

SDP lab 2 - Shadan Khan 1876267

Task1:

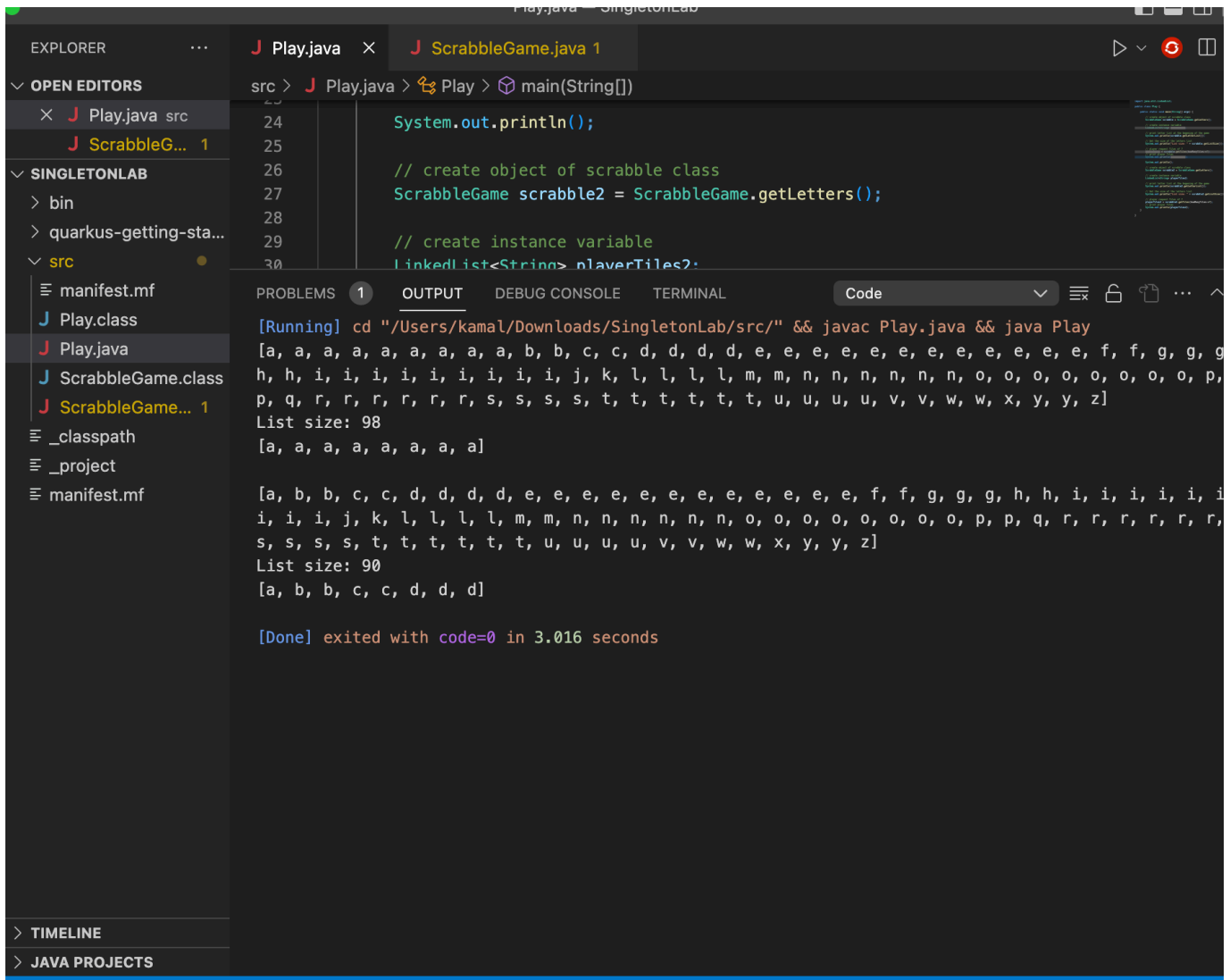
```
src > J Play.java > Play > main(String[])
1  import java.util.LinkedList;
2
3  public class Play {
4
5      Run | Debug
6      public static void main(String[] args) {
7
8          // create object of scrabble class
9          ScrabbleGame scrabble = ScrabbleGame.getLetters();
10
11         // create instance variable
12         LinkedList<String> playerTiles;
13
14         // print letter list at the begening of the game
15         System.out.println(scrabble.getLetterList());
16
17         // Get the size of the letters list
18         System.out.println("List size: " + scrabble.getListSize());
19
20         // player request Tiles of 7
21         playerTiles = scrabble.getTiles(howManyTiles: 7);
22         // print player Tiles
23         System.out.println(playerTiles);
24
25
26
27
28
29
30
31
32
33
34
35
36
```

```
src > J Play.java > Play > main(String[])
24  System.out.println();
25
26  // create object of scrabble class
27  ScrabbleGame scrabble2 = ScrabbleGame.getLetters();
28
29  // create instance variable
30  LinkedList<String> playerTiles2;
31
32  // print letter list at the begening of the game
33  System.out.println(scrabble2.getLetterList());
34
35  // Get the size of the letters list
36  System.out.println("List size: " + scrabble2.getListSize());
37
38  // player request Tiles of 7
39  playerTiles2 = scrabble2.getTiles(howManyTiles: 7);
40  // print player Tiles
41  System.out.println(playerTiles2);
42
43
44
45
```

```
src > J ScrabbleGame.java > ...
1
2 import java.util.Arrays;
3 import java.util.Collections;
4 import java.util.LinkedList;
5
6 public class ScrabbleGame {
7
8     String[] scrabbleLetters = { "a", "a", "a", "a", "a", "a", "a", "a", "a", "a",
9         "b", "b", "c", "c", "d", "d", "d", "d", "e", "e", "e", "e", "e",
