<!DOCTYPE html>

<html lang="en" xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta charset="utf-8" />

<title></title>

<script>

class Rectangle {

constructor(a,b)

{

this.getA = function () {

return a;

}

this.setA = function (value) {

if (value < 0)

throw new Error("v<0");

a = value;

}

this.setA(a); //Контроль значення яке передали у конструкторі

this.getB = function () {

return b;

}

this.setB = function (value) {

if (value < 0)

throw new Error("v<0");

b = value;

}

this.S = function () {

return a \* b;

}

this.setB(b); //Контроль значення яке передали у конструкторі

}

S() {

return this.getA() \* this.getB();

}

}

//---------------- Варіант №2

function Rectangle(a,b)

{

this.getA = function () {

return a;

}

this.setA = function (value) {

if (value < 0)

throw new Error("v<0");

a = value;

}

this.setA(a); //Контроль значення яке передали у конструкторі

this.getB = function () {

return b;

}

this.setB = function (value) {

if (value < 0)

throw new Error("v<0");

b = value;

}

this.S = function () {

return a \* b;

}

this.setB(b); //Контроль значення яке передали у конструкторі

}

//--------------

var r1 = new Rectangle(-2, 4);

alert(r1.S());

</script>

</head>

<body>

</body>

</html>

<!DOCTYPE html>

<html lang="en" xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta charset="utf-8" />

<title></title>

<script>

class Rectangle {

constructor(a,b) //Що треба знати

{

//----- Опис властивостей

Object.defineProperty(this, "A", {

get: function () {

return a;

},

set: function (value) {

if (value < 0)

throw new Error("v<0");

a = value;

}

});

Object.defineProperty(this, "B", {

get: function () {

return b;

},

set: function (value) {

if (value < 0)

throw new Error("v<0");

b = value;

}

});

//---- Контроль значень, які передали у конструктор

this.A = a;

this.B = b;

}

//--------

S() {

return this.getA() \* this.getB();

}

}

//--------------

var r1 = new Rectangle(2, 4, 9);

var tt = { a: 12, b: 8 }

Object.defineProperty(tt, "cc", { value: 99, enumerable: true });

for (var i in tt) {

alert(tt[i]);

}

</script>

</head>

<body>

</body>

</html>

<!DOCTYPE html>

<html lang="en" xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta charset="utf-8" />

<title></title>

<script>

class Client {

constructor(name,accountNumber,money,message) //Що треба знати

{

//----- Опис властивостей загальнодоступних

this.Message = message;

//----- Опис властивостей тільки для читання

Object.defineProperty(this, "Name", { value: name });

Object.defineProperty(this, "AccountNumber", { value: accountNumber });

//----- Опис властивостей (для контрольованого запису і зчитування)

Object.defineProperty(this, "Money", {

get: function () {

return money;

},

set: function (value) {

if (money < 0) throw new Error("Некоректна кількість грошей");

money = value;

}

});

//---- Контроль значень, які передали у конструктор

this.Money = money;

}

//-------- інші методи класу

toString()

{

return `${this.Name} - ${this.AccountNumber} - ${this.Money}`;

}

}

//--------------

var client = new Client("Ivan", "45567667", 9000, "Надійний клієнт");

document.write(client);

</script>

</head>

<body>

</body>

</html>