**LESSON 10 : ADVANCED WEB DEV - CLASSES AND OBJECTS**

**Teacher Prep:**

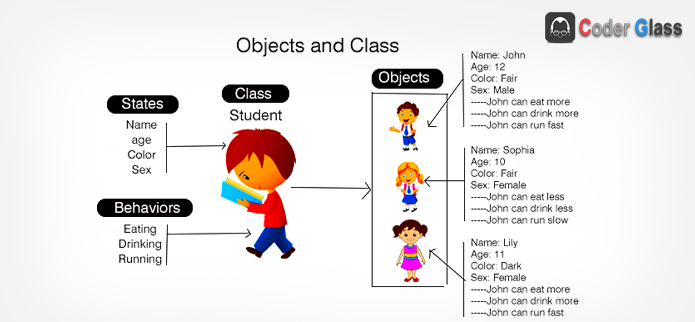
* Go through the Document thoroughly and create the projects as guided.

**Project Showcase: (to be done at the start of the class at teacher end)**

* Open the provided project on VS Code before the class starts and check the live view on your browser.

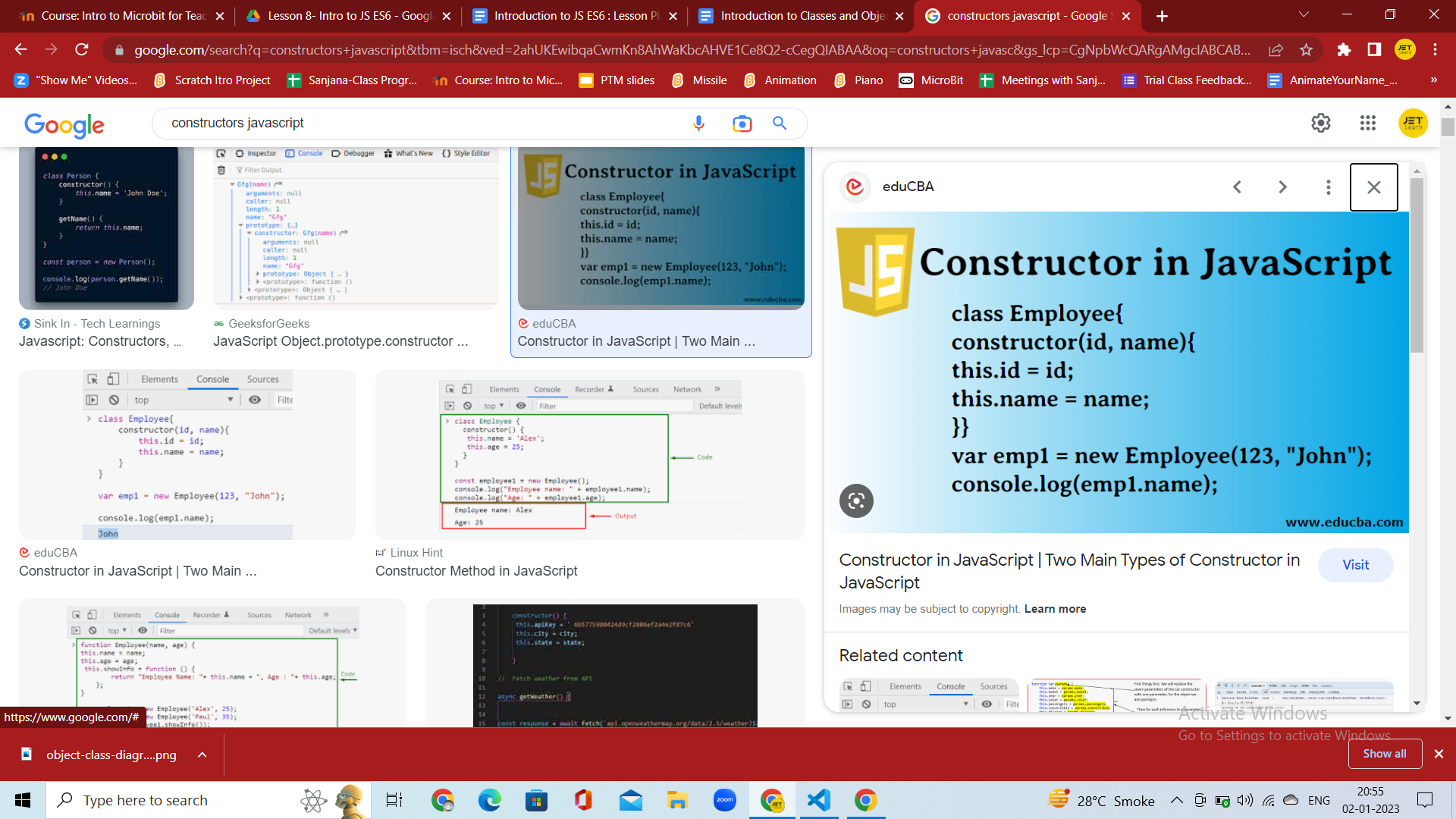
**What is Classes and Objects?**

* A class is a non-primitive or user-defined data type, while an object is an instance of a class.
* An object is an identifiable entity with some characteristics, state and behaviour.
