# FRONTEND DEVELOPMENT

BY: M WAQAS RASHID

# INTRODUCTION

- USER INTERFACE
- LAYOUT, STYLE, AND BEHAVIOR OF THE WEBSITE OR APPLICATION

# WHAT IS HTML, CSS, JS AND BOOTSTRAP?

# HTML

# HTML TAGS

- html
- head
- title
- body
- header
- nav
- main
- section
- article
- aside

- footer
- h1 - h6
- p, span, strong
- a
- ul, ol, li
- img
- form
- input, label, textarea, select, option, button

- div

# CSS

# CSS

## INTERNAL STYLESHEET

```
<head>
    <style>
        /* CSS */
    </style>
</head>
```

## INLINE STYLES

```
<p style = "font-size: 20px;"> This is a paragraph. </p>
```

# LINK CSS TO HTML

## Link to File

```
<head>

    <link rel = "stylesheet" type = "text/css" href = "style.css">

</head>
```

# CSS SELECTORS

```css
#newId{
    /* CSS */
}


.newClass{
    /* CSS */
}


p{
    /* CSS */
```

# CSS STYLING

CSS Text Styling:

- Size: Specifies the size of the font.
- Weight: Specifies the weight of the font (normal, bold, lighter, 100-900).
- Color: Specifies the color of the text.
- Font-Family: Specifies the font family for text.
- Line-height: Specifies the height of a line of text.
- Letter-spacing: Specifies the spacing between characters in a text.
- Word-spacing: Specifies the spacing between words in a text.
- Text-align: Specifies the horizontal alignment of text.

# CSS STYLING

CSS Backgrounds:

- Color: Specifies the background color of an element.
- Image: Specifies a background image for an element.

CSS Variables:

- Allows for the storage of values as variables to be reused throughout the stylesheet.

# CSS STYLING

CSS Border:

- Width: Specifies the width of an element's border.

- Style: Specifies the style of an element's border (solid, dotted, double, etc.).

- Color: Specifies the color of an element's border.

- Radius: Specifies the rounding of an element's corners.

- Top, Right, Bottom, Left: Specify separate border properties for each side of an element.

# CSS STYLING

Box Model:

* Margin: Specifies the outermost layer of space outside an element.

* Border: Specifies a border around an element.

* Padding: Specifies the innermost layer of space within an element.

* Width and Height: Specifies the width and height of an element.

# CSS STYLING

Position:

* Static: Default positioning, elements are placed in the normal flow of the document.
* Absolute: Elements are positioned relative to the nearest positioned ancestor (instead of the viewport, like fixed).
* Relative: Elements are positioned relative to their normal position.
* Fixed: Elements are positioned relative to the viewport and will not move with scrolling.
* Sticky: Elements are positioned relative

# CSS POSITIONS

```
position: relative;

top: 10px;

right: 20px;

bottom: 30px;

left: 40px;
```

```
position: absolute;

z-index: 1;
```

# BOOTSTRAP

# BOOTSTRAP

- A POWERFUL FRONT-END FRAMEWORK
- DEVELOPED BY TWITTER, NOW OPEN-SOURCE
- STREAMLINES THE WEB DEVELOPMENT PROCESS
- SAVES TIME AND ENHANCES PRODUCTIVITY

