ASP.NET MVC

INTRODUCTION

issntt@nus.edu.sg

Objectives

At the end of this lesson, students will be able to

- Describe some main tasks of servers and web browsers
- Distinguish the roles of HTML, CSS and JavaScript
- Distinguish between static and dynamic resources
- Describe the key differences when a server serves a traditional Web Browser and other types of clients
- Distinguish between .NET Core and ASP.NET Core
- Identify the correct components in .NET family when developing different platforms, e.g., Web, Desktop, and Mobile…
- Distinguish some different ASP.NET frameworks for web development
- Describe how can .NET help developers build different types of applications

Topics

- **Servers and Clients**
  - **Our focus: Web Browsers as Clients**
  - **HTTP**
- Web Browsers and HTML (with CSS, JavaScript) (Self-Study)
- Static vs Dynamic
- Web Service Overview
- ASP.NET Overview
- Some frameworks in ASP.NET

Servers vs Clients

What happens when we type *iss.nus.edu.sg* in our web browser's address bar and hit Enter?

Image by Welcome to all and thank you for your visit ! ツ from Pixabay

Servers vs Clients

PC running
Chrome browser
(Client)

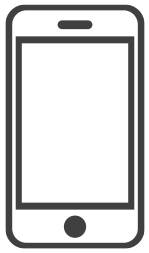
HTTP request

HTTP response
including HTML

(Server)
running a web
server

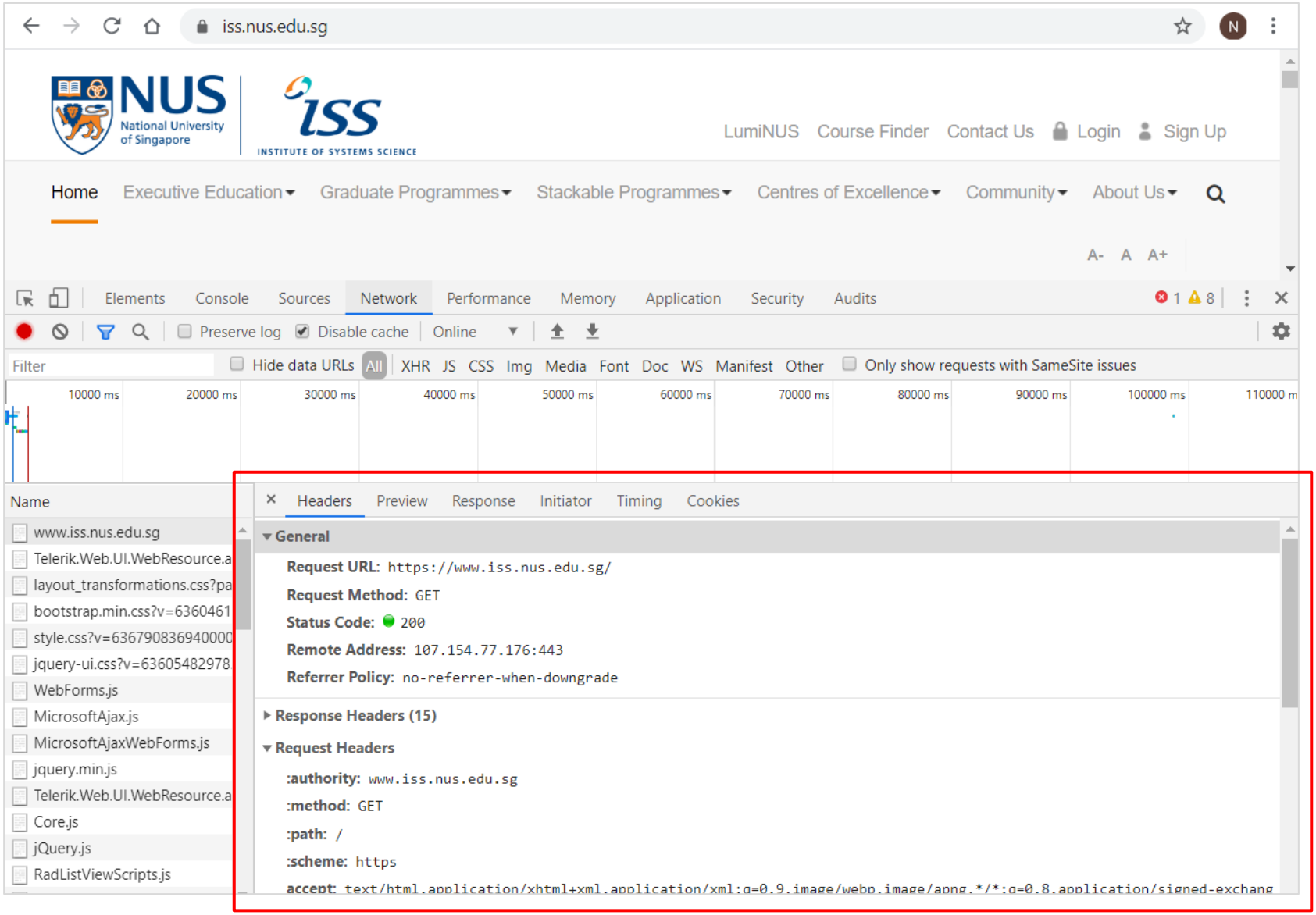
HTTP request

HTTP response
including HTML



Mobile running
Safari browser
(Client)

