

```

1 //Formatif4g2NP2234508
2 //Nikolay Filatov 2234508
3 //420-201-Introduction à la programmation
4 //Groupe 2
5 //Enseignant : Robert Aubé
6 import java.util.Scanner;
7
8 public class Formatif4G1NP2234508 {
9     public Formatif4G1NP2234508() {
10         testerEstExact();
11         testerPoserQuestionEtCorriger();
12         testerEstEntreLesBornes();
13         testerConvertirNoteACote();
14         lireNote(Integer.parseInt(lireString("Veuillez entrer une note qui sera convertie en
cote de A à F:"))));
15
16
17     }
18
19     private void lireNote(Integer reponse) {
20         if (estEntreBornes(reponse, 0, 100)) {
21             System.out.println("Vous avez entré la note : " + reponse + ", ce qui vous donne
la cote suivante: " + convertirNoteACote(reponse));
22         } else {
23             System.out.println(reponse + " Désolé, la note " + reponse + " est invalide.
Veuillez entrer une valeur entre 0 et 100 inclusivement");
24         }
25
26
27     }
28
29     private void testerConvertirNoteACote() {
30         System.out.println(convertirNoteACote(-1) == 'E');
31         System.out.println(convertirNoteACote(0) == 'E');
32         System.out.println(convertirNoteACote(59) == 'E');
33         System.out.println(convertirNoteACote(60) == 'D');
34         System.out.println(convertirNoteACote(70) == 'C');
35         System.out.println(convertirNoteACote(80) == 'B');
36         System.out.println(convertirNoteACote(90) == 'A');
37         System.out.println(convertirNoteACote(100) == 'A');
38         System.out.println(convertirNoteACote(101) == 'A');
39     }
40
41     private char convertirNoteACote(Integer note) {
42         if (note < 60) {
43             return 'E';
44         } else if (note < 70) {
45             return 'D';
46         } else if (note < 80) {
47             return 'C';
48         } else if (note < 90) {
49             return 'B';
50         } else if (note >= 90) {
51             return 'A';
52         }

```

```

53
54     return 0;
55 }
56
57 public void testerPoserQuestionEtCorriger() {
58     poserQuestionEtCorriger(lireString("Quelle est la capitale du Canada?"), "Ottawa"
59 );
60     poserQuestionEtCorriger(lireString("Quelle est la capitale de l'Angleterre?"), "
Londres");
61 }
62
63 public void poserQuestionEtCorriger(String réponse, String bonneRéponse) {
64     if (réponse.equalsIgnoreCase(bonneRéponse)) {
65         System.out.println("C'est la bonne réponse");
66     } else {
67         System.out.println("Ce n'est pas la bonne réponse");
68     }
69     System.out.println();
70 }
71
72 public void testerEstEntreLesBornes() {
73     System.out.println(estEntreBornes(74, 75, 80) == false);
74     System.out.println(estEntreBornes(75, 75, 80) == true);
75     System.out.println(estEntreBornes(80, 75, 80) == true);
76     System.out.println(estEntreBornes(81, 75, 80) == false);
77 }
78
79 public boolean estEntreBornes(Integer nombre, Integer min, Integer max) {
80
81     Boolean estEgal1;
82
83     if (nombre <= max && nombre >= min) {
84         estEgal1 = nombre <= max;
85     } else {
86         estEgal1 = false;
87     }
88     return estEgal1;
89
90 }
91
92
93 public void testerEstExact() {
94     System.out.println(estExact("abc", "abc") == true);
95     System.out.println(estExact("abc", "cbc") == false);
96     System.out.println(estExact("abc", "Abc") == true);
97 }
98
99 public boolean estExact(String s1, String s2) {
100
101     Boolean estEgal1 = false;
102
103     if (s1.equals(s2)) {
104         estEgal1 = s1.equals(s2);
105     } else if (!s1.equals(s2)) {

```

```
106     estEgal1 = s1.equalsIgnoreCase(s2);
107 }
108 return estEgal1;
109
110
111 }
112
113 public void testerEgaliteChaines() {
114     String str_L = "Limoilou";
115     String str_CL = "Cegep " + str_L;
116     String str_I = "limoilou";
117     boolean estEgal;
118
119     estEgal = str_L.equals(str_L);
120     System.out.println(str_L + " est égale à " + str_L + " -> " + estEgal);
121
122     estEgal = str_L.equals(str_CL);
123     System.out.println(str_L + " est égale à " + str_CL + " -> " + estEgal);
124
125     estEgal = str_L.equals(str_I);
126     System.out.println(str_L + " est égale à " + str_I + " -> " + estEgal);
127
128     estEgal = str_L.equalsIgnoreCase(str_I);
129     System.out.println(str_L + " est égale à " + str_I + " -> " + estEgal);
130 }
131
132 public String lireString(String q) {
133     Scanner sc = new Scanner(System.in);
134     System.out.println(q);
135     String name = sc.nextLine();
136     return name;
137 }
138
139 public static void main(String[] args) {
140
141     new Formatif4G1NP2234508();
142
143 }
144 }
145
146
```