

# DIFFERENCE BETWEEN

`Object.defineProperty()`

&

`object.property = value`

## **Dot Notation** (`object.property = value`):

Using this method, the property is:

- **Writable:** You can change its value.
- **Enumerable:** It will show up in loops like `for...in` and methods like `Object.keys()`.
- **Configurable:** You can delete the property or change its attribute

```
let obj = {};  
obj.property = "value";
```

## **Object Method** (`Object.defineProperty()`):

Using `Object.defineProperty()`, you can:

- **Control writability:** Make the property read-only by setting `writable` to `false`.
- **Control enumerability:** Prevent the property from showing up in loops and methods by setting `enumerable` to `false`.
- **Control configurability:** Prevent the property from being deleted or its attributes from being changed by setting `configurable` to `false`.
- **Define getters and setters:** Create properties with custom getter and setter functions.

```
let obj = {};  
Object.defineProperty(obj, 'property', {  
  value: 'value',  
  writable: false,  
  enumerable: false,  
  configurable: false  
});
```