Programming Project 0

**Posting ID:** 5439-820

**Contents**

1 Contents 1

2 Pledge 1

3 Reflection 1

4 Source Code 2

5 Compilation Output 2

6 Testing Strategy 3

**Pledge**

*I pledge that this submission is entirely my own work. I have not attempted to find other solutions to the same problem, and I have not looked at or shown my code or write-up to any person other than the grader, tutors, or instructor. I understand that I may discuss ideas with others but I may not share code. I understand that, should I accidentally violate any of these conditions I must inform the grader and instructor immediately before submitting my work. I understand that if I am unable to explain aspects of my code to a grader when I am asked, then it will be considered cheating, and I will receive a grade of EX (failure due to a breach of academic integrity) in the course.*

**Signed**: [Armando Minor] [01-15-16]

**Reflection**

Beginning my project one of the first things I did was understand the

desired outcome. I took into consideration what would be the best approach for myself

tackling this project. My best approach was to take the whole project and divide it into

pieces to make the workload easier. This approach helped me compile the desired

outcome with less effort and less errors, although i did have a few. I ran a few tests

during and after my project was completed to ensure the desired result was achieved.

Once everything that was asked for was completed I submitted my work.

**Source Code**

1. /\*\*
2. \* Created by Armando Minor on 1/15/16.
3. \*/
4. //\* Program will print user input and output it.
5. //\* Program will exit after user input is printed if quit or Quit is entered!

8. **import** java.util.Scanner;

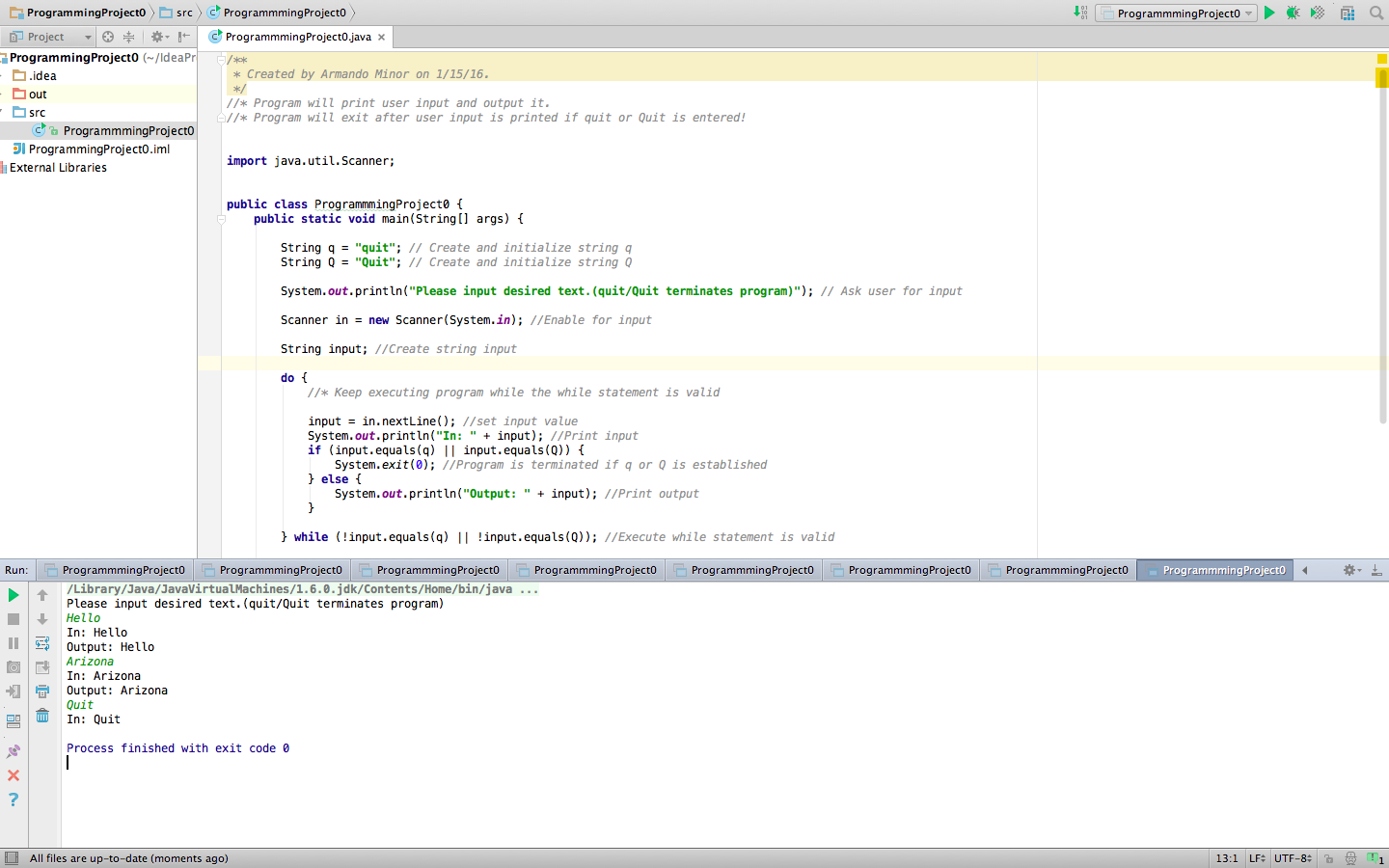
11. **public** **class** ProgrammmingProject0 {
12. **public** **static** **void** main(String[] args) {
14. String q = "quit"; // Create and initialize string q
15. String Q = "Quit"; // Create and initialize string Q
17. System.out.println("Please input desired text.(quit/Quit terminates program)"); // Ask user for input
19. Scanner in = **new** Scanner(System.in); //Enable for input
21. String input; //Create string input
23. **do** {
24. //\* Keep executing program while the while statement is valid
26. input = in.nextLine(); //set input value
27. System.out.println("In: " + input); //Print input
28. **if** (input.equals(q) || input.equals(Q)) {
29. System.exit(0); //Program is terminated if q or Q is established
30. } **else** {
31. System.out.println("Output: " + input); //Print output
32. }
34. } **while** (!input.equals(q) || !input.equals(Q)); //Execute while statement is valid





41. }
42. }

**Compilation Output**



**Testing Strategy**

My testing strategy for this assignment was too establish a running program first and then fill in the details. The first goal was to obtain code that was working before I added the details and remarks to the code. I ran tests when I first thought the test was complete to encounter errors, once the problems arose I began to fix each one independently. Taking into consideration that by fixing one problem might cause another one to occur. Soon after all errors were fixed and code was running with no errors I began to add the details.

