Accessing Object Properties

There are two ways to **access an object's properties:**

- Dot notation: .

  In our example, if we want to retrieve the property value of weapon we can do so with object dot notation by typing the variable name of the object followed by a dot (.) and the property or method name.

- Bracket notation: [ ]

  We can retrieve the same data with object bracket notation. Similar to how you might index and access a string, the syntax for bracket notation is two square brackets ([ ]) encasing the property name.

```js
// Initializing a gimli object
var gimli = {
    name: "Gimli",
    race: "dwarf",
    weapon: "axe",
    greet: function() {
        return `Hi, my name is ${this.name}!`;
    },
};
// Retrieving the value of the weapon property using the dot notation
console.log(gimli.weapon); // prints "axe"
// Retrieving the value of the weapon property using the bracket notation
console.log(gimli["weapon"]); // prints "axe"
```

Accessing Object Properties

Both dot notation and bracket notation are used regularly. Dot notation is faster and more legible but it has more limitations. Bracket notation allows access to property names stored in a variable

and must be used if an object's property contains any sort of special character.

In order to retrieve an object method you simply call it in the same way you would call a regular function: by attaching it to the object variable.

```javascript
// Initializing a gimli object
var gimli = {
    name: "Gimli",
    race: "dwarf",
    weapon: "axe",
    greet: function() {
        return `Hi, my name is ${this.name}!`;
    },
};

// Calling an object method using the dot notation
console.log(gimli.greet());
// prints "Hi, my name is Gimli!"
```