* A class is a group of objects that share common properties and behaviour.
* We can explain Classes and Objects with real world examples as shown in picture.

****

**Learn more about Constructors and Inheritance :**

[**https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/Object-oriented\_programming**](https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Objects/Object-oriented_programming)

****

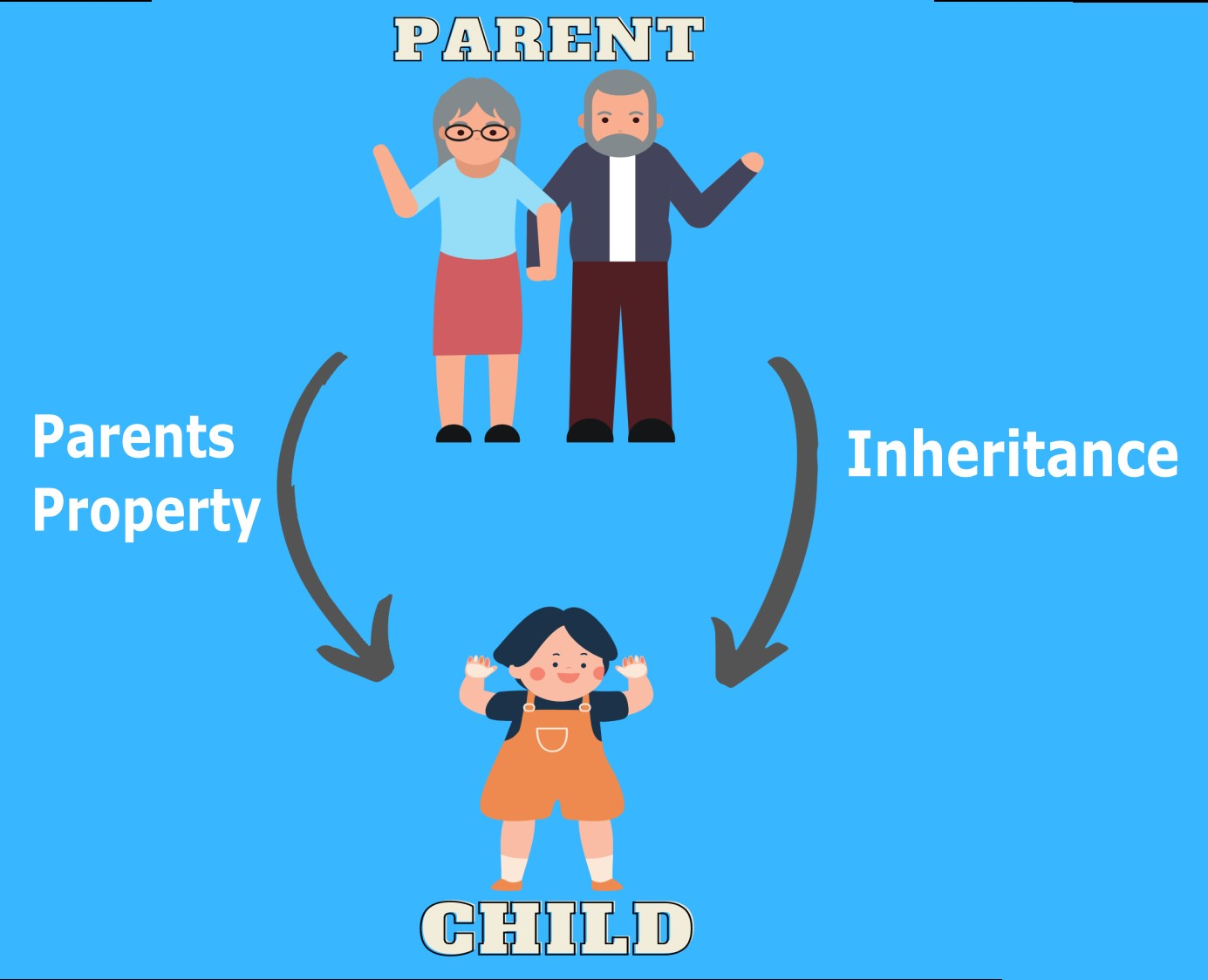
A constructor is a special function that creates and initializes an object instance of a class.

