Objects and their Internal Representation in JavaScript

Basics of Objects:

Objects are the heart of JavaScript. They serve as the building blocks for modern JavaScript applications.

Unlike primitive data types (such as numbers, strings, booleans, null, undefined, and symbols), objects can hold a combination of primitive and reference data(objects, arrays, functions, etc.) types.

An object is a reference data type, meaning that variables assigned to an object store a reference or pointer to the actual object in memory.

Key-Value Pairs:

An object is an unordered collection of related data represented as key-value pairs.

Keys (properties) can be variables or functions, while values can be any data type.

For example, an object representing a student might have properties like name, age, address, and methods like updateAddress.

Internal Representation:

JavaScript engines represent objects using various data structures.

A common representation is the hash table, where keys are hashed for efficient property access.

The hash table allows quick lookup of properties, making object access efficient