# Programming with JavaScript

# Ice breaker

# Introduction to JavaScript

Session 1

# Objectives

By the end of this session, students will learn to:

- Define JavaScript
- Differentiate between JavaScript and Java
- Describe Web Applications and Websites
- Compare and contrast between server-side and client-side scripting
- Explain basic concepts of JavaScript

# Introduction to JavaScript

JavaScript is a scripting language.

Mainly used in Web pages to improve design, validate forms, detect browsers, and create cookies.

By default, all Web browsers have in-built JavaScript support.

# HTML, CSS, and JavaScript

**HTML**

- Is used to form the skeleton of a Website
- Acts as a building block of Web Scripting
- Describes the arrangement of the content

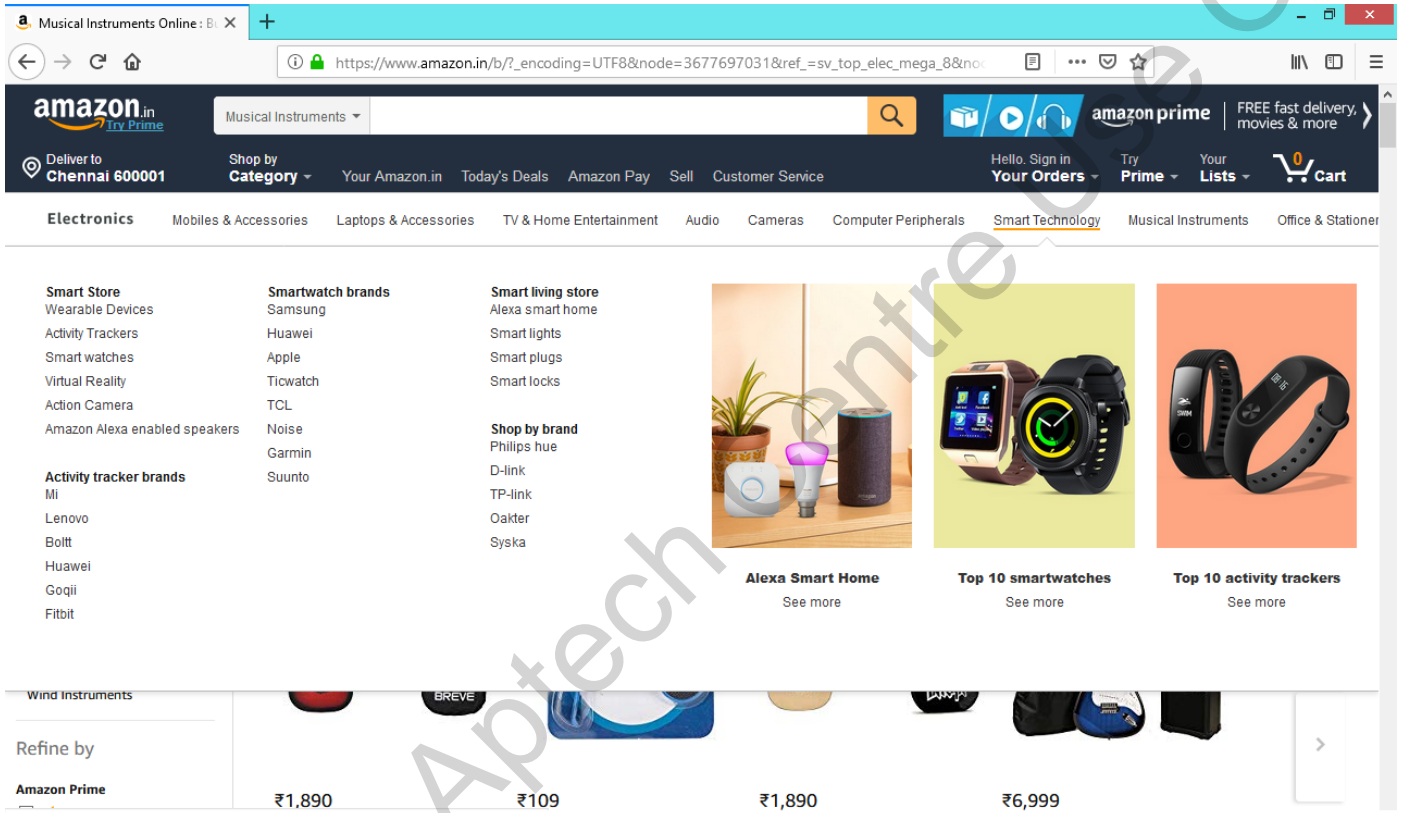
**Cascading Style Sheets (CSS)**

- Is used to create an appealing Website
- Helps design graphic elements that improve the appearance of the Websites
- Defines the color, size, and other visual aspects

**JavaScript**

- Allows Web pages to be more interactive and independent
- Enables the code to be embedded in HTML markup or stored in a separate file

# Example Website



# JavaScript and Java

| JavaScript | Java |
| --- | --- |
| Is a scripting language with light-weight components | Is an OOP language with advanced features |
| Needs only the browser engine to be interpreted while loading a Web page | Needs interpreters to convert it into a machine language for a computer system to understand |
| Has been developed by Netscape | Has been developed by Sun Microsystems |
| Has some implications with different Web browsers | Ensures that programs written in Java run exactly the same across different systems |
| Needs to be interpreted | Needs to be compiled |

# Websites and Web Applications

**Website**

- Usually contains more of static information and less of user interaction.
- Will not take any input from the user and does not interact with the Web server.
- Can be built using HTML, CSS, and JavaScript.

**Web Application**

- Is more interactive and require a connection with a Web server.
- Is dynamic as it takes input from users, interacts with the server, and provides response back to users.
- Can be built with HTML, CSS, JavaScript, and programming languages.

# Server-Side Script versus Client-Side Script

## Server-side Script

- It runs on a Web server.
- Additional components such as Webserver, Application server, and Database are involved.
