

KIT101 Programming Fundamentals

PP Task 5.3 Spot the Mistakes

Overview








Purpose: Increase your familiarity with common errors in syntax, program construction or program logic.

Task: Attempt to compile and run the five small programs, record what happens and what you might do to correct it.

Learning Outcomes: 1  2  5 

Time: Aim to complete this task before the start of Week 6.

Resources: *There are no resources dedicated solely to debugging, but the following may be relevant to features of the code samples you will examine in this task:*


- Introductory Programming Notes:
 - Essential Java Syntax templates
 - 04 Working with Primitive Data
 - 05 Using Objects
 - 06 Tracing Code by Hand
 - 08 Making Decisions
 - 09 Repeating Actions with Loops
- Online Mini-lectures:
 - Solving Problems with Computers:
 -  Using 'primitive' data
 -  Some commonly used objects
 - Making Decisions:
 -  Boolean expressions & comparing values
 -  Two-way branching with if and if-else
 -  Multi-way branching with switch
 - Repeating Actions with Loops:
 -  Loop components, while and do-while
 -  The for-loop and loop debugging tips

Submission Details

Upload the following to the MyLO submission folder for this task:

- A Word or PDF of the completed **answer sheet** (you do not need to submit any code)

Assessment Criteria

A  Completed submission will:

- Report the observable problems with each code sample
- Include reasonable suggestions for fixing the observed problems

Instructions

Being able to recognise and fix errors in program code is an important skill. In this task you will look at a series of independent programs and identify the causes of syntax (compilation time) and semantic (runtime) errors. Complete the accompanying answer sheet to document your findings and your suggested corrections.

To test each program use the following template and paste the provided code into main(), then modify the program's name and save it to its own file:

```
import java.util.Scanner;

/** Test harness */
public class ReplaceThisName {

    public static void main(String[] args) {
        //PASTE THE CODE HERE
    }

}
```

Some Sequence

1. Copy-paste the following source code into a new program using the template above:

```
Scanner sc = new Scanner(System.in);
int i, j, k;
```

```

int min, middle, max;

System.out.print("Enter three integers, separated by spaces: ");
i = sc.nextInt();
j = sc.nextInt();
k = sc.nextInt();

min = Math.min(i, Math.min(j, k));
max = Math.max(i, Math.max(j, k));
middle = i + j + k - min - max;
System.out.println("The middle value is " + middle);

```

2. **Attempt to compile it.**
3. Record the compiler error in full in the answer sheet.
4. Write a description in your own words of what you think the error means.
5. Write down what you would change to correct the error (there are several valid options).

Another Sequence

6. Copy-paste the following source code into a new program using the template above:

```

int a, b, c;

a = 10;
c = a + b;
b = 25;
System.out.println("a, b and c are " + a + ", " + b + " and " + c);

```

7. **Attempt to compile it.**
8. As with the first sample, record the compiler error in full and write down a description of the error in your own words and what you would change that would fix the error.

Reversed

9. Copy-paste the following source code into a new program using the template above. It is intended to print the user's input in reverse.

```

Scanner sc = new Scanner(System.in);
String text;

System.out.print("Enter some text: ");
text = sc.nextLine();

System.out.print("Your message reversed: ");
for (int i = text.length() - 1; i > 0; i--) {
    System.out.print( text.charAt(i) );
}
System.out.println();

```

10. **Compile and run the program.** Enter any word of your choosing.

11. In the answer sheet record:

- what input you used;
- what output you observed;
- what feature of the output helps you to identify the cause of the erroneous behaviour; and
- what you would change to correct the erroneous behaviour.

Print a List

12. Copy-paste the following source code into a new program using the template above. It is a small test program a student might write the learn about arrays of Strings.

```

String[] document = { "You", "call", "this", "a", "book?" };

System.out.println("There are " + document.length + " entries.");

for (int i = 1; i <= document.length; i++) {
    System.out.println(document[i] + " has " + document[i].length() + " letters.");
}

```

13. **Compile and run the program.**

14. In the answer sheet record:

- what output you observed (including any errors);
- which parts of the output (including the error output) you think are most useful in identifying the cause;

- a description in your own words of what you think the error means; and
- what you would change to correct the error.

Hint: Runtime errors like this are actually a little harder to interpret when running a program through DrJava because of the way it executes your programs (which is different to you using the `java` command on the command line). The last five lines of the error would not appear if you ran your program from the command line.

Conditional Café Entry

15. The following program fragment could be used as part of a system at a café entrance to warn if the café has reached its occupancy limit. (In its current form the program takes input from the user, but this would be replaced with input from a sensor near the door, counting arrivals and departures.). The code currently has a small but critical error and *always* reports the café as full. Investigate to identify it.

```
Scanner sc = new Scanner(System.in);  
final int CAPACITY = 26; //warn if number of patrons is above this  
int patrons; //the current number of patrons  
  
System.out.print("Enter current number of patrons: ");  
patrons = sc.nextInt();  
  
if (patrons > CAPACITY); {  
    System.out.println("Safe number of customers has been exceeded! Ask someone to leave.");  
    System.out.println("Make sure they've paid their bill first!");  
}
```

16. **Compile and run the program.**

- Enter a value larger than `CAPACITY` and, in the answer sheet, record what value you entered and what output you observed.
- Enter a value smaller than or equal to `CAPACITY` and, in the answer sheet, record what value you entered and what output you observed.

17. Record what you think the cause of the erroneous behaviour is, and hence what you would change to correct the error.

Do you need a hint?
