getter e setter

As funções do tipo getter e setter servem para interceptar o acesso as propriedades de determinado um objeto

getter_setter_1.js — javascriptmasterclass

JS getter_setter_1.js

```
1   const rectangle = {
2       x: 10,
3       y: 2,
4       get area() {
5           return this.x * this.y;
6       }
7   };
8   console.log(rectangle.area);
9
```

TERMINAL 1: bash

```
rodrigobranas:javascriptmasterclass $ node g
etter_setter/getter_setter_1.js
20
rodrigobranas:javascriptmasterclass $
```

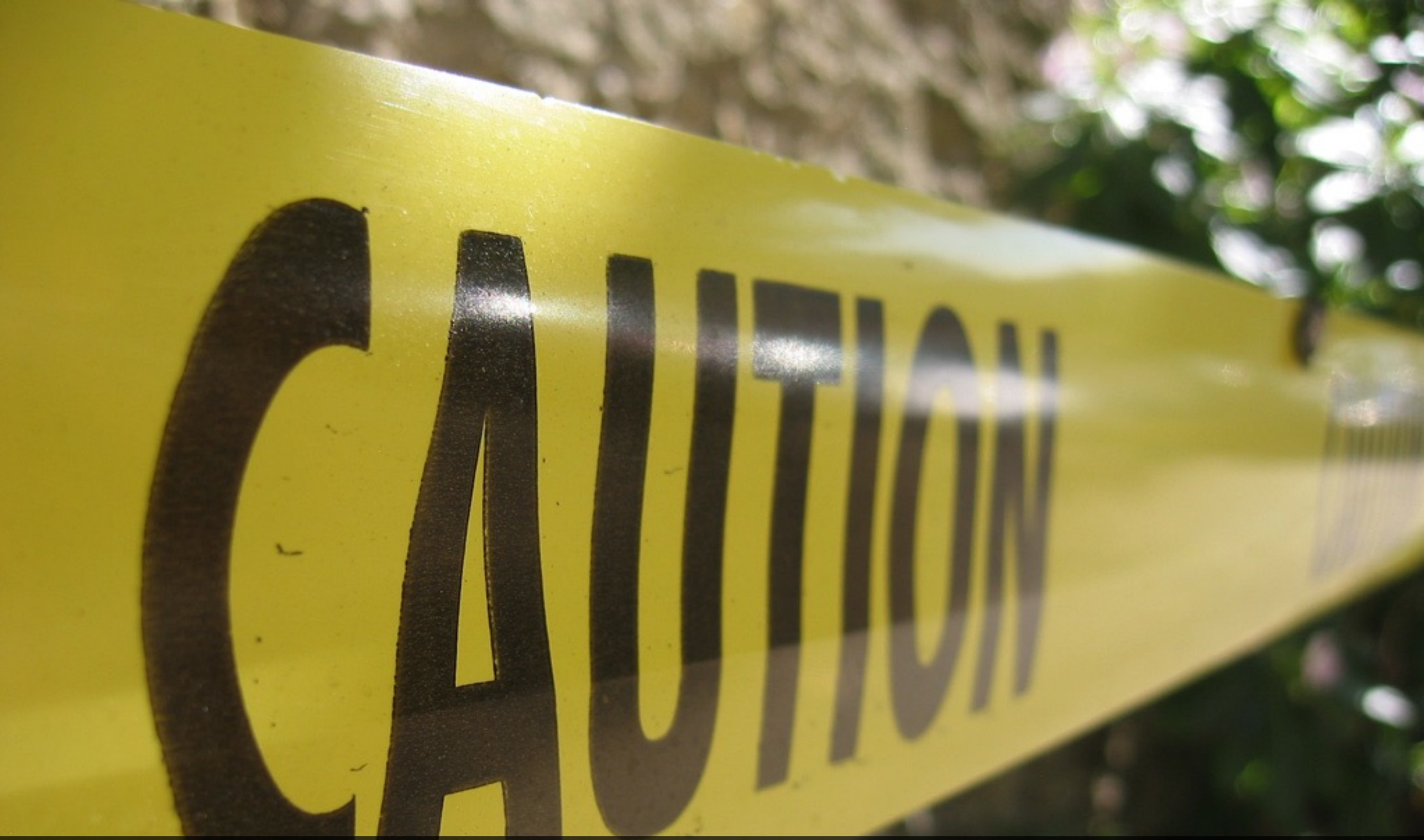
getter_setter_2.js — javascriptmasterclass

getter_setter_2.js

```javascript
const rectangle = {
    set x(x) {
        this.x = x;
    },
    set y(y) {
        this.y = y;
    },
    get area() {
        return this.x * this.y;
    }
};
rectangle.x = 10;
rectangle.y = 2;
console.log(rectangle.area);
```

TERMINAL 1: bash

```
RangeError: Maximum call stack size exceeded
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
    at Object.set x [as x] (/Users/rodrigobranas/development/workspace/javascriptmasterclass/getter_setter/getter_setter_2.js:3:16)
rodrigobranas:javascriptmasterclass $
```



