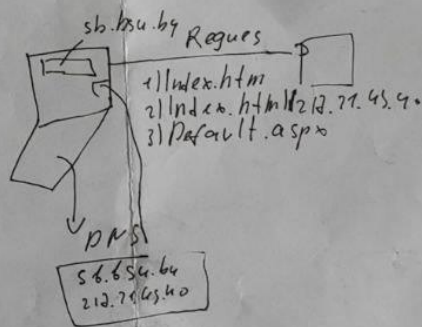
2.חסותה נא 20% סופר

$$100 - 70 = 30 \approx 0,3\%$$

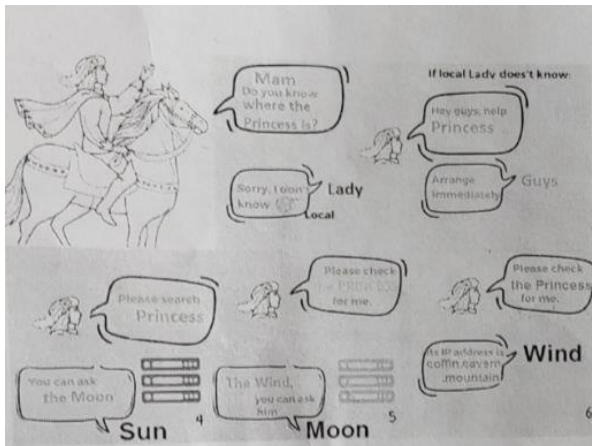
$$0,1 \cdot 0,3 = 0,03\%$$

$$1 - 0,03 = 90\% \text{ סופר}$$

ICANN - יתבטת גולות
ו IP-גולות

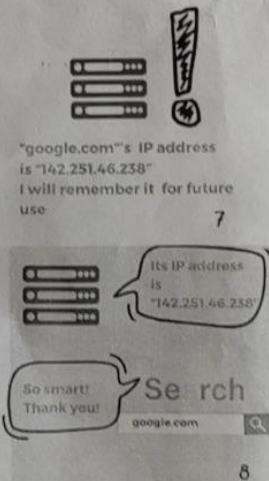
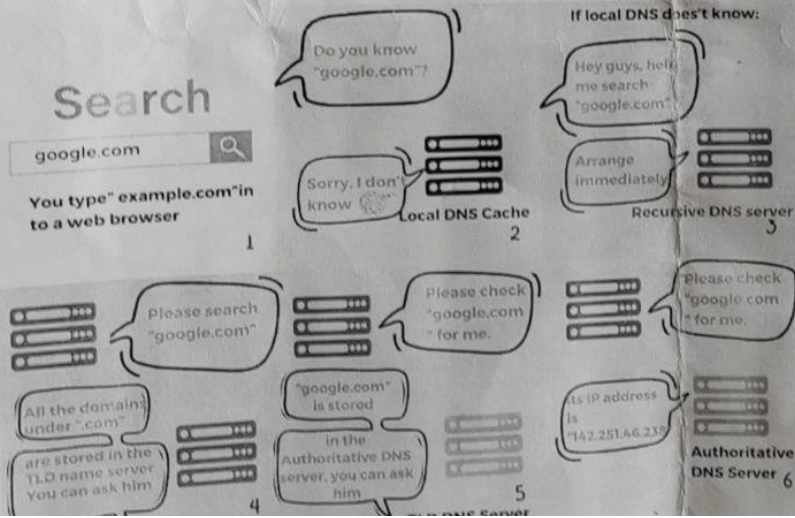
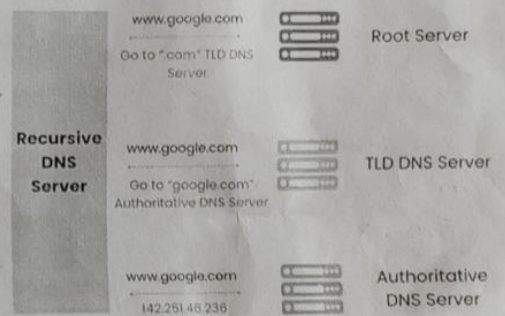


```
class ACat extends Hra
{
    String name;
    public ACat (name)
    {
        this.name = name;
    }
}
```



www.google.com

142.251.46.236



```

<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>

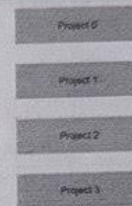
```

```

<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



<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;
    }

