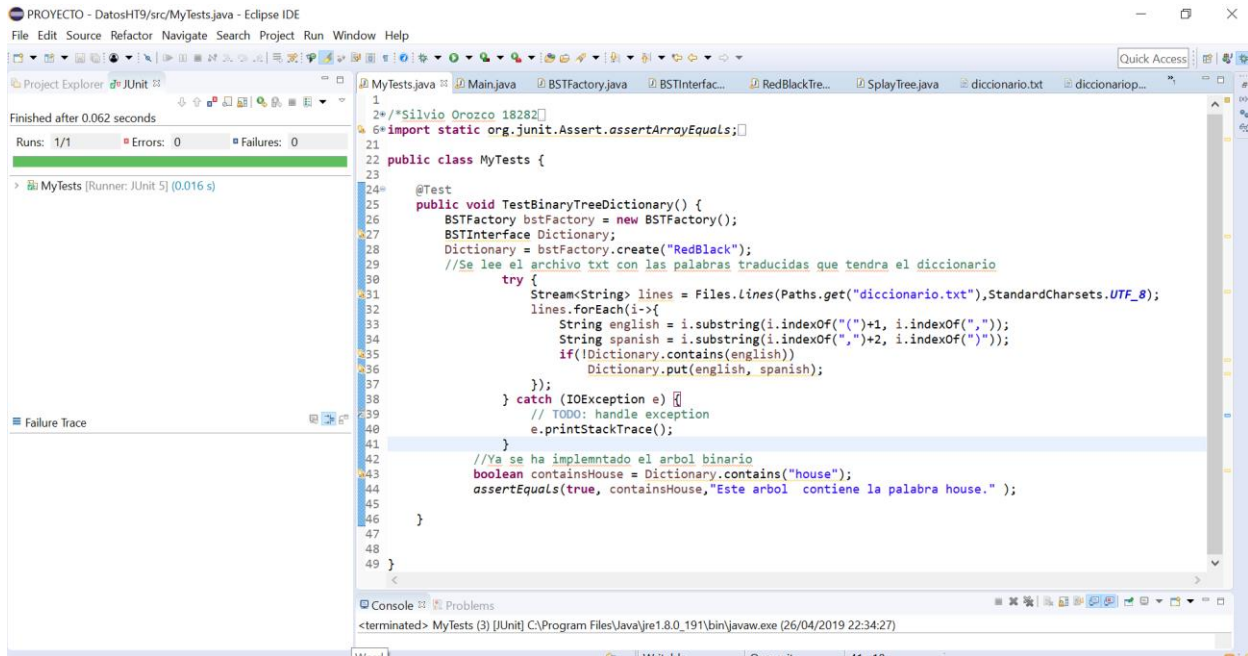


## Pruebas JUNIT

### RedBlackTree

Busca si contiene una palabra insertada en el diccionario.