10        "e", "e", "e", "e", "e", "e", "f", "f", "g", "g", "g", "h",
11        "h", "i", "i", "i", "i", "i", "i", "i", "j", "k", "l",
12        "l", "l", "l", "m", "m", "n", "n", "n", "n", "n", "n", "o", "o",
13        "o", "o", "o", "o", "o", "o", "p", "q", "r", "r", "r", "r",
14        "r", "r", "s", "s", "s", "s", "t", "t", "t", "t", "u",
15        "u", "u", "u", "v", "v", "w", "w", "x", "y", "y", "z", };
16
17     private LinkedList<String> letterList = new LinkedList<String>(Arrays.asList(scrabbleLetters));
18     private static ScrabbleGame letters = null;
19
20     // Constructor
21     private ScrabbleGame() {
22     }
23
24     // Method return a list of letters in the Game
25     public LinkedList<String> getLetterList() {
26         return this.letterList;
27     }
28
29     public static ScrabbleGame getLetters() {
30         if (letters == null) {
31             letters = new ScrabbleGame();
32         }
33         return letters;
34     }
35 }
```

```
src > J ScrabbleGame.java > ...
34
35 // Method return a list of letters for each player equal to the send integer
36 // value
37 public LinkedList<String> getTiles(int howManyTiles) {
38
39     // Tiles to be returned to the user
40     LinkedList<String> tilesToSend = new LinkedList<String>();
41
42     // Cycle through the LinkedList while adding the starting
43     // Strings to the to be returned LinkedList while deleting
44     // them from letterList
45
46     for (int i = 0; i <= howManyTiles; i++) {
47
48         tilesToSend.add(this.letterList.remove(index: 0));
49     }
50
51     // Return the number of letter tiles requested
52
53     return tilesToSend;
54 }
55
56 // Method return the size of the list
57 public Integer getListSize() {
58     Integer number = letterList.size();
59     return number;
60 }
61
62 }
63
64
65
66 }
```

