

Flyweight Unit Test

```
1 package edu.cpp.cs5800.flyweight;
2
3 import org.junit.jupiter.api.Test;
4
5 import static org.junit.jupiter.api.Assertions.assertEquals;
6 import static org.junit.jupiter.api.Assertions.assertTrue;
7
8 public class DocumentTest {
9     @Test
10    public void testLoad() {
11        Document document = new Document();
12        assertTrue(document.load("test_load.txt"));
13        String expected = "H(ARIAL,RED,12)\n";
14        assertEquals(expected, document.getContent());
15    }
16
17    @Test
18    public void testAddCharacterOneParam() {
19        Document document = new Document();
20        Character ch = new Character('Z', new CharacterFormat(CharacterFormat.Font.ARIAL,
21            CharacterFormat.Color.BLACK, 14));
22        assertTrue(document.addCharacter(ch));
23
24        String expected = "Z(ARIAL,BLACK,14)\n";
25        assertEquals(expected, document.getContent());
26    }
27
28    @Test
29    public void testAddCharacterFourParams() {
30        Document document = new Document();
31        assertTrue(document.addCharacter('X',
32            CharacterFormat.Font.VERDANA,
33            CharacterFormat.Color.BLACK,
34            14));
35
36        String expected = "X(VERDANA,BLACK,14)\n";
37        assertEquals(expected, document.getContent());
38    }
39
40    @Test
41    public void testSave() {
42        String filename = "test_save.txt";
43        Character ch = new Character('Z', new CharacterFormat(CharacterFormat.Font.ARIAL,
44            CharacterFormat.Color.BLACK, 14));
45        Document document = new Document();
46        document.addCharacter(ch);
47        assertTrue(document.save(filename));
48
49        Document document2 = new Document();
50        document2.load(filename);
51        assertEquals("Z(ARIAL,BLACK,14)\n", document2.getContent());
52    }
53}
```

Flyweight Unit Test

```
54     @Test
55     public void testGetContent() {
56         Character ch1 = new Character('Z', new CharacterFormat(CharacterFormat.Font.ARIAL,
57             CharacterFormat.Color.BLACK, 14));
58         Character ch2 = new Character('A', new CharacterFormat(CharacterFormat.Font.
VERDANA,
59             CharacterFormat.Color.BLUE, 16));
60
61         Document document = new Document();
62         document.addCharacter(ch1);
63         document.addCharacter(ch2);
64
65         String expected = "Z(ARIAL,BLACK,14)\nA(VERDANA,BLUE,16)\n";
66         assertEquals(expected, document.getContent());
67     }
68 }
69
```

Flyweight Unit Test

```
1 package edu.cpp.cs5800.flyweight;
2
3 import org.junit.jupiter.api.Test;
4
5 import static org.junit.jupiter.api.Assertions.assertEquals;
6
7 public class CharacterTest {
8     @Test
9     public void testGetColor() {
10         CharacterFormat format = new CharacterFormat(CharacterFormat.Font.CALIBRI,
11             CharacterFormat.Color.BLACK, 12);
12         assertEquals(CharacterFormat.Color.BLACK, format.getColor());
13     }
14
15     @Test
16     public void testGetFont() {
17         CharacterFormat format = new CharacterFormat(CharacterFormat.Font.CALIBRI,
18             CharacterFormat.Color.BLACK, 12);
19         assertEquals(CharacterFormat.Font.CALIBRI, format.getFont());
20     }
21
22     @Test
23     public void testGetFontSize() {
24         CharacterFormat format = new CharacterFormat(CharacterFormat.Font.CALIBRI,
25             CharacterFormat.Color.BLACK, 12);
26         assertEquals(12, format.getFontSize());
27     }
28 }
29
```

Flyweight Unit Test

```
1 package edu.cpp.cs5800.flyweight;
2
3 import org.junit.jupiter.api.Test;
4
5 import static org.junit.jupiter.api.Assertions.assertEquals;
6
7 public class CharacterFormatTest {
8     @Test
9     public void testGetSymbol() {
10         Character ch = new Character('A', null);
11         assertEquals(ch.getSymbol(), 'A');
12     }
13
14     @Test
15     public void testToString() {
16         Character ch = new Character('A', null);
17         assertEquals("A(null)", ch.toString());
18     }
19 }
20
```

Flyweight Unit Test

```
1 package edu.cpp.cs5800.flyweight;
2
3 import org.junit.jupiter.api.Test;
4
5 import static org.junit.jupiter.api.Assertions.assertEquals;
6
7 public class CharacterFactoryTest {
8     @Test
9     public void testGetCharacter() {
10         String expected = "B(ARIAL,RED,12)";
11         Character actual = CharacterFactory.getCharacter('B', CharacterFormat.Font.ARIAL,
12             CharacterFormat.Color.RED, 12);
13         assertEquals(expected, actual.toString());
14     }
15 }
16
```