

```
1 package hw4;
2 /*
3   Authors: Elizabeth Hillman, Camellia Bazargan, Vy
4   Nguyen, Jagjit Singh
5   Date: 10/29/22
6   Assignment: GamesExpo - Domain Testing
7   Problem Statement: Determine which of the six
8   competition a child is eligible for, based on their
9   age and gender.
10  */
11 import java.util.*;
12
13 public class GamesExpo {
14     static int age;
15     static String gender;
16     public static void main(String[] args)
17     {
18         Scanner scan = new Scanner(System.in);
19         //gets user age
20         System.out.println("Enter your age");
21         age = scan.nextInt();
22
23         //gets user gender
24         System.out.println("Enter your gender: boy or
25         girl");
26         gender = scan.next();
27
28         //used to determine which event the child can
29         participate in
30         if(canParticipateInStorytelling(age,gender)) {
31             System.out.println("Can participate in
32             Storytelling");
33         } else if (canParticipateInDrawing(age,gender
34         )) {
35             System.out.println("Can participate in
36             Drawing");
37         } else if (canParticipateInQuiz(age,gender)) {
38             System.out.println("Can participate in
```

```
30 Quiz");
31         } else if (canParticipateInEssayWriting(age,
    gender)) {
32             System.out.println("Can participate in
    Essay Writing");
33         } else if (canParticipateInRhyming(age)) {
34             System.out.println("Can participate in
    Rhyming");
35         } else if (canParticipateInPoetry(age)) {
36             System.out.println("Can participate in
    Poetry");
37         } else if (cannotParticipate(age)){
38             System.out.println("Can not participate in
    any event");
39         }
40     }
41
42     //predicate function that returns true if the
    child is a boy and within age > 7 and < 10
43     public static boolean canParticipateInStorytelling
    (int age, String gender)
44     {
45         return (gender.equals("boy")) && (age >= 7 &&
    age <= 10);
46     }
47
48     //predicate function that returns true if the
    child is a girl and within age > 7 and < 10
49     public static boolean canParticipateInDrawing(int
    age, String gender)
50     {
51         return (gender.equals("girl")) && (age >= 7
    && age <= 10);
52     }
53
54     //predicate function that returns true if the
    child is a boy and within age > 11 and < 15
55     public static boolean canParticipateInQuiz(int age
```

```
55 , String gender)
56     {
57         return (gender.equals("boy")) && (age >= 11
58         && age <= 15);
59     }
60     //predicate function that returns true if the
        child is a girl and within age > 10 and < 15
61     public static boolean canParticipateInEssayWriting
        (int age, String gender)
62     {
63         return (gender.equals("girl")) && (age >= 10
64         && age <= 15);
65     }
66     //predicate function that returns true if the
        child is < age 6
67     public static boolean canParticipateInRhyming(int
        age)
68     {
69         return age <= 6 && age >= 0;
70     }
71
72     //predicate function that returns true if the
        child is > age 20
73     public static boolean canParticipateInPoetry(int
        age)
74     {
75         return age >= 20;
76     }
77
78     //predicate function that returns true if the
        child is age > 15 and < 20
79     public static boolean canNotParticipate(int age)
80     {
81         return age > 15 && age < 20;
82     }
83 }
```

```
1 package hw4;
2
3 import org.junit.jupiter.api.Test;
4
5 import static org.junit.jupiter.api.Assertions.*;
6
7 class GamesExpoTest {
8
9     //Test 1: If the user is a boy and has an age on
the inclusive boundary, then they should be able to
compete in story telling
10     @Test
11     void canParticipateInStorytelling() {
12         GamesExpo.age = 7;
13         GamesExpo.gender = "boy";
14         assertTrue(GamesExpo.
canParticipateInStorytelling(GamesExpo.age, GamesExpo.
gender));
15     }
16
17     //Test 2: To test the domain using boundary values
analysis for boys who can participate in storytelling
18     @Test
19     void boundaryVASToryTelling() {
20         GamesExpo.age = 6;
21         GamesExpo.gender = "boy";
22         assertFalse(GamesExpo.
canParticipateInStorytelling(GamesExpo.age, GamesExpo.
gender));
23
24         GamesExpo.age = 7;
25         GamesExpo.gender = "boy";
26         assertTrue(GamesExpo.
canParticipateInStorytelling(GamesExpo.age, GamesExpo.
gender));
27
28         GamesExpo.age = 11;
29         GamesExpo.gender = "boy";
```

```
30         assertFalse(GamesExpo.
    canParticipateInStorytelling(GamesExpo.age, GamesExpo.
    gender));
31
32         GamesExpo.age = 10;
33         GamesExpo.gender = "boy";
34         assertTrue(GamesExpo.
    canParticipateInStorytelling(GamesExpo.age, GamesExpo.
    gender));
35     }
36
37     //Test 3: To test the domain using EP technique
for boys who can participate in storytelling.