Output:



The screenshot shows an IDE with two tabs: `Play.java` and `ScrabbleGame.java 1`. The `ScrabbleGame.java` tab is active, showing the following code:

```
src > J Play.java > Play > main(String[])
24      System.out.println();
25
26      // create object of scrabble class
27      ScrabbleGame scrabble2 = ScrabbleGame.getLetters();
28
29      // create instance variable
30      LinkedList<String> playerTiles2;
```

The `OUTPUT` tab is selected, displaying the execution results:

```
[Running] cd "/Users/kamal/Downloads/SingletonLab/src/" && javac Play.java && java Play
[a, a, a, a, a, a, a, a, a, a, a, b, b, c, c, d, d, d, d, d, e, e, e, e, e, e, e, e, e, e, e, e, e, f, f, g, g, g, h, h, i, i, i, i, i, i, i, i, i, i, j, k, l, l, l, l, m, m, n, n, n, n, n, n, n, n, n, o, o, o, o, o, o, o, o, o, o, p, p, q, r, r, r, r, r, r, s, s, s, s, s, t, t, t, t, t, t, t, t, u, u, u, u, v, v, w, w, x, y, y, z]
List size: 98
[a, a, a, a, a, a, a, a, a]

[a, b, b, c, c, d, d, d, d, d, e, e, e, e, e, e, e, e, e, e, e, e, e, f, f, g, g, g, h, h, i, i, i, i, i, i, i, i, i, j, k, l, l, l, l, m, m, n, n, n, n, n, n, n, n, n, n, n, o, o, o, o, o, o, o, o, o, o, p, p, q, r, r, r, r, r, r, s, s, s, s, s, t, t, t, t, t, t, t, t, t, u, u, u, u, u, v, v, w, w, x, y, y, z]
List size: 90
[a, b, b, c, c, d, d, d]

[Done] exited with code=0 in 3.016 seconds
```

Task2:

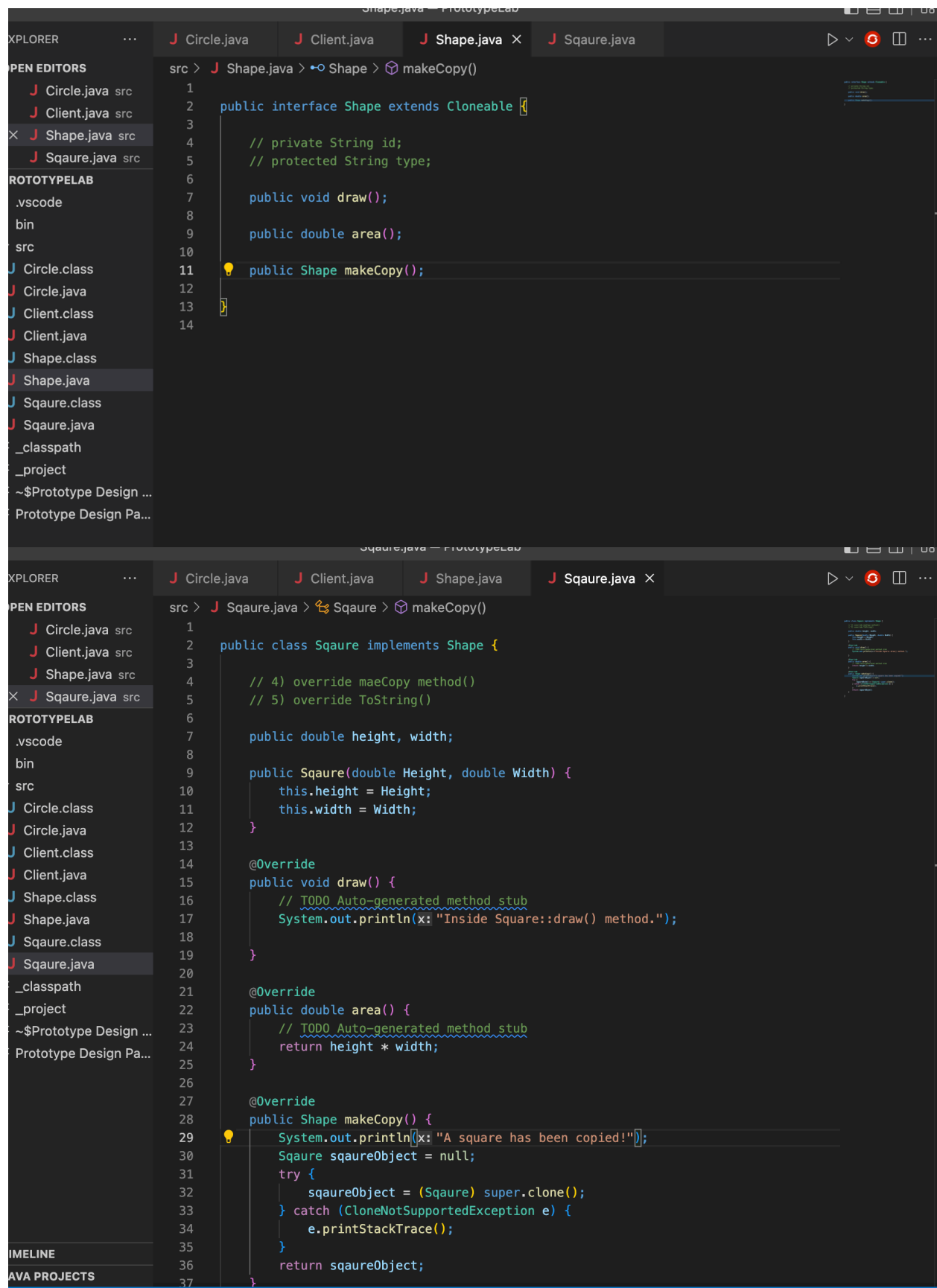
The image shows two screenshots of a VS Code editor window. The top screenshot displays the `Circle.java` file, and the bottom screenshot displays the `Client.java` file. Both files are part of a project named "PrototypeLab".