Servers vs Clients



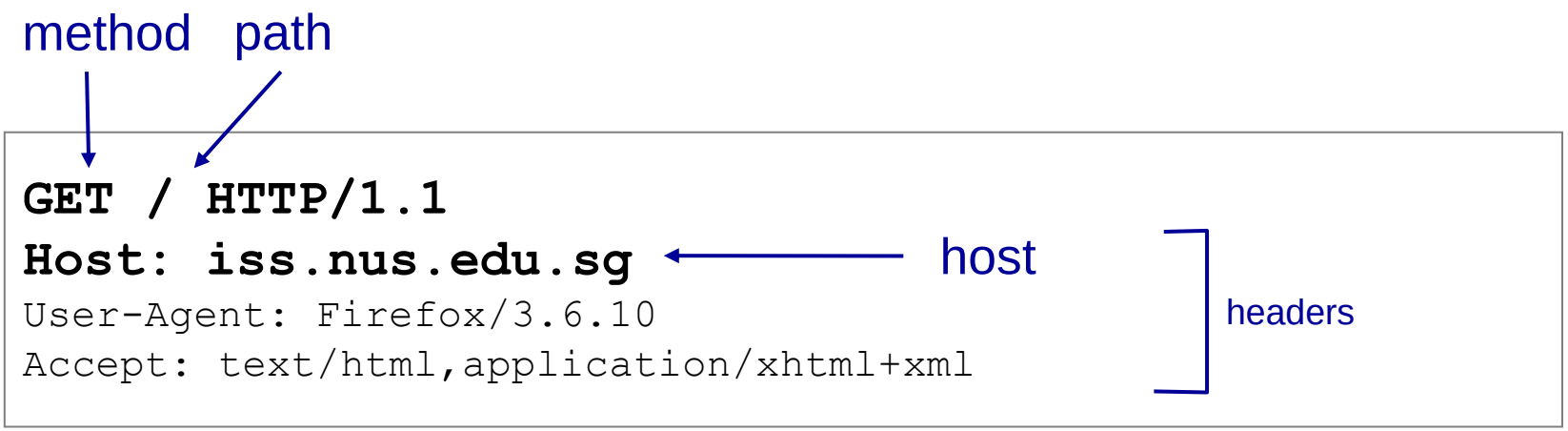
HTTP (Hypertext Transfer Protocol)

- A protocol that allows **clients** to **fetch resources**, such as HTML documents, videos, images… **from** the **servers**
- Clients and servers **communicate** by **sending messages**:
  - HTTP Request (from client to server)
  - HTTP Response (from server to client)

