

In JavaScript classes, the **get** and **set** keywords are used to define **getter** and **setter** methods, respectively. These methods allow for controlled access to an object's properties.

Getter (get)

- A getter method is defined using the **get** keyword followed by a method name. It allows you to access a property value in a controlled manner.
- When you access the property, the getter method is invoked, and you can define custom logic to return a value.
- It is typically used to compute or format a property value dynamically.

Example:

```
get width() {  
  return this._width;}  

```

In this example, accessing **rectangle.width** will call the **get width()** method, returning the value of **_width**.

Setter (set)

- A setter method is defined using the **set** keyword followed by a method name. It allows you to set a property value in a controlled manner.
- When you assign a value to the property, the setter method is invoked, allowing you to define custom logic for validation or transformation before assigning the value.
- It is typically used to enforce rules or constraints on property values.

Example:

```
set width(newWidth) {  
  if (newWidth > 0) {  
    this._width = newWidth;  
  } else {  
    console.error("Enter a Valid width");  
  }  
}
```

```
}
```

In this example, assigning a value to **rectangle.width** will call the **setWidth(newWidth)** method, which checks if the value is valid before setting **_width**.

Summary

- **Getters** allow you to retrieve property values with custom logic.
- **Setters** allow you to assign property values with validation or transformation.