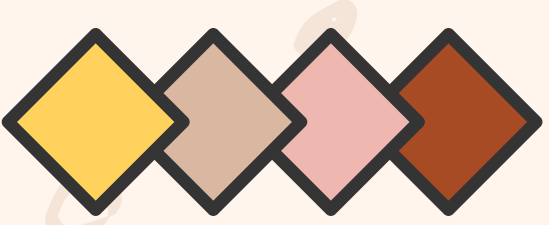


JavaScript



“

Topic

Introduction to Javascript

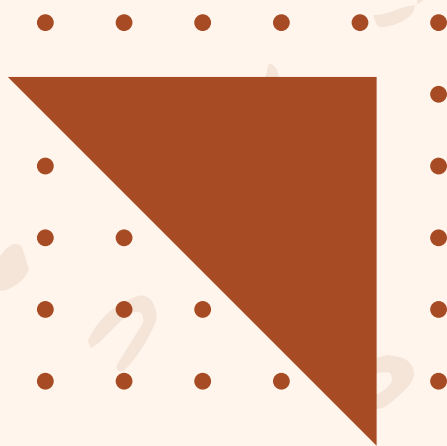
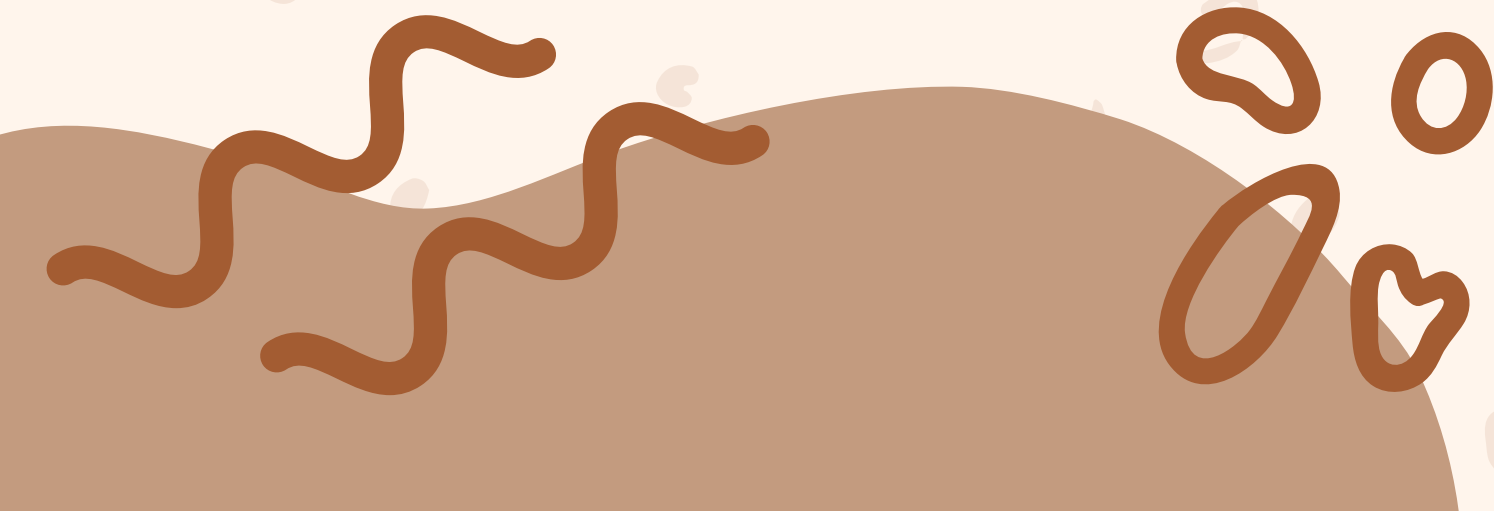
Arrays and looping

Variables and Data Types

Functions & Objects

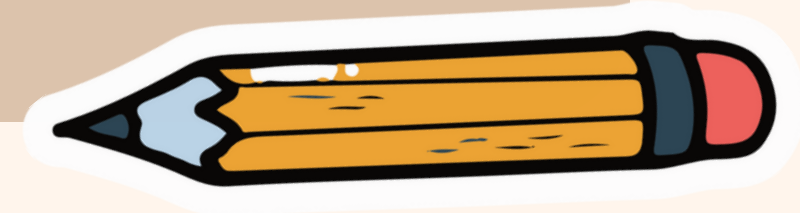
Conditional Statement

”



Introduction

- What is JavaScript? A versatile scripting language primarily used for web development.
- Not related to Java.