Image by Gerd Altmann from Pixabay

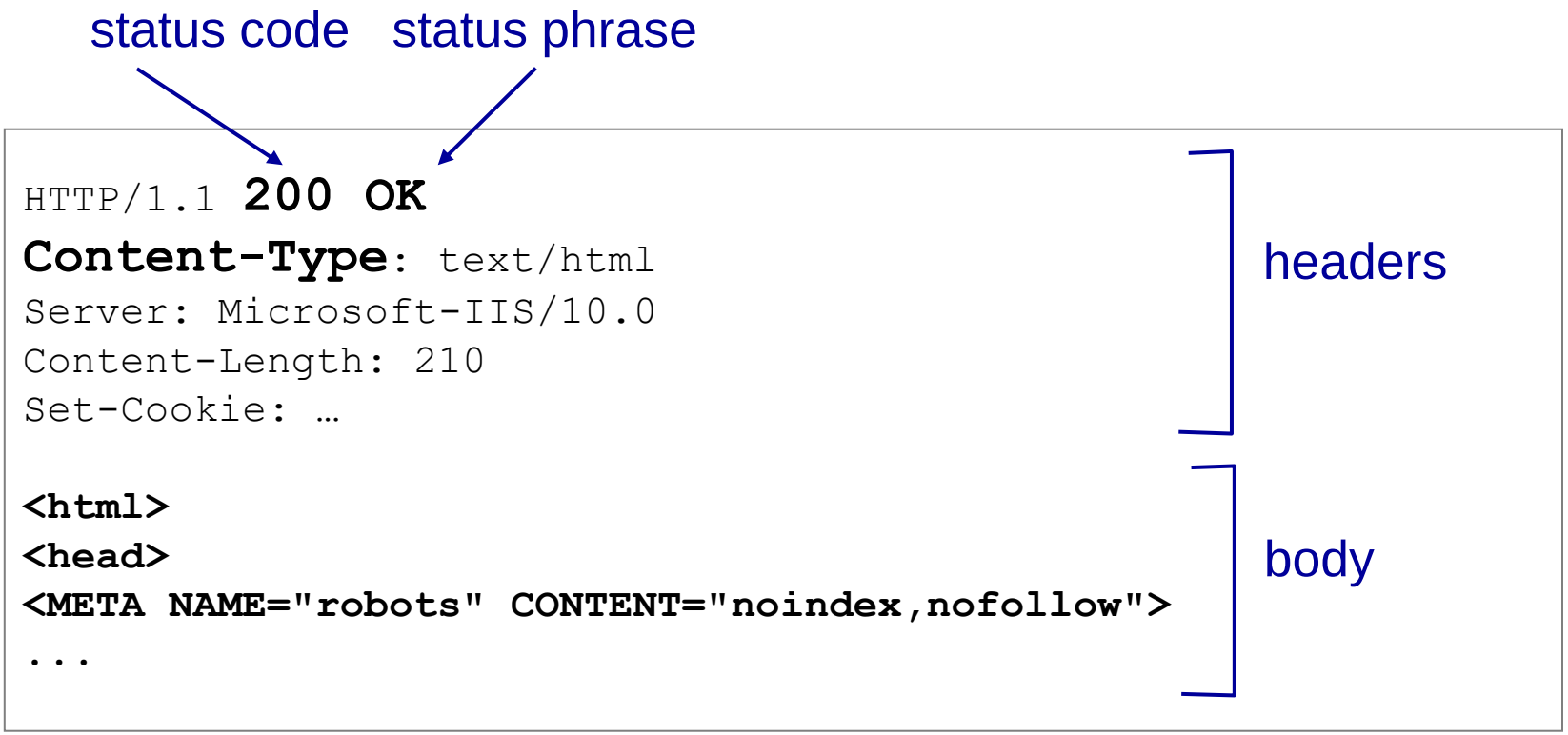
HTTP Requests

Following is a sample request when Browsers access *http://iss.nus.edu.sg*

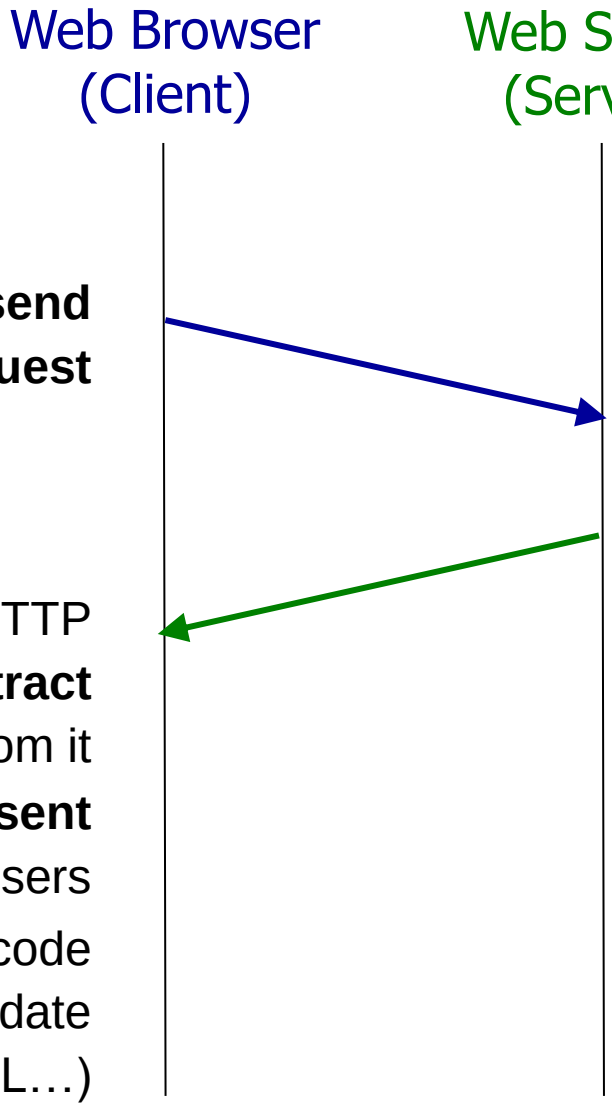


HTTP Responses

Following is a sample response when Browsers access *http://iss.nus.edu.sg*



Some Tasks of Servers and Clients



1. **Wait** for client requests

2. Prepare and **send HTTP request**

3. **Receive** and parse **HTTP request**

4. Prepare and **send HTTP response**, usually including HTML

5. Receive HTTP response and **extract HTML** from it

6. Render and **present HTML** to users

7. **Execute JS** code (handle events, validate forms, modify HTML…)

Which one, client or server, **initiates** the communication?

Next

Web Browsers receive and extract **HTML**, but why do we only see **nice looking** web-pages?