```

to enlarge photos on large screens from 150 to 200px

```

    .name {
      font-size: 2.5rem;
    }
  }

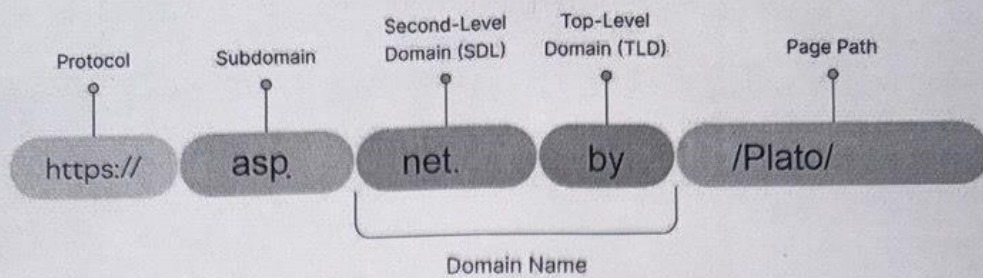
```

The same goes for fonts

```

}

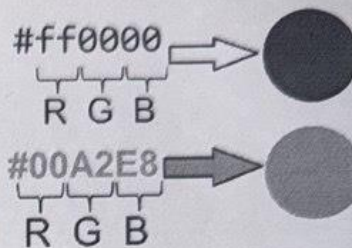
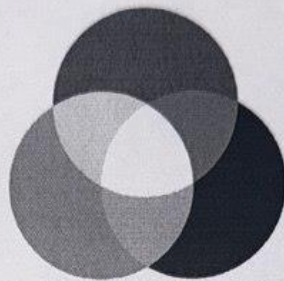
```



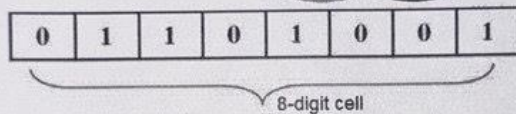
R- Red

G- Green

B – Blue



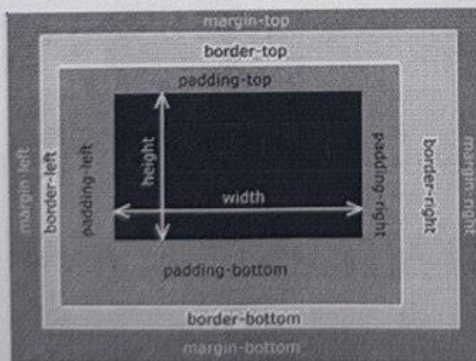
Intensity



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

```
<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>
```



ES6:

asp.net.by/es6/1.htm, asp.net.by/es6/02.htm

```
{
  let
  const
}
```

asp.net.by/es6/03.htm Arrow functions:

$x = (x, y) \Rightarrow y + ":" + x;$

$x("sbmt.by", "SBMT");$

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 {
```

map

```
  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());
```

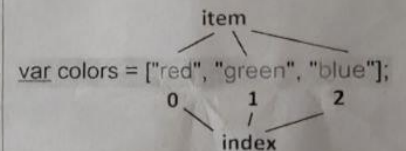
asp.net.by/es6/08.htm

Меня зовут Барсик

asp.net.by/es6/09.htm

Array.**forEach**()

```
var colors = ["red",
  "green", "blue"];
```



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

illustrated

Promise

new Promise(executor)

```
state:
  "pending"
result:
  undefined
```

resolve/value

reject/error

```
state:
  "fulfilled"
result:
  value
```

```
state:
  "rejected"
result:
  error
```

Теория относительности

300.000 км/с



сигнал отходит наблюд. - 300 км/с

относит. наблюдателя - в скорости звука

3 км/с

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

3 км

Error

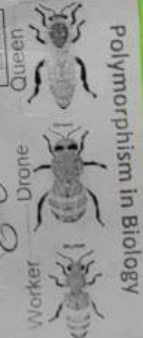
```
class ACat {
  string name;
  public ACat(n){
    this.name=n;
  }
}
ACat mycat= new ACat("Barsik");
```

