

|  |  |
| --- | --- |
| Без Object.defineProperty | З Object.defineProperty |
| class Rectangle{  constructor(sideA=0,sideB=0){  //--- Описуємо методи для зчитування і встановлення значень закритих полів  this.getA=function(){  return sideA  }  this.setA=function(value){  if(value<0)  throw new Error('the value is incorrect')  sideA=value  }  this.getB=function(){  return sideB  }  this.setB=function(value){  if(value<0)  throw new Error('the value is incorrect')  sideB=value  }    //----- Перевіряємо дані, які були передані конструктору  this.setA(sideA)  this.setB(sideB)  }  getArea(){  return this.getA()\*this.getB()  }  getPerimeter(){  return 2\*(this.getA()+this.getB())  }  scale(multiplier){  this.setA(this.getA()\*multiplier)  this.setB(this.getB()\*multiplier)  }  } | class Rectangle{  constructor(sideA=0,sideB=0){  //--- Описуємо методи для зчитування і встановлення значень закритих полів  Object.defineProperty(this,'A',{  get:function(){  return sideA  },  set:function(value){  if(value<0)  throw new Error('the value is incorrect')  sideA=value  }  })  Object.defineProperty(this,'B',{  get:function(){  return sideB  },  set:function(value){  if(value<0)  throw new Error('the value is incorrect')  sideB=value  }  })  //----- Перевіряємо дані, які були передані конструктору  this.A=sideA //Метод set буде викликано автоматично  this.B=sideB  }  getArea(){  return this.A\*this.B //Автоматично викликаються методи get  }  getPerimeter(){  return 2\*(this.A+this.B)  }  scale(multiplier){  this.A=this.A\*multiplier  this.B=this.B\*multiplier  }  } |
|  |  |

class UsingRectanle{

constructor(){

this.rect= new Rectangle() //Копозиція (поле також є об"єктом іншого класу)

}

setA(){

try{

const a=parseFloat(this.inpA.value)

this.rect.A=a

}

catch(e)

{

alert('Вводьте коректні значення для А: '+e.message)

}

}

setB(){

try{

const b=parseFloat(this.inpB.value)

this.rect.B=b

}

catch(e)

{

alert('Вводьте коректні значення для B: '+e.message)

}

}

perim(){

const p=this.rect.getPerimeter()

this.spanPerim.innerText=p

}

render(){

const table= document.createElement('table')

//---------

let tr= document.createElement('tr')

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

let td= document.createElement('td')

td.innerText='Side A'

tr.appendChild(td)

td= document.createElement('td')

this.inpA=document.createElement('input')

this.inpA.type='number'

td.appendChild(this.inpA)

let btn=document.createElement('button')

btn.innerText='set A'

btn.onclick=this.setA.bind(this)

td.appendChild(btn)

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

tr.appendChild(td)

table.appendChild(tr)

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

//---------

tr= document.createElement('tr')

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

td= document.createElement('td')

td.innerText='Side B'

tr.appendChild(td)

td= document.createElement('td')

this.inpB=document.createElement('input')

this.inpB.type='number'

td.appendChild(this.inpB)

btn=document.createElement('button')

btn.innerText='set B'

btn.onclick=this.setB.bind(this)

td.appendChild(btn)

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

tr.appendChild(td)

table.appendChild(tr)

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

tr= document.createElement('tr')

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

td= document.createElement('td')

btn=document.createElement('button')

btn.innerText='Perim'

btn.onclick=this.perim.bind(this)

td.appendChild(btn)

tr.appendChild(td)

td= document.createElement('td')

this.spanPerim=document.createElement('span')

td.appendChild(this.spanPerim)

tr.appendChild(td)

table.appendChild(tr)

return table

}

}

======================================================================

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Document</title>

<script src="./Rectangle.js"></script>

<script src="./UsingRectanle.js"></script>

<script>

window.onload=function(){

const usingRect= new UsingRectanle()

document.getElementById('container').appendChild(usingRect.render())

}

</script>

</head>

<body>

<div id="container">

</div>

</body>

</html>