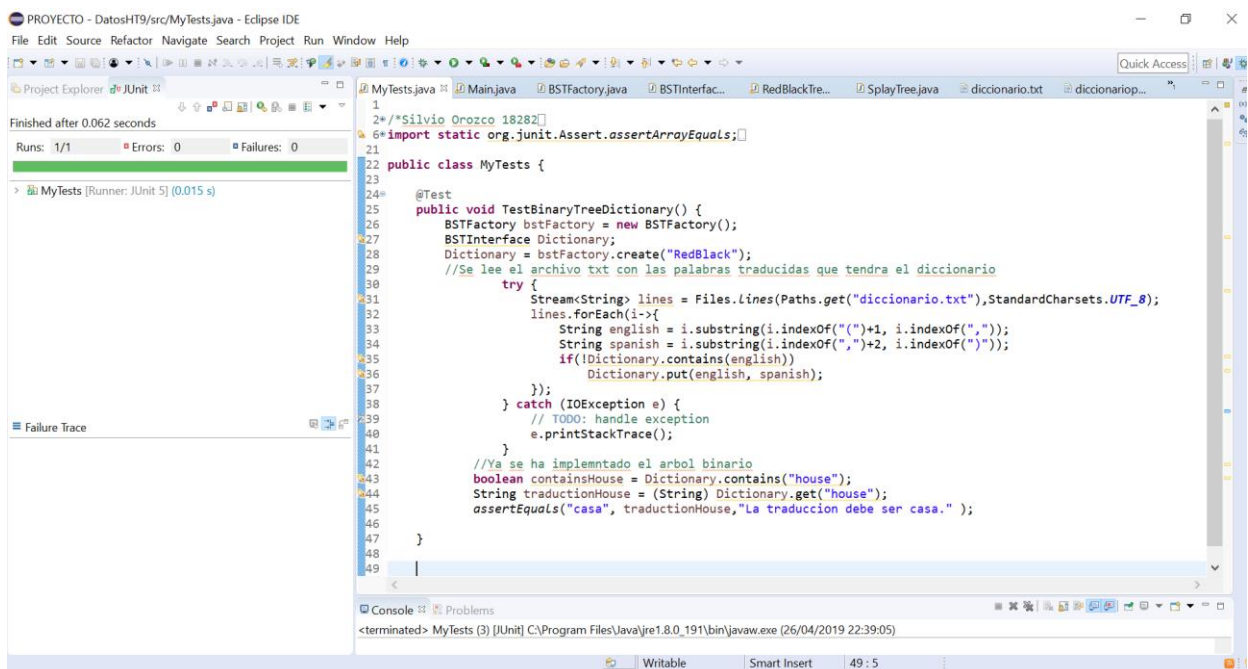
Además, el árbol inserta correctamente todos los datos contenidos en el diccionario.txt



```
1 2/*Silvio Orozco 18282
6*import static org.junit.Assert.assertEquals;
21
22 public class MyTests {
23
24     @Test
25     public void TestBinaryTreeDictionary() {
26         BSTFactory bstFactory = new BSTFactory();
27         BSTInterface Dictionary;
28         Dictionary = bstFactory.create("RedBlack");
29         //Se lee el archivo txt con las palabras traducidas que tendra el diccionario
30         try {
31             Stream<String> lines = Files.lines(Paths.get("diccionario.txt"), StandardCharsets.UTF_8);
32             lines.forEach(i->{
33                 String english = i.substring(i.indexOf("(")+1, i.indexOf(","));
34                 String spanish = i.substring(i.indexOf(",")+2, i.indexOf(")"));
35                 if(!Dictionary.contains(english))
36                     Dictionary.put(english, spanish);
37             });
38         } catch (IOException e) {
39             // TODO: handle exception
40             e.printStackTrace();
41         }
42         //Ya se ha implementado el arbol binario
43         boolean containsHouse = Dictionary.contains("house");
44         assertEquals(true, containsHouse, "Este arbol contiene la palabra house.");
45     }
46 }
47
48 }
49 }
```

Console: <terminated> MyTests (3) [JUnit] C:\Program Files\Java\jre1.8.0\_191\bin\javaw.exe (26/04/2019 22:34:27)