Image by Coffee Bean from Pixabay

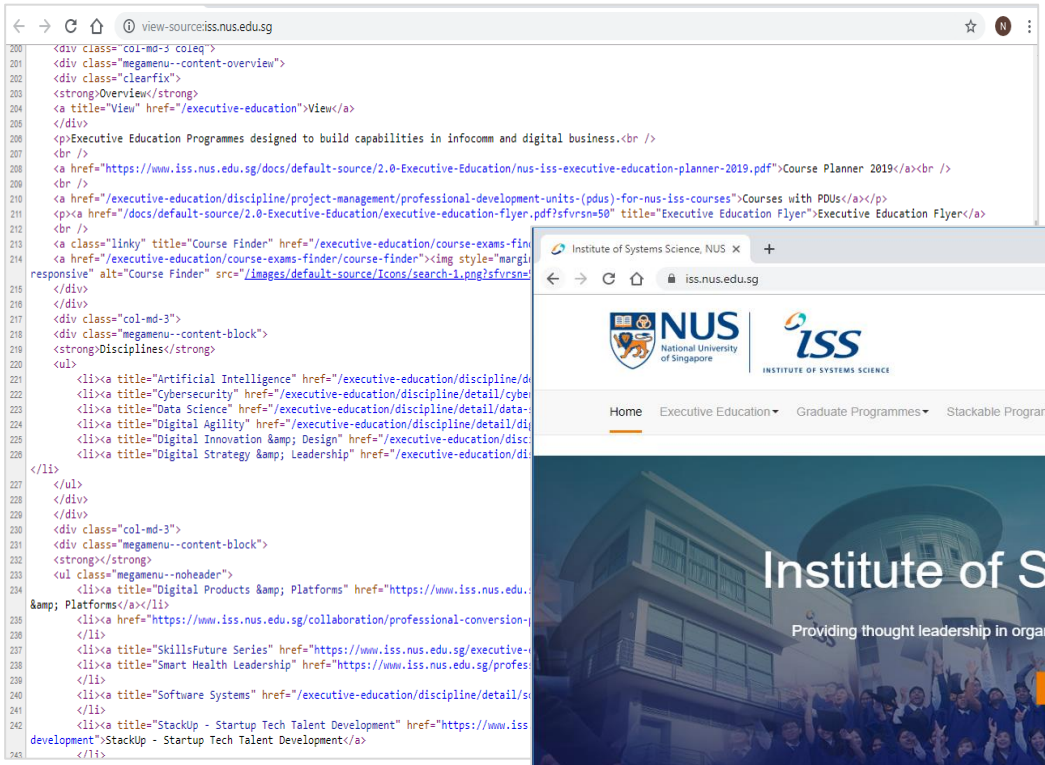
Topics

- Servers and Clients
- **Web Browsers and HTML (with CSS, JavaScript) (Self-Study)**
- Static vs Dynamic
- Web Service Overview
- ASP.NET Overview
- Some frameworks in ASP.NET

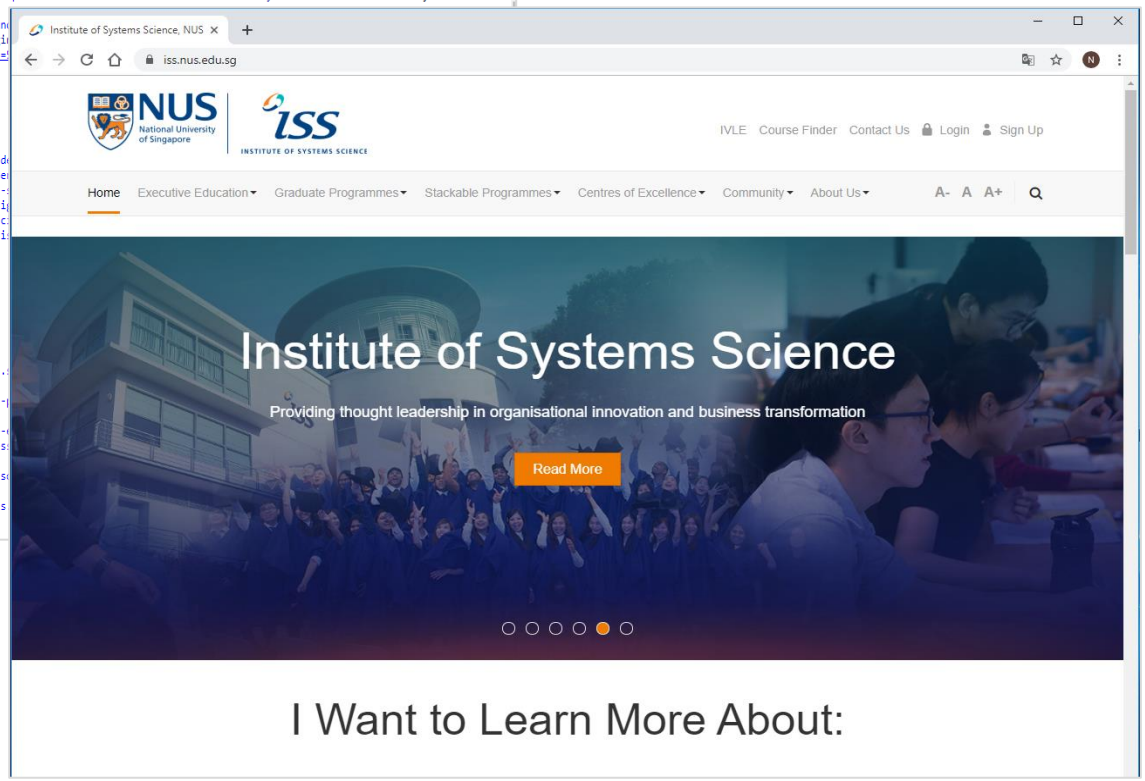
Web Browsers and HTML

Self study

Web Browsers **parse HTML**, **fetch** all **sub-resources** (images, videos…) and **render** HTML to **user-friendly** pages



