square / **javapoet**

Watch 234   Star 3,626   Fork 501

<> Code   ! Issues 38   Pull requests 1   Projects 0   Insights ▾

Improve test on TypeName #544

Merged   **swankjesse** merged 1 commit into `square:master` from `danglotb:ampl-TypeNameTest` on Feb 1

💬 Conversation 7   Commits 1   📄 Files changed 1

Changes from **all commits** ▾   1 file ▾   +2 −0 ■■■□□□□   Unified | Split

2 ■■■□□□ src/test/java/com/squareup/javapoet/TypeNameTest.java   ⊠ Show comments   View   ⌄

```
@@ -25,6 +25,7 @@
25   25
26   26       import static com.google.common.truth.Truth.assertThat;
27   27       import static org.junit.Assert.assertEquals;
     28   +   import static org.junit.Assert.assertFalse;
28   29       import static org.junit.Assert.assertNotEquals;
29   30
30   31       public class TypeNameTest {
```

```
@@ -178,5 +179,6 @@ private void assertEqualsHashCodeAndToString(TypeName a, TypeName b) {
178   179        assertEquals(a.toString(), b.toString());
179   180        assertThat(a.equals(b)).isTrue();
180   181        assertThat(a.hashCode()).isEqualTo(b.hashCode());
      182   +     assertFalse(a.equals(null));
```

ronshapiro on Feb 1   Contributor

`assertThat(a).isNotNull();`

tbroyer on Feb 1   Collaborator

He, no: this specifically about testing the `.equals(null)` contract 😊

sormuras on Feb 1   Contributor

See https://github.com/square/javapoet/blob/master/src/main/java/com/squareup/javapoet/TypeName.java#L197

ronshapiro on Feb 1 • edited   Contributor

Oh fair. What about `assertThat(a).isNotEqualTo(null)` ?

sormuras on Feb 1 • edited   Contributor

Would be more inline with

```
assertThat(a.equals(b)).isTrue();
assertThat(a.hashCode()).isEqualTo(b.hashCode());
```

I think `assertThat(a).isEqualTo(a);` is missing, too. Testing the `if (this == o) return true;` contract.

JakeWharton on Feb 1   Owner

True, but that's just an optimization. If `a.equals(b)` works then `b.equals(a)` must also work, thus we then know `a.equals(a)` will work.

sormuras on Feb 1   Contributor

True. True. Stays true.

```
181   183        }
182   184      }
```

```
}
```