

```
1 package net.wiredclub.translation;
2
3 import org.apache.commons.cli.ParseException;
4 import org.junit.jupiter.api.Test;
5
6 import java.io.IOException;
7 import java.util.HashSet;
8 import java.util.Set;
9
10 import static net.wiredclub.translation.TranslationStatusCode.STATUS_INVALID_ARGUMENT;
11 import static net.wiredclub.translation.TranslationToolTest.TEST_TRANSLATIONS_DIRECTORY;
12 import static org.junit.jupiter.api.Assertions.assertEquals;
13 import static org.junit.jupiter.api.Assertions.assertNotNull;
14 import static org.junit.jupiter.api.Assertions.assertThrows;
15 import static org.mockito.ArgumentMatchers.any;
16 import static org.mockito.Mockito.doThrow;
17 import static org.mockito.Mockito.eq;
18 import static org.mockito.Mockito.mock;
19 import static org.mockito.Mockito.times;
20 import static org.mockito.Mockito.verify;
21 import static org.mockito.Mockito.when;
22
23 public class CommandLineHelperTest {
24
25     @Test
26     void testGetTranslationConfigWithoutArguments() throws TranslationException, IOException {
27         String[] noArgs = {};
28
29         Set<String> targetLanguages = new HashSet<>() {{
30             add("de");
31             add("en");
32             add("fr");
33             add("nl");
34         }};
35
36         FileHelper fileHelperMock = mock(FileHelper.class);
37         when(fileHelperMock.discoverLanguageDirectories(any())).thenReturn(targetLanguages);
```

```
38
39     CommandLineHelper commandLineHelper = new CommandLineHelper(new DeepLHelper(), fileHelperMock);
40     TranslationConfig translationConfig = commandLineHelper.getTranslationConfig(noArgs);
41
42     assertNotNull(translationConfig);
43     assertEquals("en", translationConfig.sourceLanguage());
44     assertEquals(Set.of("de", "fr", "nl"), translationConfig.targetLanguages());
45     assertEquals("translations", translationConfig.translationsDirectory());
46     assertEquals(".", translationConfig.repositoryDirectory());
47 }
48
49 @Test
50 void testGetTranslationConfigWithValidArguments() throws TranslationException, IOException {
51     String[] args = {
52         "-s", "en",
53         "-t", "de",
54         "-p", TEST_TRANSLATIONS_DIRECTORY,
55         "-r", ".."
56     };
57
58     TranslationConfig translationConfig = new CommandLineHelper().getTranslationConfig(args);
59
60     assertNotNull(translationConfig);
61     assertEquals("en", translationConfig.sourceLanguage());
62     assertEquals(Set.of("de"), translationConfig.targetLanguages());
63     assertEquals(TEST_TRANSLATIONS_DIRECTORY, translationConfig.translationsDirectory());
64     assertEquals("..", translationConfig.repositoryDirectory());
65 }
66
67 @Test
68 void testGetTranslationConfigWithShortSyntax() throws TranslationException, IOException {
69     String[] args = {
70         "-s", "en"
71     };
72
73     Set<String> targetLanguages = new HashSet<>() {{
74         add("de");
```

```
75         add("en");
76         add("fr");
77         add("nl");
78     });
79
80     FileHelper fileHelperMock = mock(FileHelper.class);
81     when(fileHelperMock.discoverLanguageDirectories(any())).thenReturn(targetLanguages);
82
83     CommandLineHelper commandLineHelper = new CommandLineHelper(new DeepLHelper(), fileHelperMock);
84     TranslationConfig translationConfig = commandLineHelper.getTranslationConfig(args);
85
86     assertNotNull(translationConfig);
87     assertEquals("en", translationConfig.sourceLanguage());
88     assertEquals(Set.of("de", "fr", "nl"), translationConfig.targetLanguages());
89     assertEquals("translations", translationConfig.translationsDirectory());
90     assertEquals(".", translationConfig.repositoryDirectory());
91 }
92
93 @Test
94 void testGetTranslationConfigWithInvalidArguments() throws ParseException, IOException, TranslationException {
95     String[] invalidArgs = {"-x"};
96
97     CommandLineHelper commandLineHelperMock = mock(CommandLineHelper.class);
98     when(commandLineHelperMock.getTranslationConfig(invalidArgs)).thenCallRealMethod();
99     doThrow(ParseException.class).when(commandLineHelperMock).parseArguments(any(), eq(invalidArgs));
100
101     TranslationException thrownException =
102         assertThrows(TranslationException.class, () → commandLineHelperMock.getTranslationConfig(invalidArgs));
103
104     assertEquals("net.wiredclub.translation.TranslationException", thrownException.getClass().getName());
105     assertEquals(STATUS_INVALID_ARGUMENT, thrownException.statusCode());
106
107     verify(commandLineHelperMock, times(1)).displayHelp(any());
108 }
109
110 @Test
111 void testParseArgumentsWithInvalidSourceLanguage() {
```

```
112     String[] invalidArgs = {
113         "-s", "wrong language", // this is also the directory where translations live
114         "-t", "de",
115         "-p", TEST_TRANSLATIONS_DIRECTORY
116     };
117
118     CommandLineHelper commandLineHelper = new CommandLineHelper();
119
120     ParseException thrownException = assertThrows(ParseException.class,
121         () → commandLineHelper.parseArguments(commandLineHelper.defineOptions(), invalidArgs));
122
123     assertEquals("org.apache.commons.cli.ParseException", thrownException.getClass().getName());
124 }
125
126 @Test
127 void testParseArgumentsWithInvalidTargetLanguage() {
128     String[] invalidArgs = {
129         "-s", "en",
130         "-t", "wrong language", // this is also the directory where translations live
131         "-p", TEST_TRANSLATIONS_DIRECTORY
132     };
133
134     CommandLineHelper commandLineHelper = new CommandLineHelper();
135
136     ParseException thrownException = assertThrows(ParseException.class,
137         () → commandLineHelper.parseArguments(commandLineHelper.defineOptions(), invalidArgs));
138
139     assertEquals("org.apache.commons.cli.ParseException", thrownException.getClass().getName());
140 }
141 }
142
```