When a browser receives this,



It will show this

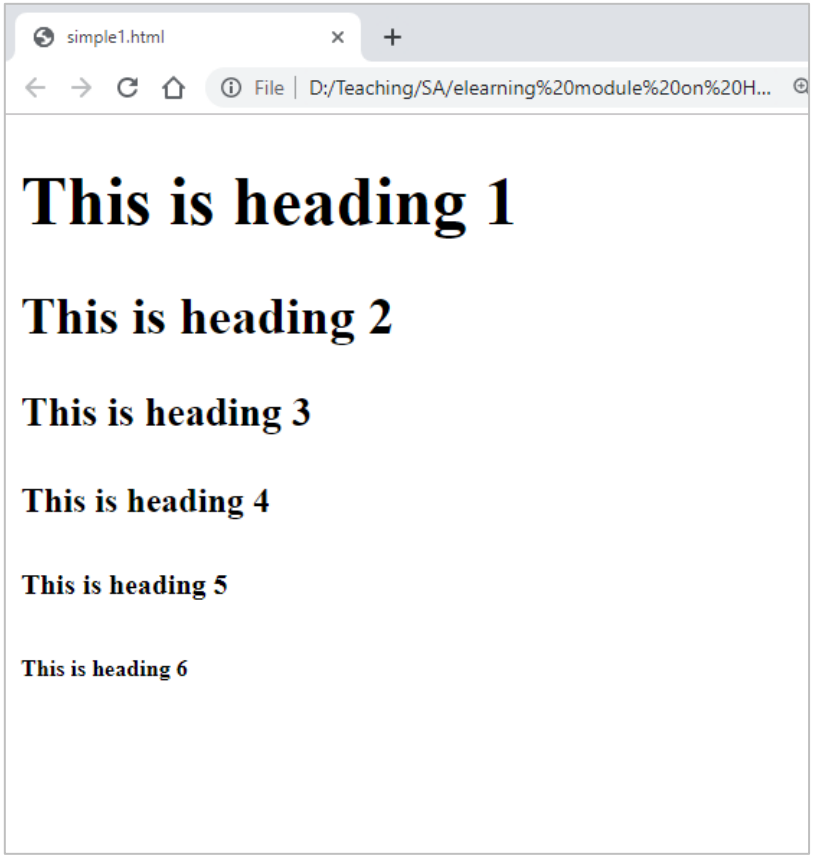
Web Browsers and HTML

Self study

```
<html>
   <body>
      <h1>This is heading 1</h1>
      <h2>This is heading 2</h2>
      <h3>This is heading 3</h3>
      <h4>This is heading 4</h4>
      <h5>This is heading 5</h5>
      <h6>This is heading 6</h6>
   </body>
</html>
```

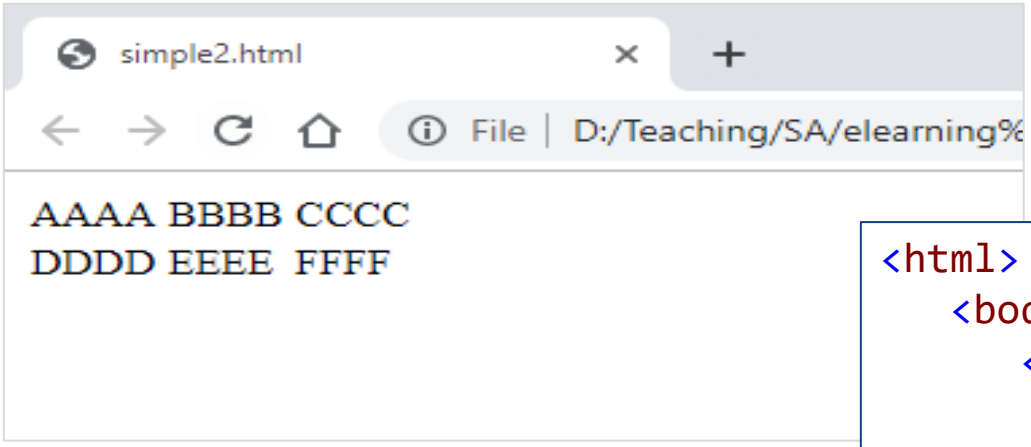
If the browser receives this,

It will show this



Web Browsers and HTML

Self study



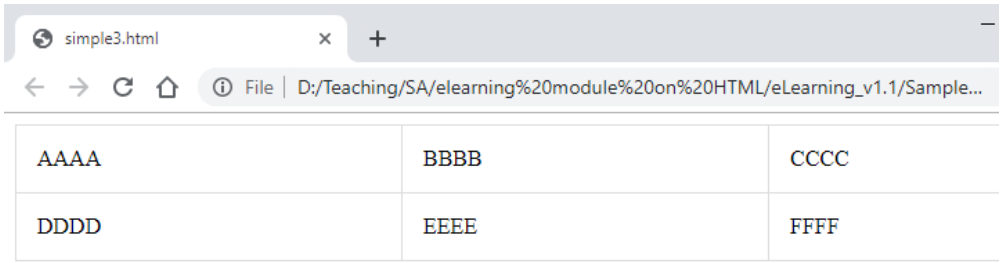
If we want the browser to show this

We need to give it this

```
<html>
   <body>
      <table>
         <tr>
            <td>AAAA</td>
            <td>BBBB</td>
            <td>CCCC</td>
         </tr>
         <tr>
            <td>DDDD</td>
            <td>EEEE</td>
            <td>FFFF</td>
         </tr>
      </table>
   </body>
</html>
```

Web Browsers and HTML (+CSS)

Self study



What do we need to do if we want to draw the **better-looking** table above?