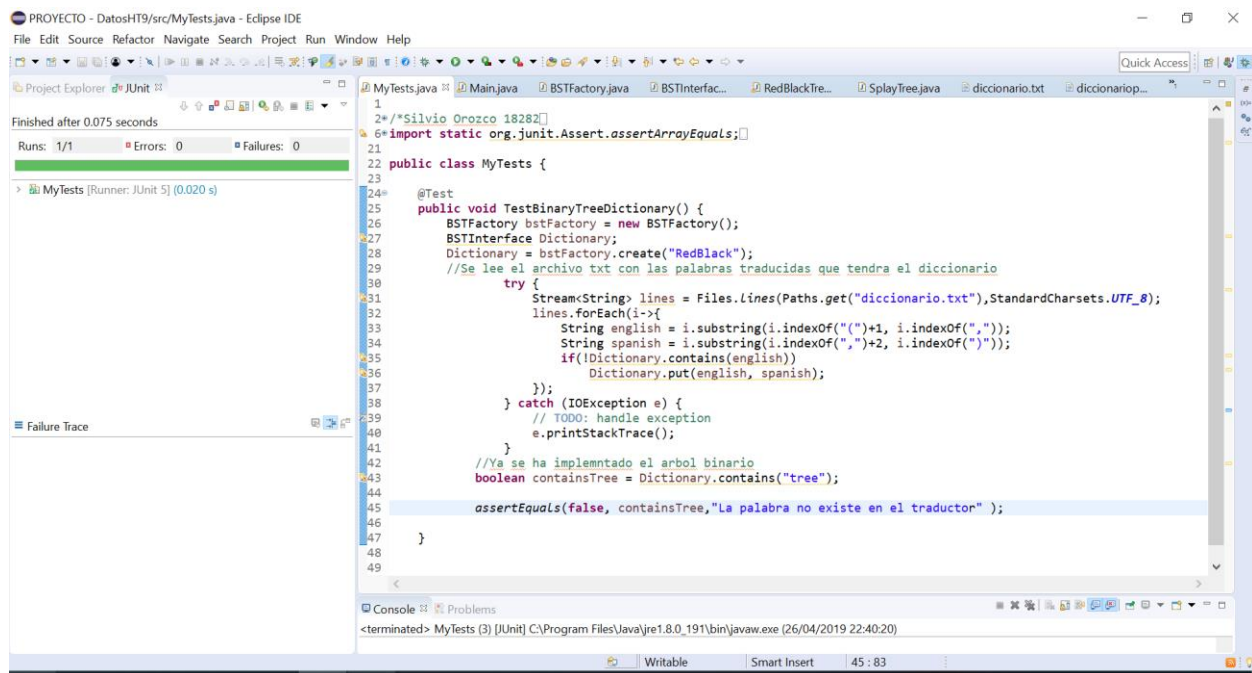
Cuando encuentra lo que obtiene del diccionario.



```
1 2/*Silvio Orozco 18282
6*import static org.junit.Assert.assertEquals;
21
22 public class MyTests {
23
24     @Test
25     public void TestBinaryTreeDictionary() {
26         BSTFactory bstFactory = new BSTFactory();
27         BSTInterface Dictionary;
28         Dictionary = bstFactory.create("RedBlack");
29         //Se lee el archivo txt con las palabras traducidas que tendra el diccionario
30         try {
31             Stream<String> lines = Files.lines(Paths.get("diccionario.txt"), StandardCharsets.UTF_8);
32             lines.forEach(i->{
33                 String english = i.substring(i.indexOf("(")+1, i.indexOf(","));
34                 String spanish = i.substring(i.indexOf(",")+2, i.indexOf(")"));
35                 if(!Dictionary.contains(english))
36                     Dictionary.put(english, spanish);
37             });
38         } catch (IOException e) {
39             // TODO: handle exception
40             e.printStackTrace();
41         }
42         //Ya se ha implementado el arbol binario
43         boolean containsHouse = Dictionary.contains("house");
44         String traductionHouse = (String) Dictionary.get("house");
45         assertEquals("casa", traductionHouse, "La traduccion debe ser casa.");
46     }
47
48 }
49 }
```