# KEY FEATURES OF BOOTSTRAP

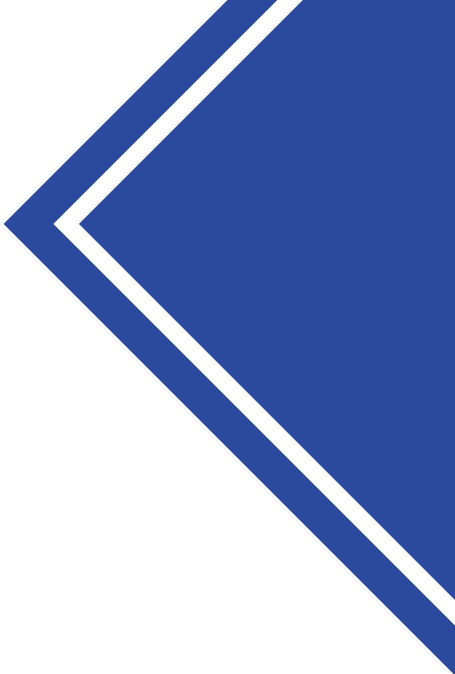- RESPONSIVE DESIGN
- PRE-STYLED COMPONENTS
- CUSTOMIZABLE THEMES

# BENEFITS OF USING BOOTSTRAP

- TIME EFFICIENCY
- CROSS-BROWSER COMPATIBILITY
- COMMUNITY AND SUPPORT

# SETTING UP BOOTSTRAP

- **VISIT THE OFFICIAL BOOTSTRAP WEBSITE (GETBOOTSTRAP.COM)**

- **Copy the appropriate link tag from the "CSS" section**

- **Paste the link tag in the head section of your HTML file**

# ADDING JAVASCRIPT DEPENDENCY

- TO ENABLE BOOTSTRAP'S JAVASCRIPT FUNCTIONALITY, ADD A LINK TO THE BOOTSTRAP JS FILE

- Copy the appropriate link tag from the "JS" section

- Place the script tag before the closing body tag (</body>) of your HTML file

# JAVASCRIPT

# WAYS TO PRINT IN JS

CONSOLE.LOG("HELLO WORLD!")
document.write("This will be written like paragraph ")
alert("testing")
prompt("FeedBack") --> will show a textbook with label(FeedBack)

# DATATYPES IN JAVASCRIPT

**TWO TYPES OF DATATYPES IN JAVASCRIPT :-**
- Primitive Data Types: undefined, null, number, string, boolean, symbol
- Reference Data Types: arrays and objects

**LET** (ONLY CAN USE IN A BLOCK)
**var** (for every datatype also)
**const** (for keeping secret)

# CONT...

**NUMBERS:**

var num1 = 55;

var num2 = 55.55;

**BOOLEANS:**

var a = true;

var b = false;

**STRING:**

var str1 = "This is a string";

var str2 = 'This is also a string';

var str3 = `This is alsooooo a string`;

# CONT...

**OBJECTS:**
VAR MARKS = {
   WAQAS: 99,
   USMAN: 90,
   AFFAN: 10
}

**UNDEFINED:**
VAR UND = UNDEFINED;
VAR UND1;
VAR N = NULL;

**ARRAYS:**
VAR ARR = [1, 2, "WQS", 4, 5];

# OPERATORS IN JS

**ARITHEMATIC OPERATORS** $(+,-,*,/)$
**assigment operators** $(=,-=,*=)$
**comparison operators** $(==,>=,<=,!=,===,!==)$
**logical operators** $(\&\&,||)$

# FUNCTIONS AND CONDITIONS

```
function sum(a, b) {
  return a + b;
}
```

```
if ,else if ,else (same as c++,c#)
```

# LOOPS

**For loop:**

```
for (var i = 0; i < arr.length; i++) {


}
```

**Foreach Loop:**

```
arr.forEach(function (element) {
  // console.log(element);
})
```

# ARRAYS

```javascript
let i = [1,"waqas",true,null,34.5];
i.length; // give length
i.pop(); // del last element of array
i.push("hehe"); // add on last
i.shift(); // del 1st element
i.unshift("Hehe"); // add on first place
const newlen = i.unshift("Hehe");
console.log(newlen); // print new array length
i.toString(); // convert all to string
i.sort(); // first convert to string and sort like wise
```

# STRING METHODS

```
let mystring = "Im a good boy! good good";
console.log(mystring.length);
console.log(mystring.indexOf("good"));
console.log(mystring.lastIndexOf("good"));
console.log(mystring.slice(0,3));
console.log(mystring.replace("good","bad"));
```

# DATE METHODS

```javascript
let currentDate = new Date();
console.log(myDate.getTime());
console.log(myDate.getFullYear());
console.log(myDate.getDay());
console.log(myDate.getMinutes());
console.log(myDate.getHours());
```

# DOM MANIPULATION

```
document.getElementById('click').click()
document.getElementById('click').style.border = 'blue'
document.getElementById('click').style.border= '2px solid blue'

Let elem = document.getElementById('click');
console.log(elem);
Let elemClass = document.getElementsByClassName("container"
console.log(elemClass);
```

# CONT...

```
elemClass[0].style.background = "yellow";
elemClass[0].classList.add("bg-primary");
elemClass[0].classList.remove("bg-primary");
elemClass[0].innerHTML
elemClass[0].innerText
```