

```
package test.java;

import static org.junit.Assert.*;

import org.junit.Before;
import org.junit.Rule;
import org.junit.Test;
import org.junit.rules.Timeout;
import org.junit.runner.RunWith;
import org.junit.runners.JUnit4;

import main.java.Article;
import main.java.Shop;

@RunWith(JUnit4.class)

public class SearchDicho2Test {

    private Article articleValid;
    private Article articleValid2;
    private Article articleValid3;
    private Shop shop;

    @Rule
    public Timeout globalTimeout = Timeout.seconds(5);

    @Before
    public void setUp() throws Exception {
        articleValid = new Article("Article1", 50, 1);
        articleValid2 = new Article("Article2", 51, 2);
        articleValid3 = new Article("Article3", 52, 3);
        shop = new Shop(10);
        shop.insertArticle(articleValid);
        shop.insertArticle(articleValid2);
        shop.insertArticle(articleValid3);
    }

    @Test
    public void searchDicho2ElementInTableTest() {
        Article articleToFind = new Article("Article3", 52, 3);
        assertTrue(shop.searchDicho2(articleToFind));
        assertTrue(articleValid3.isEqual(articleToFind));
    }

    @Test
    public void searchDicho2ElementNotInTableTest() {
        Article articleToFindFalse = new Article("ToFindFalse", 51.5, 5);
        assertFalse(shop.searchDicho2(articleToFindFalse));
        for(int i = 0; i < shop.getNbArticle(); i++) {
            assertFalse(shop.getStock()[0].isEqual(articleToFindFalse));
        }
    }

    @Test
    public void searchDicho2ElementWrongName() {
        Article articleToFindWrongName = new Article("WrongName", 52, 3);
        assertFalse(shop.searchDicho2(articleToFindWrongName));
        assertFalse(articleValid3.isEqual(articleToFindWrongName));
    }

    @Test
    public void searchDicho2ElementWrongNumber() {
        Article articleToFindWrongNumber = new Article("Article3", 52, 5);
        assertFalse(shop.searchDicho2(articleToFindWrongNumber));
        assertFalse(articleValid3.isEqual(articleToFindWrongNumber));
    }

    @Test
```

```
public void searchDicho2ElementWrongPrice() {
    Article articleToFindWrongPrice = new Article("Article3", 52.01, 3);
    assertFalse(shop.searchDicho2(articleToFindWrongPrice));
    assertFalse(articleValid3.isEqual(articleToFindWrongPrice));
}

@Test
public void searchDichoElementMini() {
    Article articleFree = new Article("Article3", 0.00, 3);
    assertFalse(shop.searchDicho2(articleFree)) ;
    shop.insertArticle(articleFree);
    assertTrue(shop.searchDicho2(articleFree)) ;
}

@Test
public void searchDichoElementMax() {
    Article articleExpensive = new Article("Article3", 10000.01, 3);
    assertFalse(shop.searchDicho2(articleExpensive)) ;
    shop.insertArticle(articleExpensive);
    assertTrue(shop.searchDicho2(articleExpensive)) ;
}
}
```