# Assignment 2

## Favour Wanjui

**Data Types and Variables:**

1. The different data types used in JavaScript variables in the provided code are:

- String (e.g., 'bed', "John")

- Number (including integers, floats, and Infinity)

- Boolean (true, false)

- Array (e.g., myRoom, marks)

- Object (e.g., countryInfo, info)

- Undefined

- Null

2. Difference between var, let, and const:

- var: Function-scoped, can be redeclared and updated

- let: Block-scoped, can be updated but not redeclared in the same scope

- const: Block-scoped, cannot be updated or redeclared

3. JavaScript allows assigning different data types to the same variable because it's a dynamically-typed language.

4. Variables declared but not initialized in JavaScript are automatically assigned the value `undefined`. Example from the code:

let student;

console.log(typeof student); // Output: undefined

5. Variable names in JavaScript are important for code readability and maintenance. They should be descriptive and follow conventions (e.g., camelCase for variables). Good examples from the code include myKiswahiliMarks and bankBalance.

**Numeric Data Types:**

1. Numeric data types in the code include:

- Integers (e.g., myKiswahiliMarks = 67)

- Floats (e.g., bankBalance = 23.78)

- Infinity (e.g., yearsInHeaven = Infinity)

2. Integers are whole numbers, floats have decimal points, and Infinity represents a value greater than any other number.

3. JavaScript automatically converts between numeric types as needed for arithmetic operations.

**String Data Type:**

1. Strings in JavaScript are represented by text enclosed in single ('') or double ("") quotes.

2. There's no difference between single and double quotes for strings in JavaScript. The choice is often a matter of style or convenience (e.g., when the string itself contains quotes).

3. Characters are treated as strings because JavaScript doesn't have a separate character type. Any single character is simply a string of length 1.

**Boolean and Undefined Data Types:**

1. Boolean variables in JavaScript represent true or false values, used for logical operations and control flow.

2. undefined in JavaScript means a variable has been declared but not assigned a value. Example:

let student;

console.log(typeof student); // Output: undefined

3. Boolean variables are crucial in conditional statements (if/else) and loops, determining the flow of program execution.

**Null Data Type:**

1. null in JavaScript represents the intentional absence of any object value.

2. Unlike undefined, which is automatically assigned, null is explicitly assigned by the programmer.

3. Example from the code:

let age = null;

console.log(age); // Output: null

**Object Data Type:**

1. Objects in JavaScript are collections of key-value pairs.

2. The `countryInfo` object structure:

let countryInfo = { citizenShip: 'Kenyan', idNumber: 44455567 };

It contains information about a person's citizenship and ID number.

3. Objects can be nested by assigning an object as a value to another object's property.

**Array Data Type:**

1. Arrays in JavaScript are ordered lists of values, used to store multiple items in a single variable.

2. Examples of arrays with different data types:

let myRoom = ['bed', 'chair', 'gas cooker', 'table', 'tv'];

let marks = [34, 56, 67, 78];

3. "Array of arrays" refers to an array containing other arrays as elements, allowing for multi-dimensional data structures.

**Variable Naming Conventions:**

1. JavaScript variable naming conventions include:

- Using camelCase for variable names

- Starting with a letter, underscore, or dollar sign

- Using descriptive names

2. Meaningful variable names improve code readability and self-documentation.

3. The code generally follows good naming conventions (e.g., `myKiswahiliMarks`, `bankBalance`), though some names could be more descriptive (e.g., `sname` could be `surname`).

**Constants in JavaScript:**

1. The const keyword is used to declare variables whose values should not be reassigned.

2. Reassigning a value to a constant variable results in an error because it violates the immutability principle of constants.

3. Example from the code:

const phoneNumber = 254789567364;

// phoneNumber = 345564734893; // This would cause an error if uncommented