Circle.java

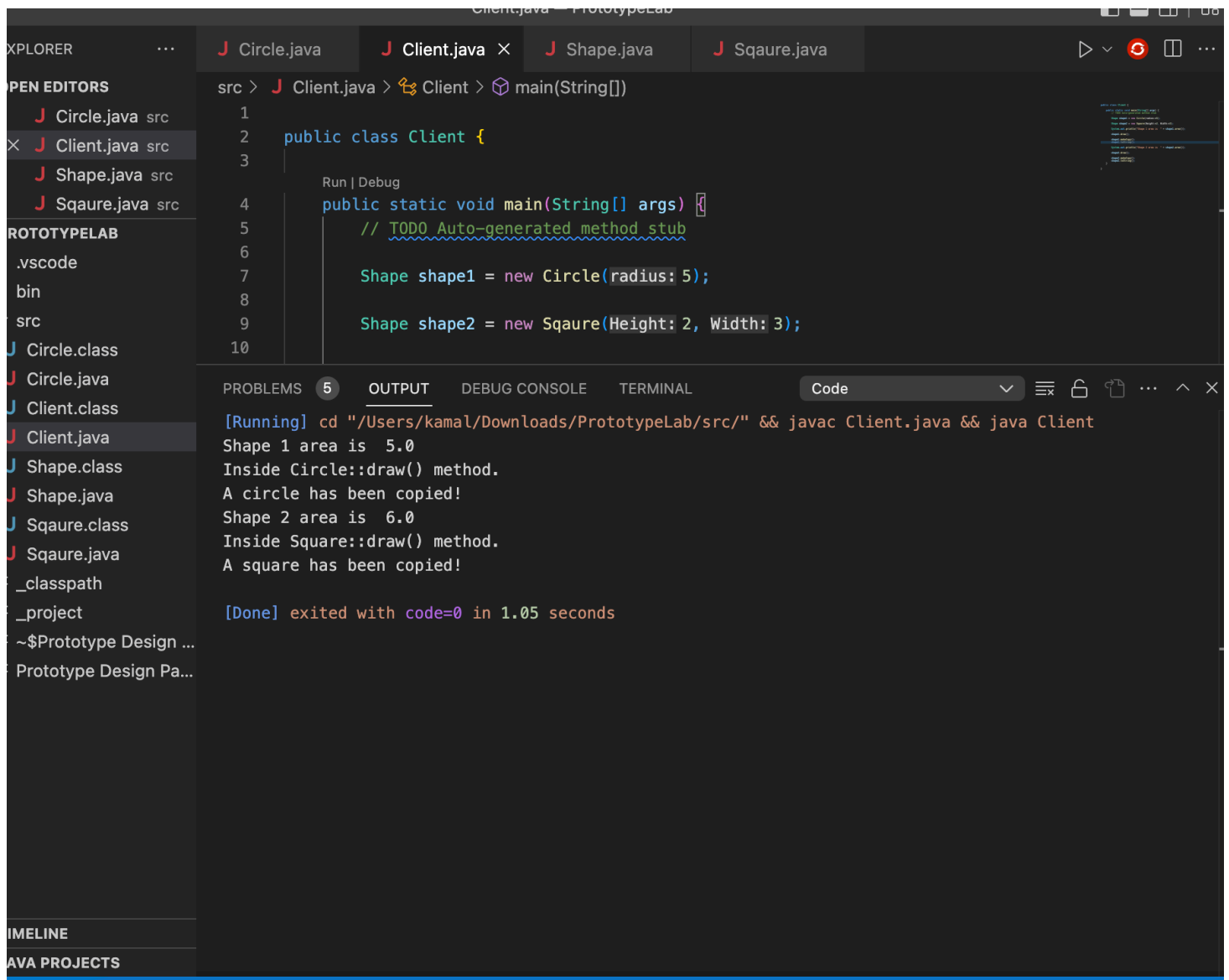
```
src > J Circle.java > Circle > makeCopy()
13  @Override
14  public void draw() {
15      // TODO Auto-generated method stub
16      System.out.println(x: "Inside Circle::draw() method.");
17  }
18
19
20  @Override
21  public double area() {
22      // TODO Auto-generated method stub
23      return radius;
24  }
25
26  @Override
27  public Shape makeCopy() {
28      System.out.println(x: "A circle has been copied!");
29      Circle circleObject = null;
30      try {
31          circleObject = (Circle) super.clone();
32      } catch (CloneNotSupportedException e) {
33          e.printStackTrace();
34      }
35      return circleObject;
36  }
37  }
```