```
class ACat {
  constructor(n) {
    this.name = n; //property
  }
  Say() {
    return "meou";
  }
  s = "My cat says <b>"
  + myCat.Say()+
  "</b> !"
```

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

Real world objects

object	object	object
 Jane 19 Student	 Emma 45 Doctor	 Ann 30 Engineer



Error

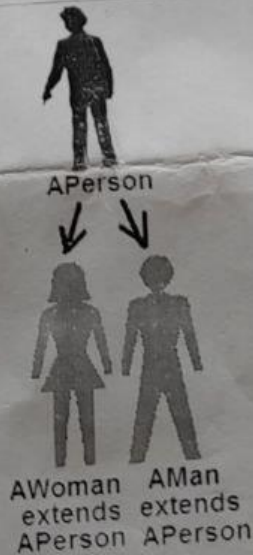
```
<script>
class APet {
  Say() {
    alert("No");
  }
}
class ACat extends APet {
  Say() {
    return "Miou";
  }
}
var myPet =
  new ACat("Barsik");
S = "My pet says <b>"
  + myPet.Say()+
  "</b> !"
</script>
```

+0.1 2025.09.24
+0.1 2025.09.24
Branapubclio

function WashDishes()

Men's version
wipe dry

Female version
wet with water



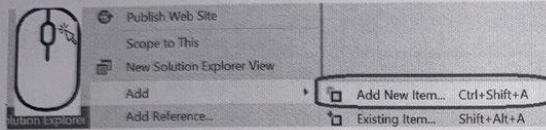
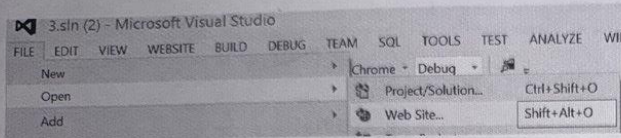
```
class AMan {
  WashDishes() {
    return 'wipe dry';
  }
}
```

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

 ira= new AWoman()	 ivan= new AMan()
---	--

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

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



Default.aspx

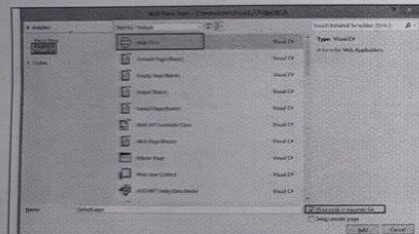
```
asp:Button#cmbStart
Start
<asp:Button ID="cmbStart"
runat="server"
OnClick="cmbStart_Click"
Text="Start" /><br />
```

Run

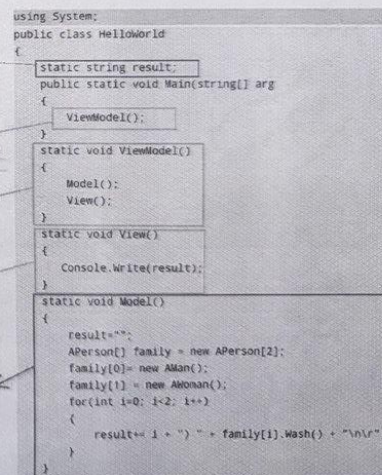
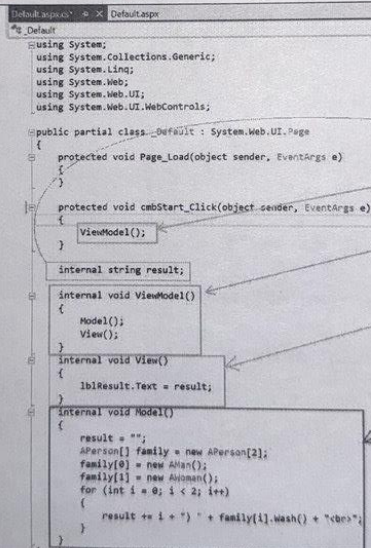
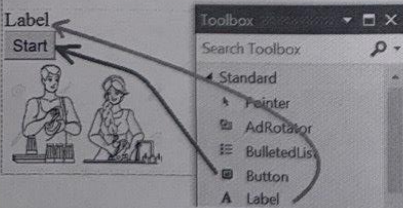
Google Chrome

Default.aspx.cs

```
protected void
cmbStart_Click(
object sender,
EventArgs e)
{
}
```



Projects: 0 1 2 3 4 5 6 7 8 9 A



```
class APerson
{
    virtual internal string Wash()
    {
        return "Wet and Dry";
    }
}
class AMan:APerson
{
    override internal string Wash()
    {
        return "Wet";
    }
}
class AWoman:APerson
{
    override internal string Wash()
    {
        return "Dry";
    }
}
```

```
class APerson
{
    virtual internal string Wash()
    {
        return "Wet and Dry";
    }
}
class AMan:APerson
{
    override internal string Wash()
    {
        return "Wet";
    }
}
class AWoman:APerson
{
    override internal string Wash()
    {
        return "Dry";
    }
}
```



ES6: event-driven programming

анонимная функция -
функция без названия

querySelector

Ways of using
JavaScript Events:

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

- 1) Inline HTML Event handlers
- 2) Add EventListener()
- 3) Traditional DOM Event handler

variable name. We
store the button in
this variable to
reuse it later

selects the first
HTML element
which the query
applies to

```
const button = document.querySelector("button");
```

variable definition,
we could use let or
var instead. But
const is preferred

The context in which
the querySelector will
search for the given
query. Here this is the
HTML file

same syntax as CSS
selectors. you can
refer to elements,
ids or classes here.