We need to add this **CSS snippet**

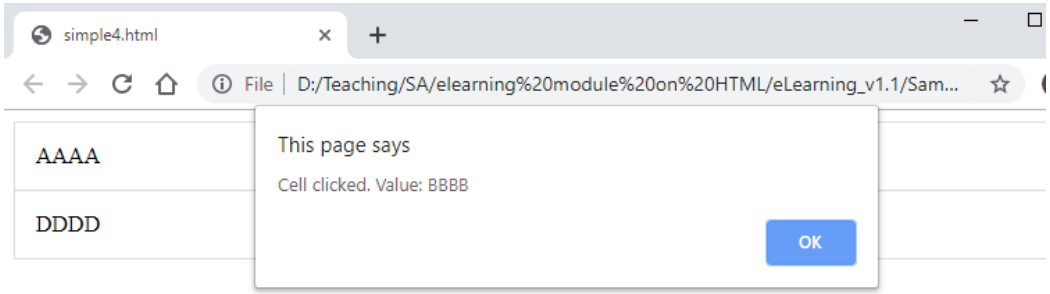
```
<style>
   table, td {
      border: 1px solid #ddd;
      text-align left;
   }

   table {
      border-collapse: collapse;
      width: 100%;
   }

   td {
      padding: 15px;
   }
</style>
```

Web Browsers and HTML (+JS)

Self study



What if we want the browser to **show** this every time **users click** on a cell?

We need to add this **JavaScript snippet**

```
<script>
  var table = document
          .getElementById("my-table");
  table.addEventListener(
              "click", function (e)
  {
    if (e.target &&
          e.target.nodeName == "TD")
    {
      alert("Cell clicked. Value: "
                + e.target.innerHTML);
    }
  });
</script>
```

HTML, CSS and JavaScript (JS)

Self study

CSS and JavaScript can be:

Embedded into HTML

```
<!DOCTYPE html>
<html>
<head>
<style>
body {color: black;}
h1   {color: blue;}
p    {color: red;}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Which one is better?

Linked from separate files

```
<!DOCTYPE html>
<html>
<head>
  <link rel="stylesheet"
      href="styles.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

```
<style>
body {color: black;}
h1   {color: blue;}
p    {color: red;}
</style>
```

styles.css

Course Pre-requisite

- This course requires you **understand and write** HTML, CSS, and JavaScript code
- By now, you are supposed to have **completed** the **e-Learning module** on HTML, CSS and JavaScript
- If you **haven't done** or have forgotten to do so, please **pick up** the knowledge **asap**

Topics

- Servers and Clients
- Web Browsers and HTML (with CSS, JavaScript) (Self-Study)
- **Static vs Dynamic**
- Web Service Overview
- ASP.NET Overview
- Some frameworks in ASP.NET

Static vs Dynamic Resources

Static

Delivered to the client **exactly as stored** in the server

Dynamic

Delivered to the client **differently** based on

- **user input** , and/or
- some **other factors**

Quiz

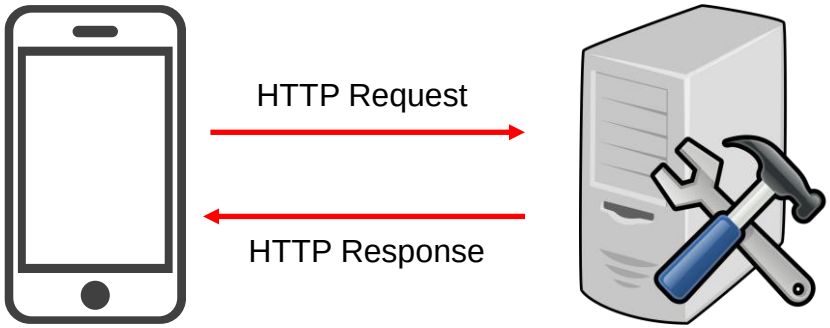
Which one(s) are more **likely** static?

Which one(s) are more **likely** dynamic?

1. An image file
2. ISS about us page
3. Google search page
4. An Amazon product page
5. A linked CSS file
6. A linked JS file
7. Your profile page after you login

Next

In the Client-Server model, **besides Web Browsers**, what **else** can be clients?



Topics

- Servers and Clients
- Web Browsers and HTML (with CSS, JavaScript) (Self-Study)
- Static vs Dynamic
- **Web Service Overview**
- ASP.NET Overview
- Some frameworks in ASP.NET

Problem

We are designing the **log in** functionality to Instagram for devices in **3 different platforms** (Windows, Android, iOS…)

What is the **fewest number** of **servers** do we need?

Web Services

Some other types of Clients besides Web Browsers

Mobile Apps

Desktop Apps

HTML5 Apps

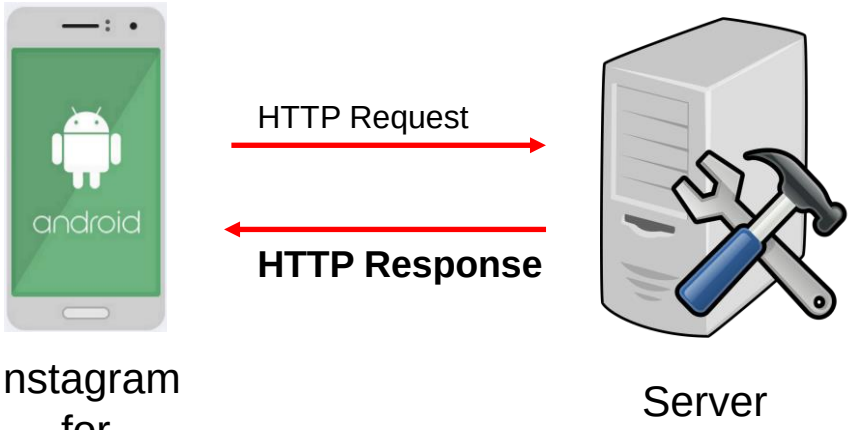
Servers that **consume another server**'s web services

In fact, **anything that can generate HTTP Requests** and send to Servers properly will be a client

Question

Compared to Web Browser, for these types of clients above, what should be **different** in the **HTTP Responses**?