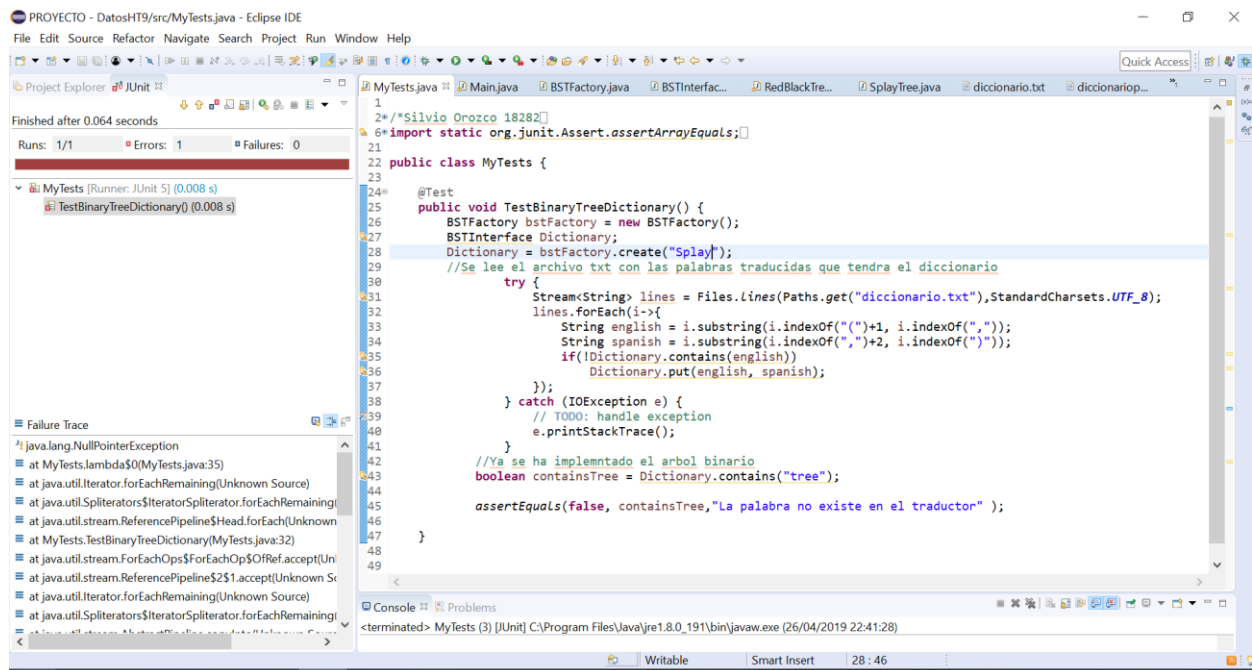
Console: <terminated> MyTests (3) [JUnit] C:\Program Files\Java\jre1.8.0\_191\bin\javaw.exe (26/04/2019 22:39:05)

Cuando la palabra no existe en el diccionario.



