***Vanilla JavaScript Basics:***

// LOGGING OUTPUT

alert('Hello World'); // Do not use for debugging. Stops script and only strings

console.log('Hello World');

console.error('This is an error');

console.warn('This is a warning');

// VARIABLES - var, let, const

let age = 30;

// let can be re-assigned, const can not

age = 31;

// DATA TYPES - String, Number, Boolean, null, undefined

const name = 'Brad';

const age = 37;

const rating = 3.5;

const isCool = true;

const x = null;

const y = undefined;

let z; // undefined

// Check type

console.log(typeof z);

// STRINGS

// Concatenation

console.log('My name is ' + name + ' and I am ' + age);

// Template literal (better)

console.log(`My name is ${name} and I am ${age}`);

// String methods & properties

const s = 'Hello World';

let val;

// Get length

val = s.length;

// Change case

val = s.toUpperCase();

val = s.toLowerCase();

// Get sub string

val = s.substring(0, 5);

// Split into array

val = s.split('');

// ARRAYS - Store multiple values in a variable

const numbers = [1,2,3,4,5];

const fruits = ['apples', 'oranges', 'pears', 'grapes'];

console.log(numbers, fruit);

// Get one value - Arrays start at 0

console.log(fruits[1]);

// Add value

fruits[4] = 'blueberries';

// Add value using push()

fruits.push('strawberries');

// Add to beginning

fruits.unshift('mangos');

// Remove last value

fruits.pop();

// // Check if array

console.log(Array.isArray(fruits));

// // Get index

console.log(fruits.indexOf('oranges'));

// OBJECT LITERALS

const person = {

firstName: 'John',

age: 30,

hobbies: ['music', 'movies', 'sports'],

address: {

street: '50 Main st',

city: 'Boston',

state: 'MA'

}

}

// Get single value

console.log(person.name)

// Get array value

console.log(person.hobbies[1]);

// Get embedded object

console.log(person.address.city);

// Add property

person.email = 'jdoe@gmail.com';

// Array of objects

const todos = [

{

id: 1,

text: 'Take out trash',

isComplete: false

},

{

id: 2,

text: 'Dinner with wife',

isComplete: false

},

{

id: 3,

text: 'Meeting with boss',

isComplete: true

}

];

// Get specific object value

console.log(todos[1].text);

// Format as JSON

console.log(JSON.stringify(todos));

// LOOPS

// For

for(let i = 0; i <= 10; i++){

console.log(`For Loop Number: ${i}`);

}

// While

let i = 0

while(i <= 10) {

console.log(`While Loop Number: ${i}`);

i++;

}

// Loop Through Arrays

// For Loop

for(let i = 0; i < todos.length; i++){

console.log(` Todo ${i + 1}: ${todos[i].text}`);

}

// For...of Loop

for(let todo of todos) {

console.log(todo.text);

}

// HIGH ORDER ARRAY METHODS (show prototype)

// forEach() - Loops through array

todos.forEach(function(todo, i, myTodos) {

console.log(`${i + 1}: ${todo.text}`);

console.log(myTodos);

});

// map() - Loop through and create new array

const todoTextArray = todos.map(function(todo) {

return todo.text;

});

console.log(todoTextArray);

// filter() - Returns array based on condition

const todo1 = todos.filter(function(todo) {

// Return only todos where id is 1

return todo.id === 1;

});

// CONDITIONALS

// Simple If/Else Statement

const x = 30;

if(x === 10) {

console.log('x is 10');

} else if(x > 10) {

console.log('x is greater than 10');

} else {

console.log('x is less than 10')

}

// Switch

color = 'blue';

switch(color) {

case 'red':

console.log('color is red');

case 'blue':

console.log('color is blue');

default:

console.log('color is not red or blue')

}

// Ternary operator / Shorthand if

const z = color === 'red' ? 10 : 20;

// FUNCTIONS

function greet(greeting = 'Hello', name) {

if(!name) {

// console.log(greeting);

return greeting;

} else {

// console.log(`${greeting} ${name}`);

return `${greeting} ${name}`;

}

}

// ARROW FUNCTIONS

const greet = (greeting = 'Hello', name = 'There') => `${greeting} ${name}`;

console.log(greet('Hi'));

// OOP

// Constructor Function

function Person(firstName, lastName, dob) {

// Set object properties

this.firstName = firstName;

this.lastName = lastName;

this.dob = new Date(dob); // Set to actual date object using Date constructor

// this.getBirthYear = function(){

// return this.dob.getFullYear();

// }

// this.getFullName = function() {

// return `${this.firstName} ${this.lastName}`

// }

}

// Get Birth Year

Person.prototype.getBirthYear = function () {

return this.dob.getFullYear();

}

// Get Full Name

Person.prototype.getFullName = function() {

return `${this.firstName} ${this.lastName}`

}

// Instantiate an object from the class

const person1 = new Person('John', 'Doe', '7-8-80');

const person2 = new Person('Steve', 'Smith', '8-2-90');

console.log(person2);

// console.log(person1.getBirthYear());

// console.log(person1.getFullName());

// Built in constructors

const name = new String('Kevin');

console.log(typeof name); // Shows 'Object'

const num = new Number(5);

console.log(typeof num); // Shows 'Object'

// ES6 CLASSES

class Person {

constructor(firstName, lastName, dob) {

this.firstName = firstName;

this.lastName = lastName;

this.dob = new Date(dob);

}

// Get Birth Year

getBirthYear() {

return this.dob.getFullYear();

}

// Get Full Name

getFullName() {

return `${this.firstName} ${this.lastName}`

}

}

const person1 = new Person('John', 'Doe', '7-8-80');

console.log(person1.getBirthYear());

// ELEMENT SELECTORS

// Single Element Selectors

console.log(document.getElementById('my-form'));

console.log(document.querySelector('.container'));

// Multiple Element Selectors

console.log(document.querySelectorAll('.item'));

console.log(document.getElementsByTagName('li'));

console.log(document.getElementsByClassName('item'));

const items = document.querySelectorAll('.item');

items.forEach((item) => console.log(item));

// MANIPULATING THE DOM

const ul = document.querySelector('.items');

// ul.remove();

// ul.lastElementChild.remove();

ul.firstElementChild.textContent = 'Hello';

ul.children[1].innerText = 'Brad';

ul.lastElementChild.innerHTML = '<h1>Hello</h1>';

const btn = document.querySelector('.btn');

// btn.style.background = 'red';

// EVENTS

// Mouse Event

btn.addEventListener('click', e => {

e.preventDefault();

console.log(e.target.className);

document.getElementById('my-form').style.background = '#ccc';

document.querySelector('body').classList.add('bg-dark');

ul.lastElementChild.innerHTML = '<h1>Changed</h1>';

});

// Keyboard Event

const nameInput = document.querySelector('#name');

nameInput.addEventListener('input', e => {

document.querySelector('.container').append(nameInput.value);

});