- Server provides results for user's dynamic requests.
- Specific response is sent back for each user's request.

## Client-side Script

- It runs on a Web browser.
- Browser is the only component involved.
- Response is faster; server load is reduced.
- Code runs on browser that is downloaded from the Web server

# JavaScript Data Types

Data is classified into six major data types in JavaScript:

| | |
|---|---|
| Numbers | • The number type can be used both for integer and floating point numbers such as var x=1234; and var y=12.34; |
| Strings | • One or more character values are represented by the string type.<br>• For example: var a="abcd"; and var b=' '; |
| Boolean | • Boolean data type represents a true or false value.<br>• For example: var outcome = true; and Boolean(2>4); |
| Objects | • Objects are special data types used for the collection of various data.<br>• For example: var vehicle = {type: "Suzuki", model: "Baleno", color: "Ray Blue"}; |
| Functions | • A function is a block of code that is used to perform a specific task.<br>• For example: function mul(n1,n2) {return n1*n2;} |
| Undefined | • Undefined variables are those that do not have any data type such as number or string.<br>• For example: var undef; |

# JavaScript Variables

- Variable helps us to store values.
- A variable name is used to assign values.
- The value changes during the execution of a program.

Example:

```
var a= 2;
var b= 4;
var c= a+ b;
```

Output:

```
The value of c is: 6
```

# Control Statements 1-2

Control statements are blocks of code that perform a specific operation based on conditions provided.

## if … else

- **Definition:**
  - The 'if' statement will execute a block of code when a required condition is satisfied.

- **Syntax:**

```
if (condition)
    {
    Statements to be executed If
    condition is true
    }
```

## switch…case

- **Definition:**
  - The 'switch' statement checks different conditions. It checks for each case against the condition until a correct match is found.

- **Syntax:**

```
switch (condition)
{
case condition 1: statements; break;
case condition 2: statements; break;
...
case condition n: statements; break;
default: statements;}
```

# Control Statements 2-2

## do…While

- **Definition:**
  - The 'do...while' loop checks for the condition at the end of the loop.
  - The 'do' loop will be executed at least once, even if the condition is false.