**Inheritance:**

Inheritance enables you to define a class that takes all the functionality from a parent class and allows you to add more.

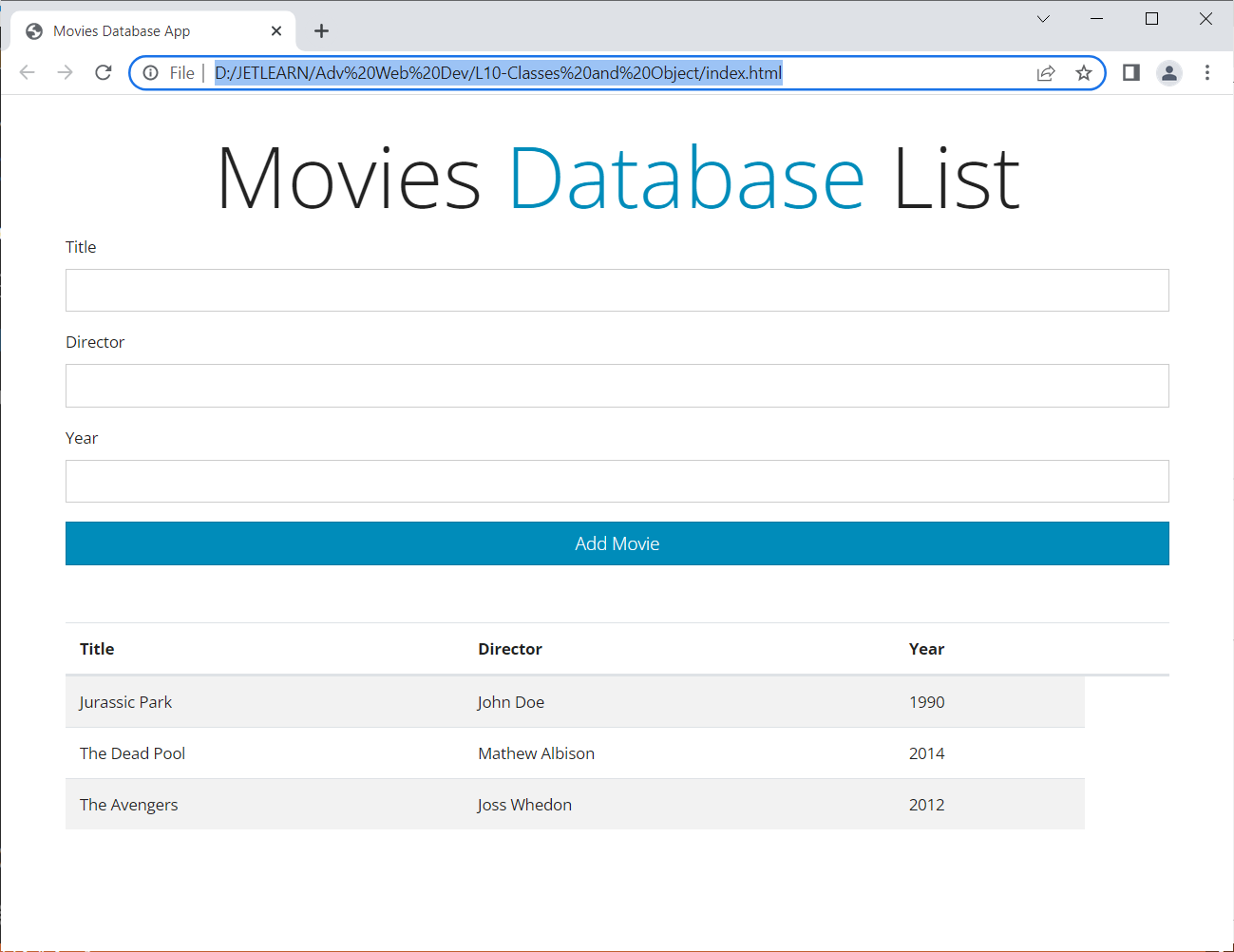
Using class inheritance, a class can inherit all the methods and properties of another class.