38     @Test
39     void equivalencePartitionStoryTelling() {
40         GamesExpo.age = 8;
41         GamesExpo.gender = "boy";
42         assertTrue(GamesExpo.
    canParticipateInStorytelling(GamesExpo.age, GamesExpo.
    gender));
43     }
44
45     //Test 4: If the user is a girls and has an age on
the inclusive boundary, then they should be able to
compete in drawing
46     @Test
47     void canParticipateInDrawing() {
48         GamesExpo.age = 7;
49         GamesExpo.gender = "girl";
50         assertTrue(GamesExpo.canParticipateInDrawing(
    GamesExpo.age, GamesExpo.gender));
51     }
52
53     //Test 5: To test the domain using boundary values
analysis for girls who can participate in drawing
54     @Test
55     void boundaryVADrawing() {
56         GamesExpo.age = 6;
```

```
57         GamesExpo.gender = "girl";
58         assertFalse(GamesExpo.canParticipateInDrawing(
    GamesExpo.age, GamesExpo.gender));
59
60         GamesExpo.age = 7;
61         GamesExpo.gender = "girl";
62         assertTrue(GamesExpo.canParticipateInDrawing(
    GamesExpo.age, GamesExpo.gender));
63
64         GamesExpo.age = 11;
65         GamesExpo.gender = "girl";
66         assertFalse(GamesExpo.canParticipateInDrawing(
    GamesExpo.age, GamesExpo.gender));
67
68         GamesExpo.age = 10;
69         GamesExpo.gender = "girl";
70         assertTrue(GamesExpo.canParticipateInDrawing(
    GamesExpo.age, GamesExpo.gender));
71     }
72
73     //Test 6: To test the domain using EP technique
for boys who can participate in drawing.
74     @Test
75     void equivalencePartitionDrawing() {
76         GamesExpo.age = 8;
77         GamesExpo.gender = "girl";
78         assertTrue(GamesExpo.canParticipateInDrawing(
    GamesExpo.age, GamesExpo.gender));
79     }
80
81     //Test 7: If the user is of age greater than 6,
then they should not be able to compete in rhyiming
82     @Test
83     void canParticipateInRhyiming() {
84         GamesExpo.age = 7;
85         assertFalse(GamesExpo.canParticipateInRhyiming(
    GamesExpo.age));
86     }
```

```
87
88     //Test 8: To test the domain using boundary
values analysis for boys and girls who can
participate in Rhyming
89     @Test
90     void boundaryVARhyming() {
91         GamesExpo.age = -1;
92         assertFalse(GamesExpo.canParticipateInRhyming(
93             GamesExpo.age));
94         GamesExpo.age = 0;
95         assertTrue(GamesExpo.canParticipateInRhyming(
96             GamesExpo.age));
97         GamesExpo.age = 7;
98         assertFalse(GamesExpo.canParticipateInRhyming(
99             GamesExpo.age));
100        GamesExpo.age = 6;
101        assertTrue(GamesExpo.canParticipateInRhyming(
102            GamesExpo.age));
103    }
104    //Test 9: To test the domain using EP technique
for boys and girls who can participate in
storytelling.
105    @Test
106    void equivalencePartitionRhyming() {
107        GamesExpo.age = 4;
108        assertTrue(GamesExpo.canParticipateInRhyming(
109            GamesExpo.age));
110    }
111    //Test 10: If the user is a boy and has an age on
the inclusive boundary, then they should be able to
compete in quiz
112    @Test
113    void canParticipateInQuiz() {
```

```
114         GamesExpo.age = 11;
115         GamesExpo.gender = "boy";
116         assertTrue(GamesExpo.canParticipateInQuiz(
GamesExpo.age, GamesExpo.gender));
117     }
118
119     //Test 11: To test the domain using boundary
values analysis for boys who can participate in quiz
120     @Test
121     void boundaryVAQuiz() {
122         GamesExpo.age = 10;
123         GamesExpo.gender = "boy";
124         assertFalse(GamesExpo.canParticipateInQuiz(
GamesExpo.age, GamesExpo.gender));
125
126         GamesExpo.age = 11;
127         GamesExpo.gender = "boy";
128         assertTrue(GamesExpo.canParticipateInQuiz(
GamesExpo.age, GamesExpo.gender));
129
130         GamesExpo.age = 16;
131         GamesExpo.gender = "boy";
132         assertFalse(GamesExpo.canParticipateInQuiz(
GamesExpo.age, GamesExpo.gender));
133
134         GamesExpo.age = 15;
135         GamesExpo.gender = "boy";
136         assertTrue(GamesExpo.canParticipateInQuiz(
GamesExpo.age, GamesExpo.gender));
137     }
138
139     //Test 12: To test the domain using EP technique
for boys who can participate in a quiz.
