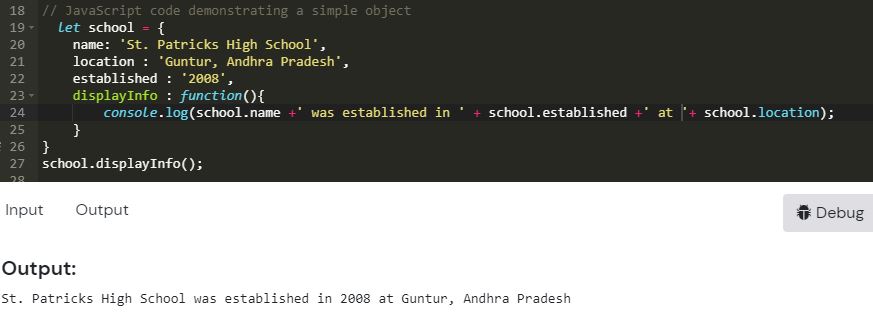
**OBJECTS IN JAVASCRIPT** **:-** These are the most important data type and form the building blocks for modern JavaScript. These objects are quite different from JavaScript’s primitive data types (Number, String, Boolean, null, undefined, and symbol) in the sense that these primitive data types all store a single value each (depending on their types). Objects can hold many values in form of Key: value pair. These keys can be variables or functions and are called properties and methods, respectively, in the context of an object. An object can be created with figure brackets {…} with an optional list of properties. A property is a “key: value” pair, where a key is a string (also called a “property name”), and the value can be anything.

**INTERNAL REPRESENTATION OF OBJECTS IN JS WITH EXAMPLE**



In this example **“name”, “location”,**and **“established”** are all **“keys”** and **“St. Patricks High School”, “Guntur, Andhra Pradesh” and “2008”**are values of these keys respectively. Each of these keys is referred to as **properties** of the object. An object in JavaScript may also have a function as a member, in which case it will be known as a **method** of that object. Here “**displayInfo”** is a method of the school object that is being used to work with the object’s data, stored in its properties.

**KEY POINTS ABOUT OBJECTS :-**

* Objects are more complex and each object may contain any combination of these primitive data-types as well as reference data-types.
* An object is a reference data type. Variables that are assigned a reference value are given a reference or a pointer to that value. That reference or pointer points to the location in memory where the object is stored. The variables don’t actually store the value.
* Loosely speaking, objects in JavaScript may be defined as an unordered collection of related data, of primitive or reference types, in the form of “key: value” pairs.