

```

package practica3junit;

import static org.junit.jupiter.api.Assertions.*;

import org.junit.jupiter.api.Test;

class PujaTest {

    @Test
    public void PujaTest1() {
        int valorinicial = 150;
        Puja prueba = new Puja(0, "Jose", "Bici", 150);
        prueba.modificarprecioinicial(valorinicial);
        assertEquals(prueba.pujainicial, valorinicial);
        assertEquals(prueba.pujamaxima, valorinicial);
    }

    @Test
    public void PujaTest2() {
        int valorinicial = 200;
        Puja prueba = new Puja(0, "Jose", "Bici", 300);
        prueba.modificarprecioinicial(valorinicial);
        assertEquals(prueba.pujainicial, valorinicial);
        assertEquals(prueba.pujamaxima, valorinicial);
    }

    //-----
    ---

    @Test
    public void PujaTest3() {
        String nombrepujador = "Josepe";
        int nuevapuja = 400;
        Puja prueba = new Puja(0, "Jose", "Bici", 300);
        prueba.realizarpuja(nuevapuja, nombrepujador);
        assertEquals(prueba.pujamaxima, nuevapuja);
        assertEquals(prueba.pujadormaximo, nombrepujador);
    }
    /*El nombre del pujador es diferente y la puja es mayor
    por lo que el programa se ejecuta correctamente*/

    @Test
    public void PujaTest4() {
        String nombrepujador = "Josepe";
        int nuevapuja = 250;

```

```

        Puja prueba = new Puja(0, "Jose", "Bici", 300);
        prueba.realizarpuja(nuevapuja, nombrepujador);
        assertEquals(prueba.pujamaxima, nuevapuja);
        assertEquals(prueba.pujadormaximo, nombrepujador);
    }
    /*El nombre del pujador es diferente pero la puja es menor
    por lo que el programa nos da error*/

    @Test
    public void PujaTest5() {
        String nombrepujador = "Josepe";
        int nuevapuja = 400;
        Puja prueba = new Puja(0, "Jose", "Bici", 300);
        prueba.pujadormaximo = "Josepe";
        prueba.realizarpuja(nuevapuja, nombrepujador);
        assertEquals(prueba.pujamaxima, nuevapuja);
        assertEquals(prueba.pujadormaximo, nombrepujador);
    }
    /*El nombre del pujador es diferente pero la puja es menor
    por lo que el programa nos da error*/

    //-----

    @Test
    public void PujaTest6() {
        String estadoproductoesperado = "Cerrado";
        Puja prueba = new Puja(0, "Jose", "Bici", 150);
        prueba.pujamaxima = 300;
        prueba.pujadormaximo = "Daniel";
        prueba.adjudicarproducto();
        assertEquals(estadoproduktoesperado, prueba.estadoprodukt);
    }
    /*Como el nombre del pujador m ximo no est  vac o y la puja m xima
    es
        mayor a la pujainicial, el produkt es adjudicado*/

    @Test
    public void PujaTest7() {
        String estadoproductoesperado = "Cerrado";
        Puja prueba = new Puja(0, "Jose", "Bici", 150);
        prueba.pujamaxima = 300;
        prueba.pujadormaximo = "";
        prueba.adjudicarprodukt();
        assertEquals(estadoproduktoesperado, prueba.estadoprodukt);
    }

```

```

    }

    /*Como el nombre del pujador m ximo est  vac o, la puja no es
    v lida
    y el producto no es adjudicado a nadie*/

    @Test
    public void PujaTest8() {
        String estadoproductoesperado = "Cerrado";
        Puja prueba = new Puja(0, "Jose", "Bici", 150);
        prueba.pujamaxima = 50;
        prueba.pujadormaximo = "Daniel";
        prueba.adjudicarproducto();
        assertEquals(estadoproduktoesperado,prueba.estadoprodukt);
    }

    /*Como la puja m xima es diferente al precio inicial, la puja es
    v lida
    (a pesar de que sea menor al precio inicial)*/

    @Test
    public void PujaTest9() {
        String estadoproductoesperado = "Cerrado";
        Puja prueba = new Puja(0, "Jose", "Bici", 150);
        prueba.pujamaxima = 150;
        prueba.pujadormaximo = "Daniel";
        prueba.adjudicarproducto();
        assertEquals(estadoproduktoesperado,prueba.estadoprodukt);
    }

    /*La puja no ser a v lida ya que la puja m xima es igual al precio
    inicial*/

    @Test
    public void PujaTest10() {
        String estadoproductoesperado = "Cerrado";
        Puja prueba = new Puja(0, "Jose", "Bici", 150);
        prueba.pujamaxima = 300;
        prueba.pujadormaximo = "Daniel";
        prueba.adjudicarproducto();
        assertEquals(estadoproduktoesperado,prueba.estadoprodukt);
    }

    /*Como el nombre del pujador m ximo no est  vac o y la puja m xima
    es
    mayor a la pujainicial, el producto es adjudicado*/

```

```
//-----  
-  
  
@Test  
public void PujaTest11() {  
    Puja prueba = new Puja(0, "Jose", "Bici", 150);  
    String nuevonombreproducto = "Cereales";  
    prueba.producto = "Cereales";  
    prueba.modificarproducto(nuevonombreproducto);  
    assertEquals(nuevonombreproducto,prueba.producto);  
}  
/*Como el nombre del producto es igual al anterior, el programa  
nos advierte de que tiene el mismo nombre*/  
  
@Test  
public void PujaTest12() {  
    Puja prueba = new Puja(0, "Jose", "Bici", 150);  
    String nuevonombreproducto = "Cereales";  
    prueba.producto = "Motocicleta";  
    prueba.modificarproducto(nuevonombreproducto);  
    assertEquals(nuevonombreproducto,prueba.producto);  
}  
/*Se sobrescribe correctamente el nombre*/  
  
//-----  
-  
  
@Test  
public void PujaTest13() {  
    Puja prueba = new Puja(0, "Jose", "Bici", 150);  
    String nuevonombrevendedor = "Antonio";  
    prueba.nombrevendedor = "Juan Antonio";  
    prueba.modificarvendedor(nuevonombrevendedor);  
    assertEquals(nuevonombrevendedor,prueba.nombrevendedor);  
}  
/*Se sobrescribe correctamente el nombre*/  
  
@Test  
public void PujaTest14() {  
    Puja prueba = new Puja(0, "Jose", "Bici", 150);  
    String nuevonombrevendedor = "Antonio";  
    prueba.nombrevendedor = "Antonio";  
    prueba.modificarvendedor(nuevonombrevendedor);  
    assertEquals(nuevonombrevendedor,prueba.nombrevendedor);  
}
```

```
/*Como el nombre del vendedor es igual al anterior, el programa  
nos advierte de que tiene el mismo nombre*/
```

```
}
```