Client.java

```
src > J Client.java > Client > main(String[])
1  public class Client {
2
3      Run | Debug
4      public static void main(String[] args) {
5          // TODO Auto-generated method stub
6
7          Shape shape1 = new Circle(radius: 5);
8
9          Shape shape2 = new Sqaure(Height: 2, Width: 3);
10
11         System.out.println("Shape 1 area is " + shape1.area());
12
13         shape1.draw();
14
15         shape1.makeCopy();
16         shape1.toString();
17
18         System.out.println("Shape 2 area is " + shape2.area());
19
20         shape2.draw();
21
22         shape2.makeCopy();
23         shape2.toString();
24     }
25 }
26
27 }
```



Output:



The screenshot shows a VS Code IDE with a project named 'PrototypeLab'. The Explorer sidebar on the left lists files: Circle.class, Circle.java, Client.class, Client.java, Shape.class, Shape.java, Sqaure.class, Sqaure.java, _classpath, and _project. The Open Editors tab shows four files: Circle.java, Client.java, Shape.java, and Sqaure.java. The Client.java file is active, showing the following code:

```
src > J Client.java > Client > main(String[])
1
2 public class Client {
3
4     Run | Debug
5     public static void main(String[] args) {
6         // TODO Auto-generated method stub
7
8         Shape shape1 = new Circle(radius: 5);
9
10        Shape shape2 = new Sqaure(Height: 2, Width: 3);
```

The Output panel at the bottom shows the execution results:

```
PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL Code
[Running] cd "/Users/kamal/Downloads/PrototypeLab/src/" && javac Client.java && java Client
Shape 1 area is 5.0
Inside Circle::draw() method.
A circle has been copied!
Shape 2 area is 6.0
Inside Square::draw() method.
A square has been copied!

[Done] exited with code=0 in 1.05 seconds
```