- **Syntax:**

```
do
{
Statements to be
executed;
}
while (condition);
```

## While

- **Definition:**
  - The 'while' loop executes a statement or a block of code repeatedly as long as the condition is true.
  - When condition is false, the loop will exit.

- **Syntax:**

```
while (condition)
{Statements to be
executed if condition
is true}
```

## For loop

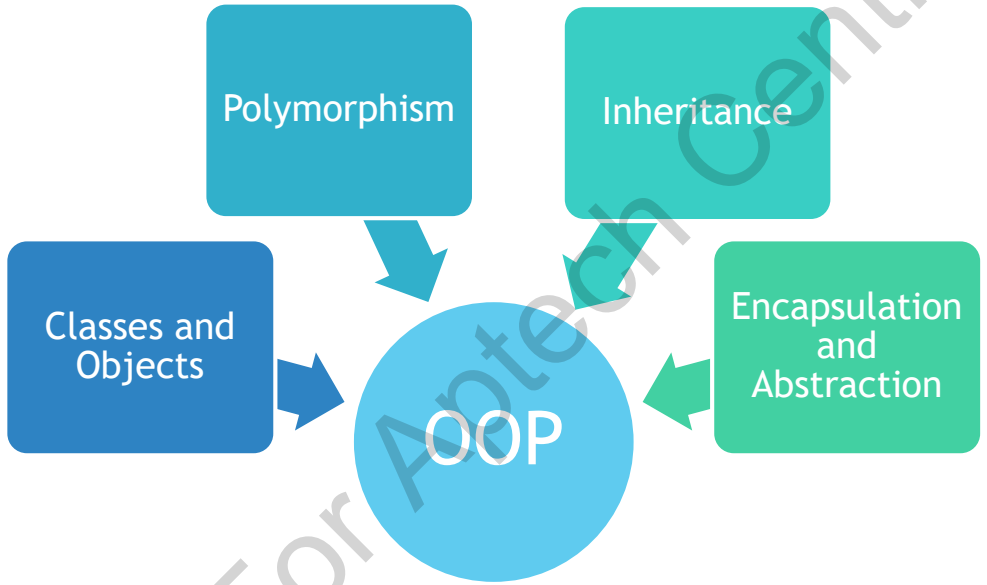
- **Definition:**
  - The 'for' loop is used to execute a block of code multiple times for a specific condition.
  - It has three parts: the loop initialization, the condition is true or not, and the iteration statement that can increase or decrease the counter.

- **Syntax:**

```
for (initialization;
condition; iteration
statement)
{Statements to be
executed if test
condition is true}
```

# Object-Oriented Programming (OOP)

- In OOP, the major building block of a script is an object.
- Each object is a scripting unit consisting of a state (variables) and behavior (methods).
- JavaScript is an OOP-based scripting language.



# Creating Classes and Objects in JavaScript

- A class is similar to a function, but a class should be declared before accessing it in the script.
- Objects can be created using one of the three ways:

Using object literal

Using new keyword to create instance of Object directly

Using an object constructor and new keyword

# JavaScript Modules

A module is a piece of code that returns a specific value can be written in a separate file and imported in the script.

JavaScript has standard syntax for using modules within a script:

- **export** keyword can be used to export any data.
- A constant, a function, and any other variable can be exported.
- **import** keyword can be used to import the module from another module.
- repeat and ex functionality from the library module is used in the main module.

Modules differ from standard scripts.

- HTML-style comment syntax is not supported in modules.
- The **export** and **import** syntax is only available within modules.

# Session Summary

- JavaScript is a scripting language used in Websites and Web applications.
- HTML is often used to form the skeleton of a Website.
- A static Website will not take any input from the user nor interact with the Web server.
- Web applications are interactive as compared to static Websites and require a connection with a Web server.
- Server-side scripts run on a Web server that takes input from the client and interacts with it.
- Client-side scripts run on a Web browser that acts as an environment and processing of the script happens at the end user's computer.