Hint: besides **real data**, HTML includes many **extra things** to **please** the **human eyes**



Web Services

```
{
  "id": "C001",
  "cartItems": [
    {
      "id": "IT001",
      "product" : {
        "productId": "P001",
        "name": "Pringles",
        "unitPrice": 3.20
      },
      "quantity": 1
    },
    {
      "id": "IT002",
      "product" : {
        "productId": "P002",
        "name": "Chocolates",
        "unitPrice": 4.50
      },
      "quantity": 2,
    }
  ],
  "grandTotal": 12.20
}
```

An example of JSON data

Next

So we want to **develop web apps**. How can .NET and ASP.NET **help** us?

Topics

- Servers and Clients
- Web Browsers and HTML (with CSS, JavaScript) (Self-Study)
- Static vs Dynamic
- Web Service Overview
- **ASP.NET Overview**
  - **.NET**
  - **ASP.NET**
  - **Frameworks vs Libraries**
- Some frameworks in ASP.NET

.NET

https://www.youtube.com/watch?v=eIHKZfgddLM

What can be developed with .NET?

Our focus in this course

Web
Build web apps and services for Windows, Linux, macOS, and Docker.

Mobile
Use a single codebase to build native mobile apps for iOS, Android, and Windows.

Desktop
Create beautiful and compelling desktop apps for Windows and macOS.

Microservices
Create independently deployable microservices that run on Docker containers

Gaming
Develop 2D and 3D games for the most popular desktops, phones, and consoles.

Machine Learning
Add vision algorithms, speech processing, predictive models, and more to your apps.

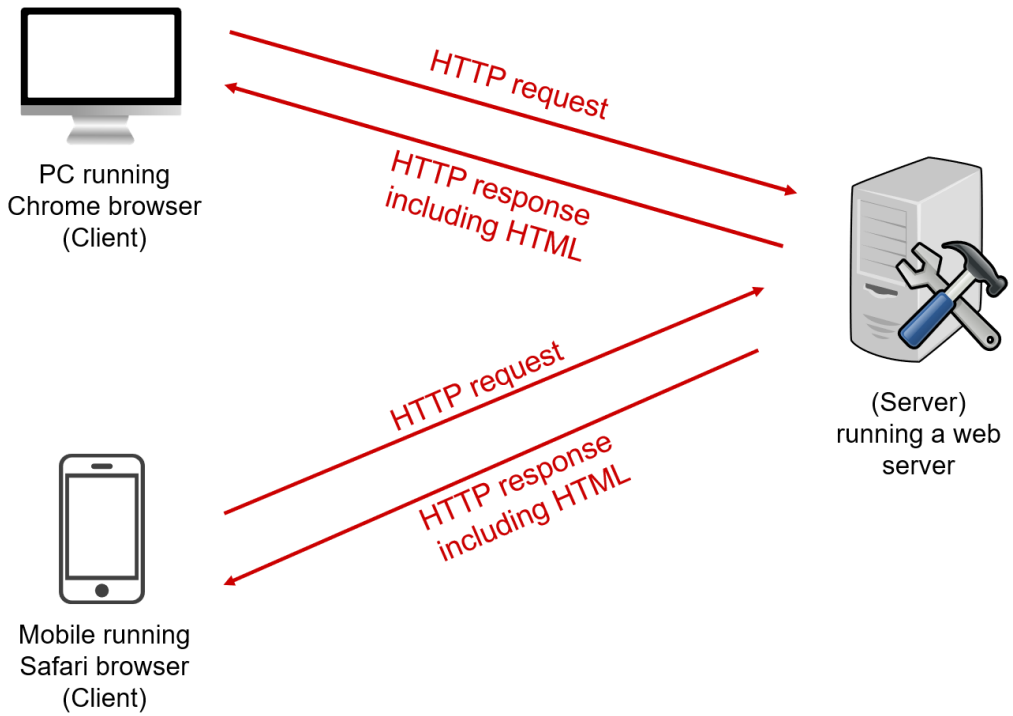
Cloud
Consume existing cloud services, or create and deploy your own.

Internet of Things
Make IoT apps, with native support for the Raspberry Pi and other single-board computers.

What technologies are we likely to use to develop **Desktop apps** for **Windows** and **macOS**?

ASP.NET

A **server-side** technology for **web development** in which **servers** do **most** of the work



Remember FOPCS and OOPCS?

1. How many **categories** of **data types**? What is the key difference between them?
2. What are **libraries**? What libraries do we usually use?
3. How do **libraries** differ from **frameworks**?