Variables & Constants

variable: container to store a value

```
let variable = value;
```

Constants: Store data that doesn't change.

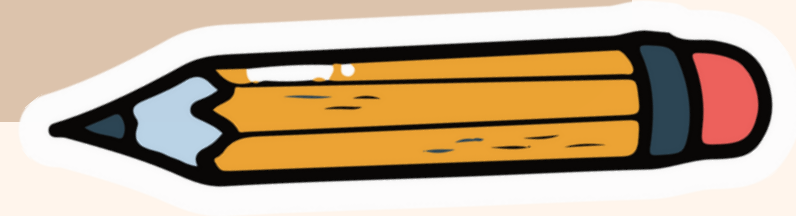
```
const maxScore = 100;
```

Conditional Statement



Using if, else to make decisions.

```
if (age >= 18) {  
    console.log("You can vote.");  
} else {  
    console.log("You're too young to vote.");  
}
```



Array & Looping

Array

Arrays: Lists that store multiple items.

```
let colors = ["red", "green", "blue"];
```

Looping

Used to repeat actions.

```
for (let i = 0; i < colors.length; i++) {  
  console.log(colors[i]);  
}
```

Functions & Objects

Functions

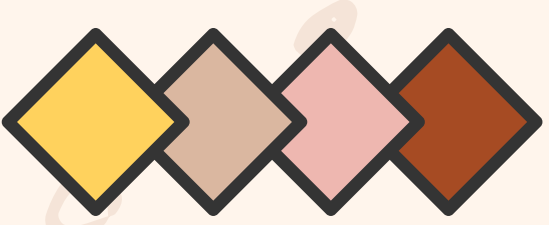
Functions: Reusable pieces of code.

```
function greet(name) {  
  return "Hello, " + name + "!";  
}
```

Objects

Objects: Group related data together.

```
let book = {  
  title: "To Kill a Mockingbird",  
  author: "Harper Lee"  
};
```



Advanced Js Topic



Introduction to ES6

Export and Imports

Arrow function

Rest and Spread Operator

Callbacks

Promise, Async await



Introduction to Modern JavaScript (ES6)

Arrow Function: Shorter ways to write functions.

```
const add = (a, b) => a + b;
```

Template literals: for easier string creation

```
let value = 'world'  
console.log(`hello ${value}`);
```

Callbacks Functions

- What is a Callback? A function passed into another function as an argument.
- Common in asynchronous operations.

```
function doSomething(callback) {  
  // Some operation  
  callback("Done!");  
}  
  
doSomething(message => {  
  console.log(message);  
});
```

Export and Imports

- Export: Make code available for use in another file.
- Import: Bring in code from another file.

```
// math.js
export const add = (a, b) => a + b;

// app.js
import { add } from './math';
console.log(add(1, 2)); // Output: 3
```

Rest and Spread Operator

- Rest: Collects multiple elements into an array.
- Spread: Spreads an array into multiple elements.

```
// Rest
function sum(...nums) {
  return nums.reduce((a, b) => a + b);
}
```

```
// Spread
let arr = [1, 2, 3];
let arr2 = [...arr, 4, 5];
```

Promise, Async, and Await

- Promise: Handles asynchronous operations.
- Async/Await: A cleaner way to handle Promises.

```
// Using Promises
fetchData().then(data => {
  console.log(data);
});

// Using Async/Await
async function getData() {
  let data = await fetchData();
  console.log(data);
}
```



**THANK YOU
SO MUCH!**