Utilize chaves diferentes para a função setter e a propriedade do objeto

getter_setter_3.js — javascriptmasterclass

JS getter_setter_3.js

```
1  const rectangle = {
2      set x(x) {
3          this._x = x;
4      },
5      set y(y) {
6          this._y = y;
7      },
8      get area() {
9          return this._x * this._y;
10     }
11 };
12 rectangle.x = 10;
13 rectangle.y = 2;
14 console.log(rectangle.area);
15
```

TERMINAL 1: bash

```
rodrigobranas:javascriptmasterclass $ node g
etter_setter/getter_setter_3.js
20
rodrigobranas:javascriptmasterclass $
```

getter_setter_4.js — javascriptmasterclass

JS getter_setter_4.js

```
const rectangle = {
    set x(x) {
        if (x > 0) {
            this._x = x;
        } else {
            console.log("Invalid value for x");
        }
    },
    set y(y) {
        if (y > 0) {
            this._y = y;
        } else {
            console.log("Invalid value for y");
        }
    },
    get area() {
        return this._x * this._y;
    }
};
rectangle.x = -10;
rectangle.y = -2;
console.log(rectangle.area);
```

TERMINAL

1: bash

```
rodrigobranas:javascriptmasterclass $ node g
etter_setter/getter_setter_4.js
Invalid value for x
Invalid value for y
NaN
rodrigobranas:javascriptmasterclass $
```

getter_setter_5.js — javascriptmasterclass

JS getter_setter_5.js

```
const rectangle = {
    set x(x) {
        if (x > 0) {
            this._x = x;
        } else {
            console.log("Invalid value for x");
        }
    },
    set y(y) {
        if (y > 0) {
            this._y = y;
        } else {
            console.log("Invalid value for y");
        }
    },
    get area() {
        return this._x * this._y;
    }
};
rectangle.x = 10;
rectangle.y = 2;
console.log(rectangle.area);
```

TERMINAL

1: bash

```
rodrigobranas:javascriptmasterclass $ node g
etter_setter/getter_setter_5.js
20
rodrigobranas:javascriptmasterclass $
```

Por meio da operação defineProperty da Object API, também é possível definir funções do tipo getter e setter

getter_setter_6.js — javascriptmasterclass

getter_setter_6.js

```
1   const rectangle = {};
2   Object.defineProperty(rectangle, "area", {
3       get() {
4           return this.x * this.y;
5       }
6   });
7   rectangle.x = 10;
8   rectangle.y = 2;
9   console.log(rectangle.area);
10
```

TERMINAL 1: bash

```
rodrigobranas:javascriptmasterclass $ node g
etter_setter/getter_setter_6.js
20
rodrigobranas:javascriptmasterclass $
```

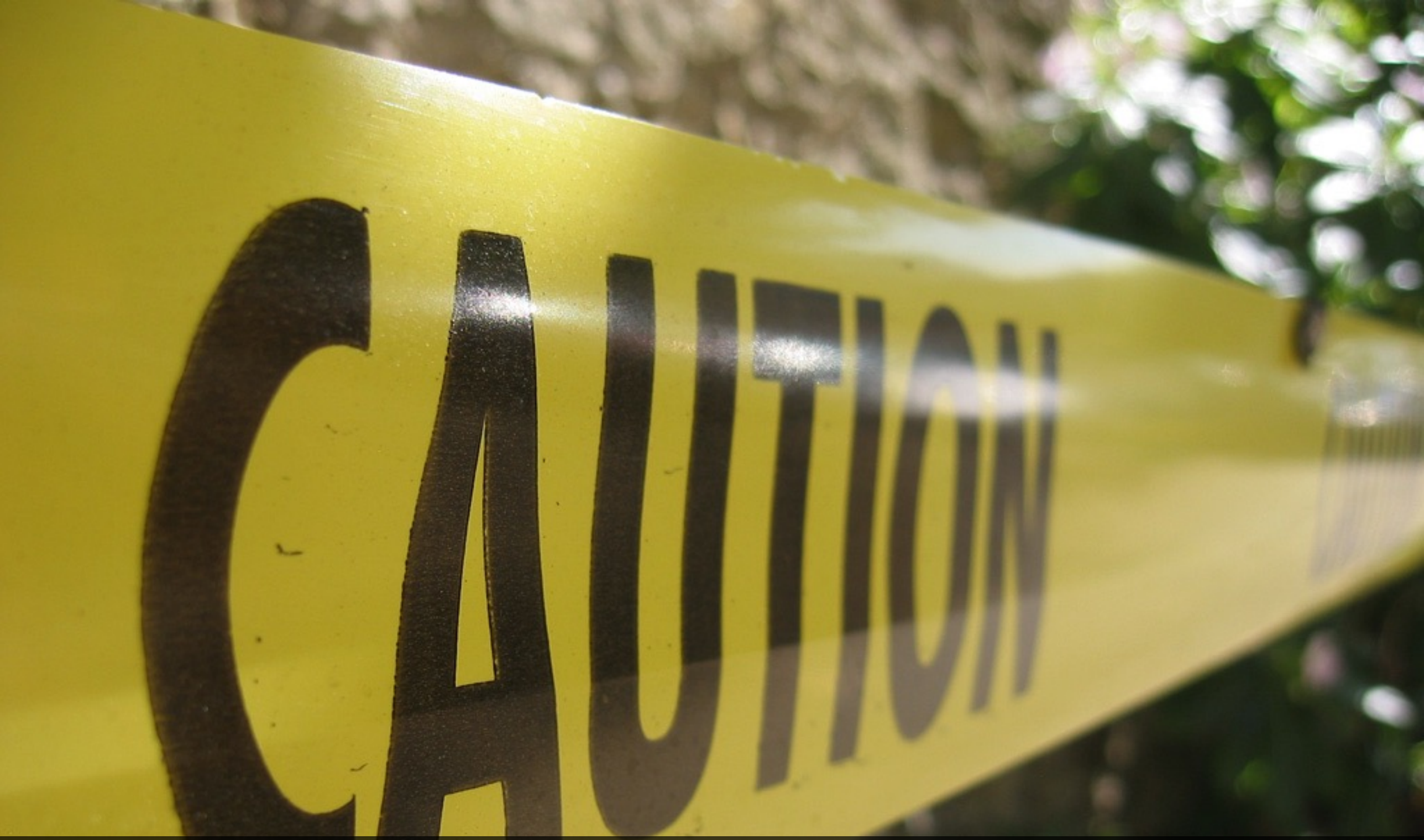
getter_setter_7.js — javascriptmasterclass