Inheritance is a useful feature that allows code reusability.



**Movie Database Project:**

**Output:**

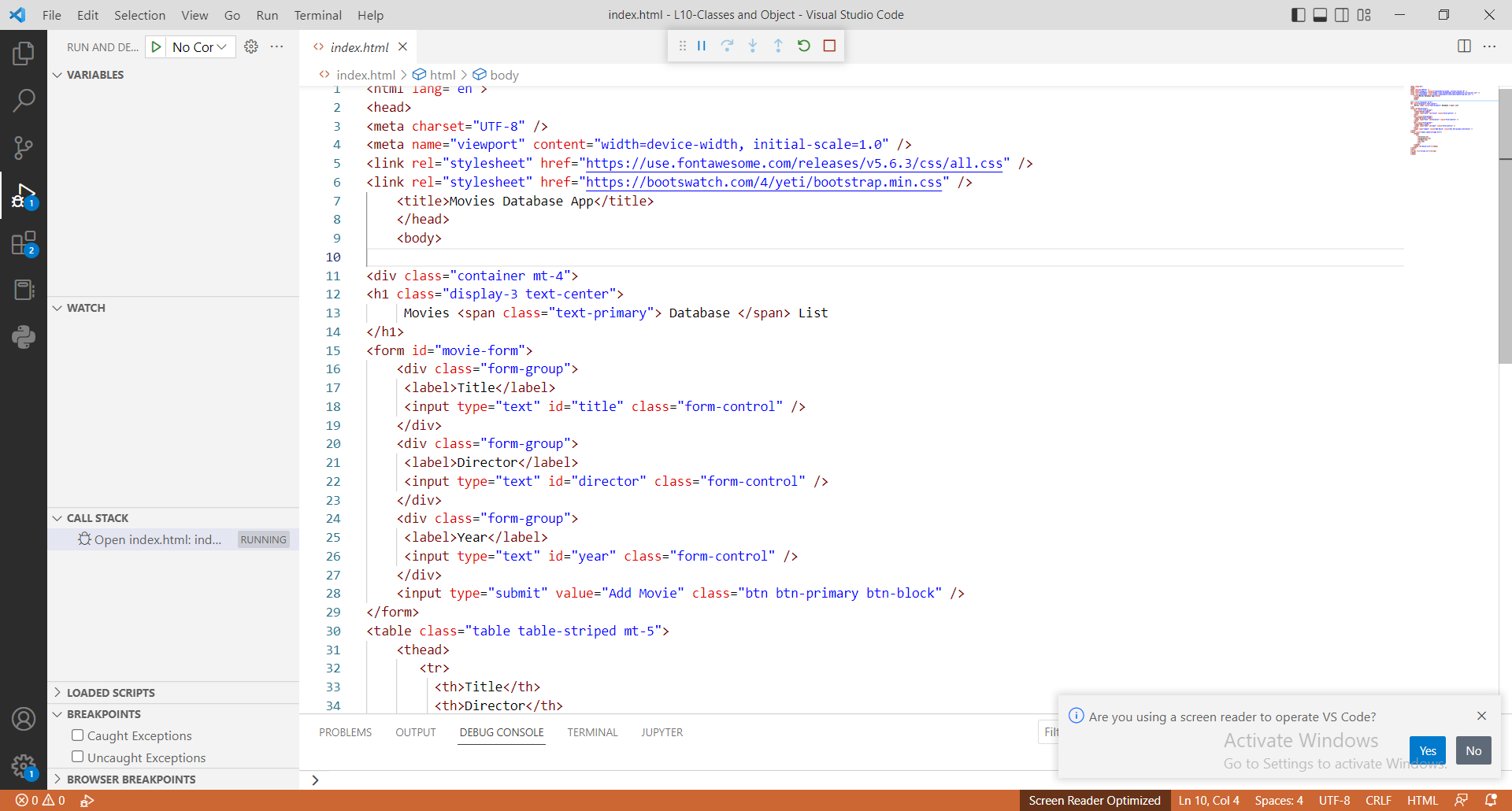


Main aim of this project is to