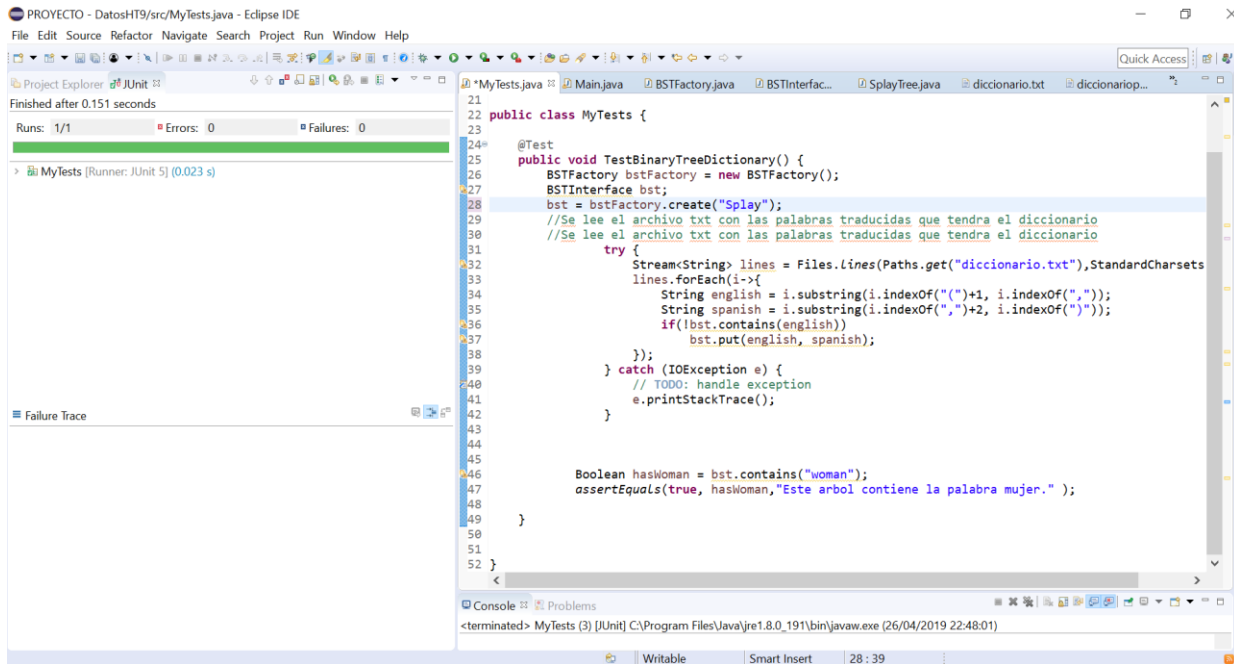
SplayTree

Existía un error con el nombre que se implementaba con Splay tree



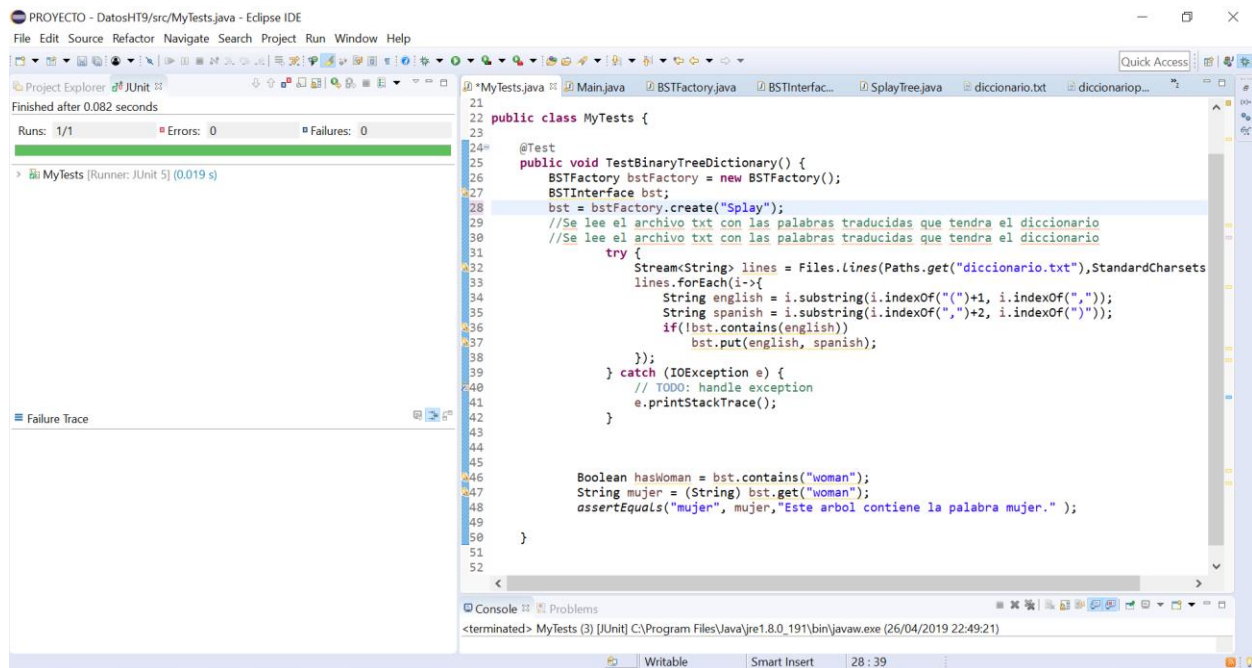
Busca si contiene una palabra insertada en el diccionario.

Además, el árbol inserta correctamente todos los datos contenidos en el diccionario.txt



```
21
22 public class MyTests {
23
24     @Test
25     public void TestBinaryTreeDictionary() {
26         BSTFactory bstFactory = new BSTFactory();
27         BSTInterface bst;
28         bst = bstFactory.create("Splay");
29         //Se lee el archivo txt con las palabras traducidas que tendra el diccionario
30         //Se lee el archivo txt con las palabras traducidas que tendra el diccionario
31         try {
32             Stream<String> lines = Files.lines(Paths.get("diccionario.txt"), StandardCharsets
33             lines.forEach(i->{
34                 String english = i.substring(i.indexOf("(")+1, i.indexOf(","));
35                 String spanish = i.substring(i.indexOf(",")+2, i.indexOf(")"));
36                 if(!bst.contains(english))
37                     bst.put(english, spanish);
38             });
39         } catch (IOException e) {
40             // TODO: handle exception
41             e.printStackTrace();
42         }
43
44
45         Boolean hasWoman = bst.contains("woman");
46         assertEquals(true, hasWoman, "Este arbol contiene la palabra mujer. ");
47     }
48 }
49
50
51
52 }
```

Cuando encuentra lo que obtiene del diccionario.



Cuando la palabra no existe en el diccionario.