getter_setter_7.js

```javascript
const rectangle = {};
Object.defineProperty(rectangle, "area", {
    get() {
        return this.x * this.y;
    }
});
Object.defineProperty(rectangle, "x", {
    set(x) {
        this.x = x;
    }
});
Object.defineProperty(rectangle, "y", {
    set(y) {
        this.y = y;
    }
});
rectangle.x = 10;
rectangle.y = 2;
console.log(rectangle.area);
```

TERMINAL   1: bash

```
RangeError: Maximum call stack size exceeded
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:8:8)
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:9:16)
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:9:16)
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:9:16)
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:9:16)
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:9:16)
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:9:16)
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:9:16)
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:9:16)
    at Object.set (/Users/rodrigobranas/deve
lopment/workspace/javascriptmasterclass/gett
er_setter/getter_setter_7.js:9:16)
rodrigobranas:javascriptmasterclass $
```



Utilize chaves diferentes para a função setter e a propriedade do objeto

getter_setter_8.js — javascriptmasterclass

JS getter_setter_8.js

```
const rectangle = {};
Object.defineProperty(rectangle, "area", {
    get() {
        return this._x * this._y;
    }
});
Object.defineProperty(rectangle, "x", {
    set(x) {
        this._x = x;
    }
});
Object.defineProperty(rectangle, "y", {
    set(y) {
        this._y = y;
    }
});
rectangle.x = 10;
rectangle.y = 2;
console.log(rectangle.area);
```

TERMINAL 1: bash

```
rodrigobranas:javascriptmasterclass $ node g
etter_setter/getter_setter_8.js
20
rodrigobranas:javascriptmasterclass $
```

getter_setter_9.js — javascriptmasterclass

JS getter_setter_9.js

```
        return this._x * this._y;
    }
});
Object.defineProperty(rectangle, "x", {
    set(x) {
        if (x > 0) {
            this._x = x;
        } else {
            console.log("Invalid value for x");
        }
    }
});
Object.defineProperty(rectangle, "y", {
    set(y) {
        if (y > 0) {
            this._y = y;
        } else {
            console.log("Invalid value for y");
        }
    }
});
rectangle.x = -10;
rectangle.y = -2;
console.log(rectangle.area);
```

TERMINAL 1: bash

```
rodrigobranas:javascriptmasterclass $ node g
etter_setter/getter_setter_9.js
Invalid value for x
Invalid value for y
NaN
rodrigobranas:javascriptmasterclass $
```

getter_setter_10.js — javascriptmasterclass

JS getter_setter_10.js

```
        return this._x * this._y;
    }
});
Object.defineProperty(rectangle, "x", {
    set(x) {
        if (x > 0) {
            this._x = x;
        } else {
            console.log("Invalid value for x");
        }
    }
});
Object.defineProperty(rectangle, "y", {
    set(y) {
        if (y > 0) {
            this._y = y;
        } else {
            console.log("Invalid value for y");
        }
    }
});
rectangle.x = 10;
rectangle.y = 2;
console.log(rectangle.area);
```

TERMINAL 1: bash

```
rodrigobranas:javascriptmasterclass $ node g
etter_setter/getter_setter_10.js
20
rodrigobranas:javascriptmasterclass $
```