explain the usage of Classes and Objects in Web Development.

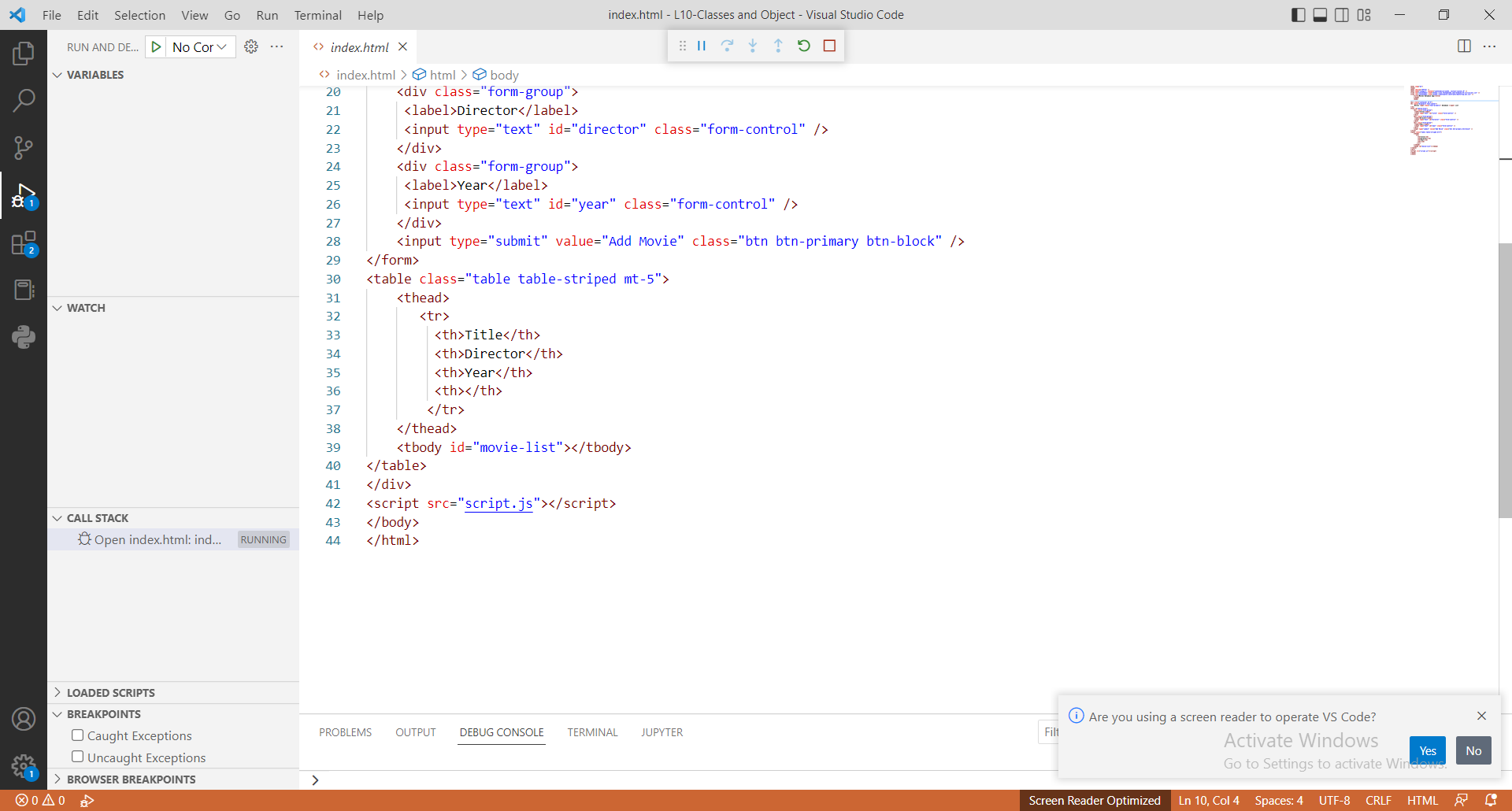
We will be explaining Inserting and Viewing the details using Classes and Objects.

