**OBJECTS AND IT’S INTERNAL REPRESENTATION IN JAVASCRIPT**

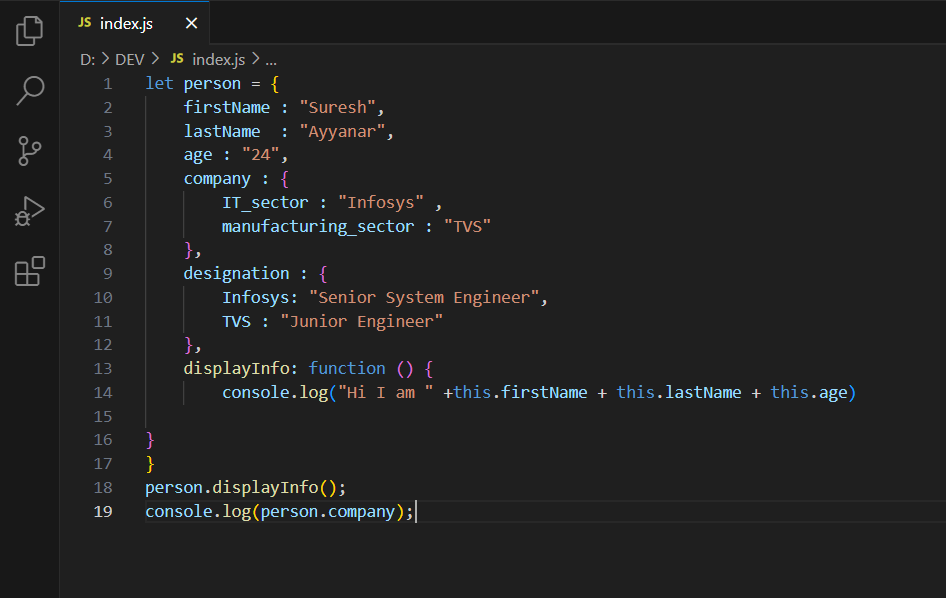
Objects are important data types in JavaScript and forms the building blocks for modern JavaScript. Objects are different from other primitive datatypes (Number, String, Boolean, null, undefined and symbol). Primitive data types contain one value but objects can hold many values in the form of **Key : value pair**. These keys are strings (or symbols) and values can be of any data type, including other objects. Objects are used to represent real-world entities, data structures, and more complex data types.

One of easiest way to create a JavaScript object is object literal, simply define keys and values inside curly braces as shown below:

let person = {firstName: “Suresh”, lastName: “Ayyanar”, age: “24” };

**Internal Representation of Objects:**

Example code snippet:



Internally, JavaScript engines use various data structures to represent objects efficiently. One common approach is using a hash table or a similar structure to store the object’s properties and their corresponding values. This allows for fast access and manipulation of properties.

## **Accessing Object Methods**

we access an object method with the following syntax:

objectName.methodName()

**Key-value pair for the above code snippet:**

|  |  |
| --- | --- |
| **KEY** | **VALUE** |
| firstName | Suresh |
| lastName | Ayyanar |
| age | 24 |
| company | Another object for company |
| designation | Another object for designation |

Objects in JavaScript are dynamic in nature, when we add a property the engine adds a key-value pair in the table. In the same way when we remove or modify a property the engine does the changes accordingly.