```
1
2   //Name -
3   //Date -
4   //Class -
5   //Lab  -
6
7   import static java.lang.System.*;
8   import java.util.Scanner;
9
10  public class CipherRunner
11  {
12      public static void main( String args[] )
13      {
14          boolean playAgain = true;
15          while( playAgain )
16              playAgain = play();
17      }
18
19      private static boolean play(){
20          entryText();
21          String code = new String("");
22          int cryChoice = 0;
23          //the user enters a string
24          code = askCode();
25          //let the user pick the type of data encryption
26          cryChoice = askMethod();
27          //demonstrate encryption and decryption
28          displayResults(cryChoice, code);
29          //continue in a loop until the user chooses to stop
30          return playAgain();
31      }
32
33      private static void entryText() {
34          System.out.print('\u000C');
35
36          out.println("Welcome to the Cipher Runner, today you will be able to send
            secret messages to your friends");
37          out.println("Please enter a word or phrase");
38          out.print(":");
39
40      }
41
42      private static String askCode() {
43          String code = new String("");
44          Scanner keyboard = new Scanner(System.in);
45
46          boolean notValid = true;
47          while (notValid)
48              try {
49                  code = keyboard.nextLine();
50                  if( onlyAlpha(code) )
51                      notValid = false;
52                  else {
53                      out.println("you entered an invalid response, remember only
                        characters a-z, A-Z, and spaces are allowed");
54                      out.print(":");
55                  }
56              } catch (Exception e) {
57                  keyboard.next();
58                  out.println("you entered an invalid response, remember only characters
                    a-z, A-Z, and spaces are allowed");
59                  out.print(":");
60              }
61
62          return code;
63      }
```

```
64
65      private static int askMethod() {
66          int cryChoice = 0;
67          Scanner keyboard = new Scanner(System.in);
68
69          out.println("Thank you. Now please pick a meathod of encryption you would like
            to use: ");
70          out.println( "Your choices are:"      +
71                       "\n\t1) Cesear Excryption" +
72                       "\n\t2) Box Encryption"   );
73          out.print(":");
74
75          boolean notValid = true;
76          while (notValid)
77              try {
78                  cryChoice = keyboard.nextInt();
79                  if( cryChoice == 1 || cryChoice == 2 )
80                      notValid = false;
81                  else {
82                      out.println("you entered an invalid response, remember:");
83                      out.println( "Your choices are:"      +
84                                   "\n\t1) Cesear Excryption" +
85                                   "\n\t2) Box Encryption"   );
86                      out.print(":");
87                  }
88              } catch (Exception e) {
89                  keyboard.next();
90                  out.println("you entered an invalid response, remember:");
91                  out.println( "Your choices are:"      +
92                               "\n\t1) Cesear Excryption" +
93                               "\n\t2) Box Encryption"   );
94                  out.print(":");
95              }
96
97          return cryChoice;
98      }
99
100     private static void displayResults(final int cryChoice, final String code) {
101         TopLeftColumnRow box;
102         Caesar caesar;
103         switch(cryChoice) {
104             case 1:
105                 caesar = new Caesar( askShift() );
106
107                 out.print( "\"" + code + "\" encrypted with the Cesear Encryption looks
                    like: " );
108                 out.println(caesar.encode(code));
109                 out.print("After decrypting it with the Cesear Encryption looks like:
                    \"" );
110                 out.println( caesar.decode( caesar.encode(code) ) + "\"" );
111                 break;
112             case 2:
113                 box = new TopLeftColumnRow();
114
115                 out.print( "\"" + code + "\" encrypted with the Box Encryption looks
                    like: " );
116                 out.println( box.encode(code) );
117                 out.print("After decrypting it with the Box Encryption looks like: \"" );
118                 out.println( box.decode( box.encode(code) ) + "\"" );
119                 break;
120             default:
121                 out.println("IM A STUPID CODER");
122         }
123     }
124
125     private static int askShift() {
```

```
126             out.println("How long would you like the shift to be? (0 will randomize the
                shift)");
127             out.print(":");
128 
129             int choice = 0;
130             Scanner keyboard = new Scanner(System.in);
131 
132             boolean notValid = true;
133             while (notValid)
134                 try {
135                     choice = keyboard.nextInt();
136                     if( choice > 0 && choice < 27 ) {
137                         notValid = false;
138                     } else {
139                         out.println("you entered an invalid response, remember only numbers
                            from 0-26 are allowed");
140                         out.print(":");
141                     }
142                 } catch (Exception e) {
143                     keyboard.next();
144                     out.println("you entered an invalid response, remember only numbers
                        from 0-26 are allowed");
145                     out.print(":");
146                 }
147 
148             return choice;
149         }
150 
151         private static boolean playAgain() {
152             out.println("Would you like to play again? (Y/N)");
153             out.print(":");
154 
155             String ans = "";
156             boolean choice = false;
157             Scanner keyboard = new Scanner(System.in);
158 
159             boolean notValid = true;
160             while (notValid)
161                 try {
162                     ans = keyboard.nextLine();
163                     if       ( ans.toUpperCase().charAt(0) == 'Y' ) {
164                         notValid = false;
165                         choice = true;
166                     } else if( ans.toUpperCase().charAt(0) == 'N' ) {
167                         notValid = false;
168                         choice = false;
169                     } else {
170                         out.println("you entered an invalid response, remember:");
171                         out.println("(Y or N)");
172                         out.print(":");
173                     }
174                 } catch (Exception e) {
175                     keyboard.next();
176                     out.println("you entered an invalid response, remember:");
177                     out.println("(Y or N)");
178                     out.print(":");
179                 }
180 
181             return choice;
182         }
183 
184         private static boolean onlyAlpha(String name) {
185             char[] chars = name.toCharArray();
186             boolean isChar, isSpace;
187             isChar = isSpace = false;
188 
```

```
189             for (char c : chars) {
190                 if( Character.isLetter(c) )
191                     isChar = true;
192                 else if ( c != ' ' )
193                     isSpace = true;
194             }
195 
196             return isChar || isSpace;
197         }
198     }
```