***For Slow Learner: Give the Starter Project code i.e Index.html and focus on only JS part.***

Create a form structure HTML code to Insert data from the user.



Adding Table HTML code to display the information:



**Script.JS**

Create a class of Movie with constructor to store the pattern of information:

class Movie {

constructor(title, director, year) {

this.title = title

this.director = director

this.year = year

}

}

Below is an Array of information to be displayed prior when the page loads: *(can be skipped in the beginning and added later on)*

const defaultMovies = [

{

title: 'Jurassic Park',

director: 'John Doe',

year: '1990'

},

{

title: 'The Dead Pool',

director: 'Mathew Albison',

year: '2014'

},

{

title: 'The Avengers',

director: 'Joss Whedon',

year: '2012'

}

]

Create another Class which has functions to Add movie and Display information:

*Add appropriate ID names with reference to index.html page.*

A new row will generate in table with **createElement** function.

**Appenchild** function will append the newly added information to the previous list.

class UI{

static addMovieToList = (movie) => {

const list = document.getElementById('movie-list');

const row = document.createElement('tr')

row.innerHTML=`<td>${movie.title}</td><td>${movie.director}</td><td>${movie.year}</td>`

list.appendChild(row)

}

static displayMovies=()=>{

defaultMovies.forEach(movie => UI.addMovieToList(movie))

}

}

Call the function to display information from the UI class:

UI.displayMovies()

Adding the information from form when Submit button is Clicked:

document.querySelector('#movie-form').addEventListener('submit', addAMovie)

function addAMovie(e) {

// prevent actual submission

e.preventDefault()

// Get Form Values

const title =document.querySelector('#title').value;

const director =document.querySelector('#director').value;

const year =document.querySelector('#year').value;

// Instantiate a new Movie object

const movie = new Movie(title,director,year)

// Add object to UI

UI.addMovieToList(movie)

}

**Homework:** Explore more about Classes and Objects.

Add the DELETE data feature to the webpage to delete a particular entry and code the same.

You can add a Delete button beside each row.