Frameworks

A framework is a piece of software which provides **generic** functionalities that can be selectively **customized** by additional code

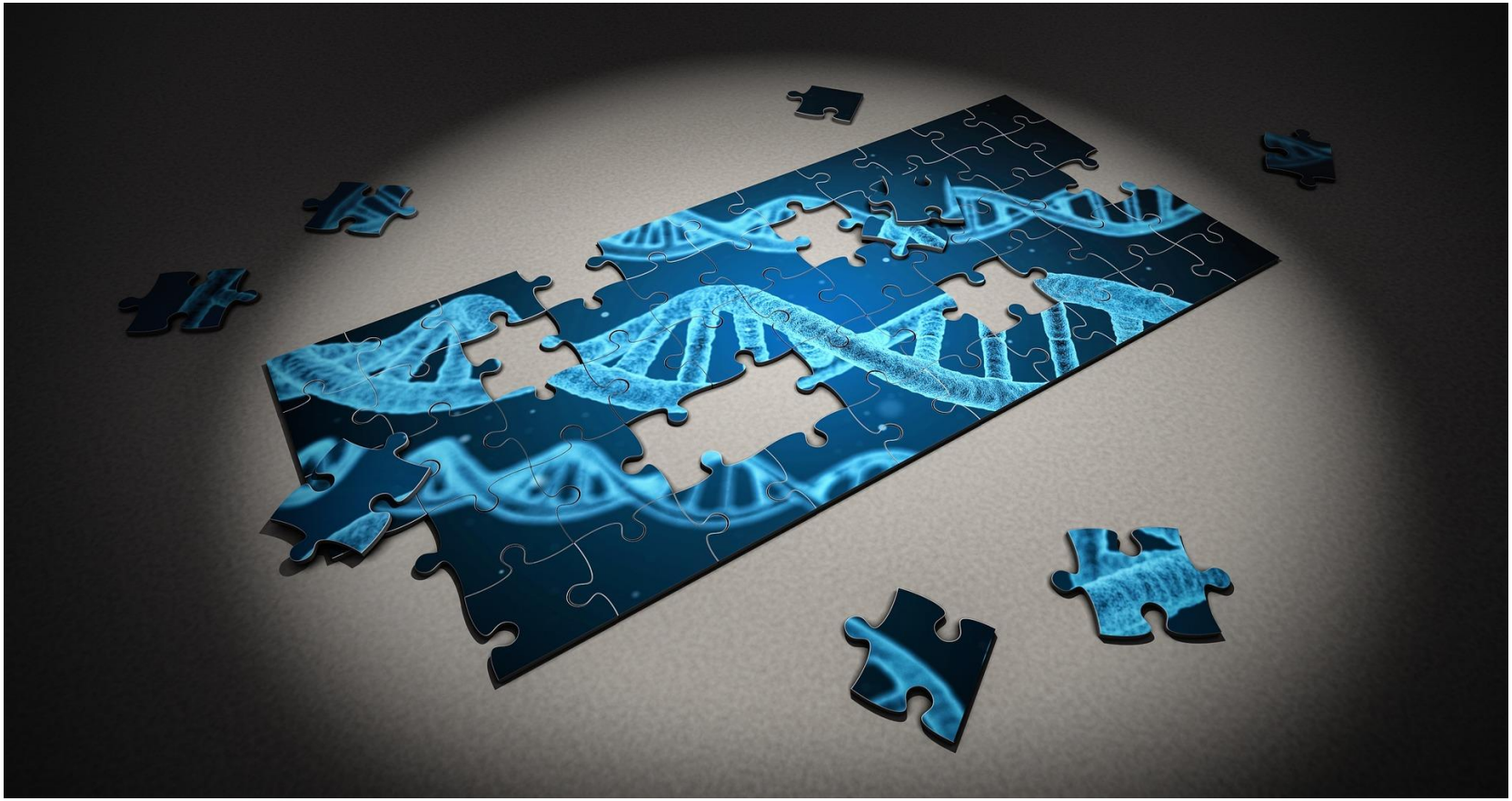


Image by Arek Socha from Pixabay

Libraries vs Frameworks

LIBRARY		FRAMEWORK	
01	More targeted	More generic	01
	Provide functionalities to solve **specific problems** (e.g. Math, string, analytics, graph…)	Provide functionalities to solve **domain/class of problems** (e.g. mobile apps, web apps…)	
02	Control is in our code	Control is in the framework	02
	Do not dictate the way we code. **Our code call** libraries whenever we need	**Dictate** the ways we code. **Frameworks call** our code at specific times.	

https://www.youtube.com/watch?v=D_MO9vIRBcA

Does a framework usually contain some libraries? Does a library usually contain some frameworks?

Topics

- Server and Client
- Web Browsers and HTML (with CSS, JavaScript) (Self-Study)
- Static vs Dynamic
- Web Service Overview
- ASP.NET Overview
- **Some frameworks in ASP.NET**

Web Frameworks

Following are some ASP.NET frameworks for **web development**

ASP.NET Web Forms	ASP.NET Framework MVC	ASP.NET Core 3.1	ASP.NET Core 6.0

ASP.NET Core 6.0 is our **focus** in this course

Readings

- Client-Server Model https://www.geeksforgeeks.org/client-server-model/
- The web standards model – HTML CSS and JavaScript https://www.w3.org/wiki/The_web_standards_model_-_HTML_CSS_and_JavaScript
- An overview of HTTP https://developer.mozilla.org/en-US/docs/Web/HTTP/Overview
- Web Service https://en.wikipedia.org/wiki/Web_service
- .NET Core, .NET Framework, Xamarin – The “What and When to use it” https://devblogs.microsoft.com/cesardelatorre/net-core-1-0-net-framework-xamarin-the-whatand-when-to-use-it/
- ASP.NET Overview https://learn.microsoft.com/en-us/aspnet/core/introduction-to-aspnet-core?view=aspnetcore-6.0