140     @Test
141     void equivalencePartitionQuiz() {
142         GamesExpo.age = 13;
143         GamesExpo.gender = "boy";
144         assertTrue(GamesExpo.canParticipateInQuiz(
```



```
144 GamesExpo.age, GamesExpo.gender));
145     }
146
147     //Test 13: If the user is a girl and has an age
on the inclusive boundary, then they should be able
to compete in essay writing
148     @Test
149     void canParticipateInEssayWriting() {
150         GamesExpo.age = 10;
151         GamesExpo.gender = "girl";
152         assertTrue(GamesExpo.
canParticipateInEssayWriting(GamesExpo.age, GamesExpo
.gender));
153     }
154
155     //Test 14: To test the domain using boundary
values analysis for girls who can participate in
essay writing
156     @Test
157     void boundaryVAEssay() {
158         GamesExpo.age = 9;
159         GamesExpo.gender = "girl";
160         assertFalse(GamesExpo.
canParticipateInEssayWriting(GamesExpo.age, GamesExpo
.gender));
161
162         GamesExpo.age = 10;
163         GamesExpo.gender = "girl";
164         assertTrue(GamesExpo.
canParticipateInEssayWriting(GamesExpo.age, GamesExpo
.gender));
165
166         GamesExpo.age = 16;
167         GamesExpo.gender = "girl";
168         assertFalse(GamesExpo.
canParticipateInEssayWriting(GamesExpo.age, GamesExpo
.gender));
169
```

```
170         GamesExpo.age = 15;
171         GamesExpo.gender = "girl";
172         assertTrue(GamesExpo.
    canParticipateInEssayWriting(GamesExpo.age, GamesExpo
        .gender));
173     }
174
175     //Test 15: To test the domain using EP technique
for boys who can participate in storytelling.
176     @Test
177     void equivalencePartitionEssay() {
178         GamesExpo.age = 13;
179         GamesExpo.gender = "girl";
180         assertTrue(GamesExpo.
    canParticipateInEssayWriting(GamesExpo.age, GamesExpo
        .gender));
181     }
182
183     //Test 16: If the user is any gender and has an
age on the inclusive boundary, then they should be
able to compete in poetry
184     @Test
185     void canParticipateInPoetry() {
186         GamesExpo.age = 20;
187         assertTrue(GamesExpo.canParticipateInPoetry(
    GamesExpo.age));
188     }
189
190     //Test 17: To test the domain using boundary
values analysis for girls who can participate in
poetry
191     @Test
192     void boundaryVAPoetry() {
193         GamesExpo.age = 19;
194         assertFalse(GamesExpo.canParticipateInPoetry(
    GamesExpo.age));
195
196         GamesExpo.age = 20;
```

```
197         assertTrue(GamesExpo.canParticipateInPoetry(
198             GamesExpo.age));
199         GamesExpo.age = 100;
200         assertTrue(GamesExpo.canParticipateInPoetry(
201             GamesExpo.age));
202     }
203     //Test 18: To test the domain using EP technique
204     for boys who can participate in storytelling.
205     @Test
206     void equivalencePartitionPoetry() {
207         GamesExpo.age = 40;
208         assertTrue(GamesExpo.canParticipateInPoetry(
209             GamesExpo.age));
210     }
211     //Test 19: If the user is any gender and has an
212     age on the inclusive boundary, then they should be
213     told that there are no events they can compete in
214     @Test
215     void canNotParticipate() {
216         GamesExpo.age = 16;
217         assertTrue(GamesExpo.canNotParticipate(
218             GamesExpo.age));
219     }
220     //Test 20: To test the domain using boundary
221     value analysis for any gender user who can not
222     participate in any event
223     @Test
224     void boundaryVACanNotParticipate() {
225         GamesExpo.age = 15;
226         assertFalse(GamesExpo.canNotParticipate(
227             GamesExpo.age));
228         GamesExpo.age = 17;
229         assertTrue(GamesExpo.canNotParticipate(
```

```
224 GamesExpo.age));
225
226         GamesExpo.age = 20;
227         assertFalse(GamesExpo.canNotParticipate(
    GamesExpo.age));
228     }
229
230     //Test 21: To test the domain using EP technique
    for any gender user who can not participate in any
    competition
231     @Test
232     void equivalencePartitionCanNotParticipate() {
233         GamesExpo.age = 18;
234         assertTrue(GamesExpo.canNotParticipate(
    GamesExpo.age));
235     }
236 }
```