```
<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; }
);
```

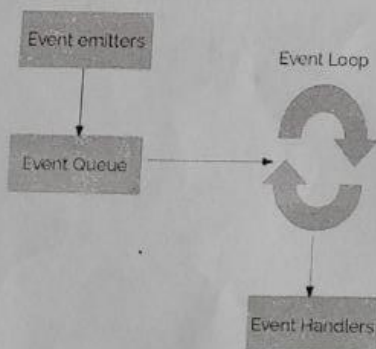
```
<html>
<head>
  <title>Hello</title>
</head>
<script>
  function Hello(){
    document.querySelector("h2").innerHTML = "Hello JS!";
  }
</script>
<body>
  <h2></h2>
```

```
<script>
  document.querySelector('button').onclick =
    Hello;
</script>
```

```
<button onclick="Hello()">Click Me!</button>
<script>
  document.querySelector('button').onclick = Hello;
</script>
</body>
</html>
```

Uncaught **counter.htm:13**
TypeError: Cannot set
properties of null (setting
'onclick')
at counter.htm:13:45

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



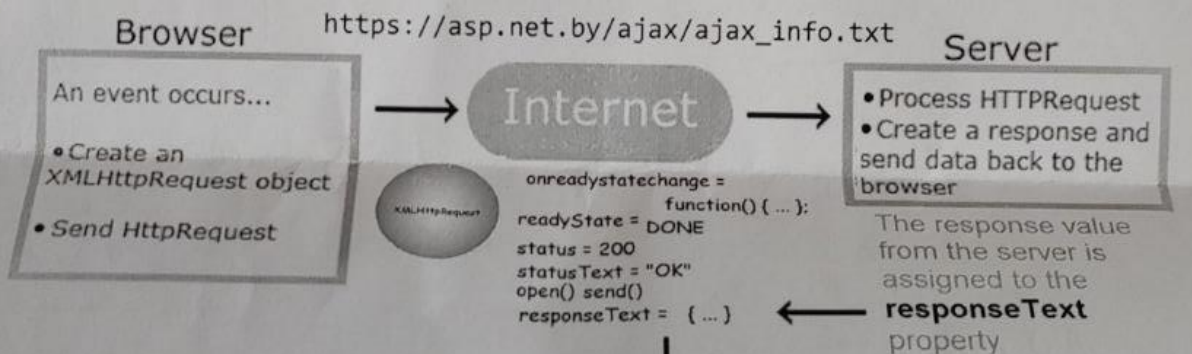
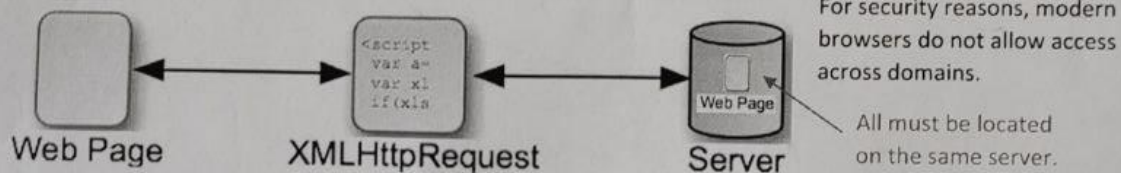
```
<button onclick="Hello()">Click
Here</button>
<script>
  let counter = 0;
  function Hello(){
    counter++;
    const heading =
      document.querySelector('h1');
    heading.innerHTML = counter;
  }
</script>
```

+0.1
2.12.25

{"managers":[<=JSON vs XML=> < managers >
 { "firstName":"Bill",
 "lastName":"Gates" },
 { "firstName":"Mike",
 "lastName":"Dell" },
]}

+0.1 K

<manager> <firstName>Bill</firstName> <lastName>Gates</lastName> </manager>	
<manager> <firstName>Mike</firstName> <lastName>Dell</lastName> </manager>	



```

<!DOCTYPE html>
<html>
<body>
<script>
function loadDoc() {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange =
  function() {
    if (this.readyState == 4 && this.status == 200) {
      document.getElementById("demo").innerHTML =
      this.responseText;
    }
  };
  xhttp.open("GET", "ajax_info.txt", true);
  xhttp.send();
}
</script>

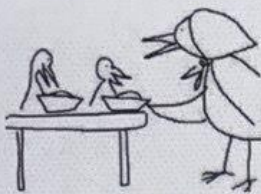
<div id="demo">
  <h2>Let AJAX change this text.</h2>
  <button type="button" onclick="loadDoc()">GO
  </button>
</div>

```

Browser

- Process the returned data using JavaScript
- Update page content

2.10.
12.0
3.30
11.20
14.50



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

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

When the page is done loading

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 it's first going to pass in as takes as input, input the first button the button then the second button then the third button

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

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

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

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

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

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

```
<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>
```

