

## CSA0961 – JAVA

### PRACTISE 4\_1

1. The formula for converting gallons to liters is: 1 US gallon = 3.785 liters. This program will convert a specific number of gallons (10) to liters and then display the output. The concepts in this practice will be explored in more detail throughout the course. Create a new project, package, and java class with a main method. Use the code below as a starting point and complete the code for the program. (Name your package galToLit and class GalToLit).

```
package galToLit;

public class GalToLit {

public static void main(String[] args) {

    // declare variables double gallons=10;

    double liters=0; // add your calculation here //output the result to user

    System.out.println(gallons+" gallons equals "+liters+" liters"); } }
```

ANSWER :

```
package helloworld;

public class hellomain {

    public static void main(String[] args) {

        // declare variables

        double gallons = 10;

        double liters = 0;

        // add your calculation here

        liters = gallons * 3.785; // 1 US gallon = 3.785 liters

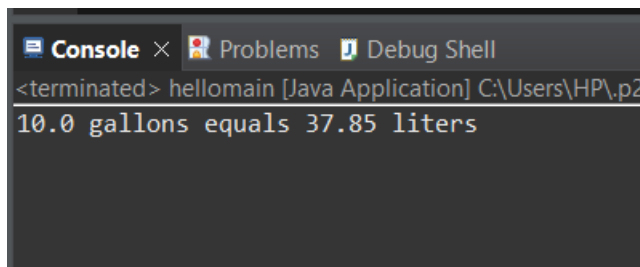
        // output the result to user

        System.out.println(gallons + " gallons equals " + liters + " liters");

    }

}
```

OUTPUT :

A screenshot of a Java IDE's console window. The window has a dark background and a light-colored text area. At the top, there are three tabs: 'Console', 'Problems', and 'Debug Shell'. The 'Console' tab is active. Below the tabs, the text '<terminated> hellomain [Java Application] C:\Users\HP\p2' is displayed. The main output of the program is '10.0 gallons equals 37.85 liters'.

2. The Scanner class can be used to accept input from the user. Modify the code written in step 2 to prompt a user for the number of gallons to compute. To declare an instance of the Scanner class, use the code below: `Scanner in = new Scanner(System.in);` Your Java IDE may prompt you to import the `java.util.Scanner` package, or you can manually enter the import statement between the package name and the class declaration as shown below:

```
package galToLit;

import java.util.Scanner;

public class GalToLit {
```

To get a decimal value from the user, use the `in.nextDouble()` method and assign to the gallons variable.

ANSWER :

```
package helloworld;
```

```
import java.util.Scanner;
```

```
public class hellomain {

    public static void main(String[] args) {

        // Create a Scanner object to read input
        Scanner in = new Scanner(System.in);

        // Declare variables
        double gallons = 0;
        double liters = 0;

        // Prompt the user for the number of gallons
        System.out.print("Enter the number of gallons: ");
```

```

// Get the number of gallons from user input
gallons = in.nextDouble();

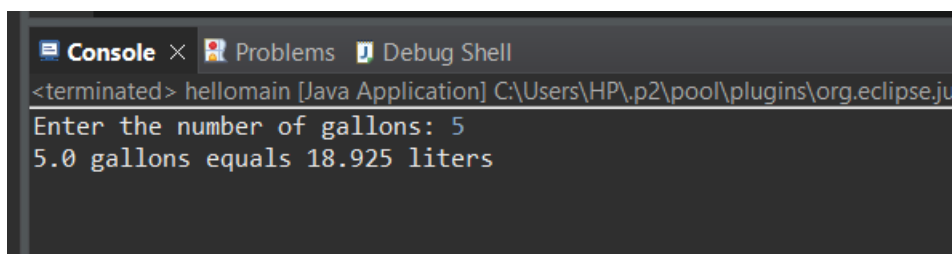
// Perform the conversion
liters = gallons * 3.785; // 1 US gallon = 3.785 liters

// Output the result to the user
System.out.println(gallons + " gallons equals " + liters + " liters");

// Close the scanner
in.close();
}
}

```

OUTPUT :



```

Console x Problems Debug Shell
<terminated> hellomain [Java Application] C:\Users\HP\p2\pool\plugins\org.eclipse.j
Enter the number of gallons: 5
5.0 gallons equals 18.925 liters

```

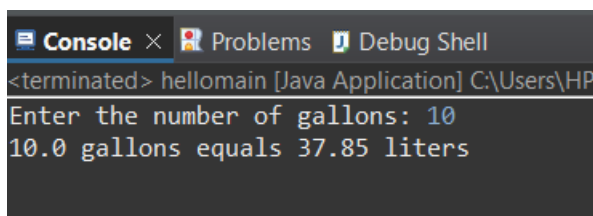
3. Describe three ways you can test the program that converts gallons to liters.

TESTING THE ABOVE PROGRAM WITH DIFFERENT INPUTS :

- **Example Inputs and Expected Outputs:**

Input: 10 gallons

Expected Output: 10 gallons equals 37.85 liters



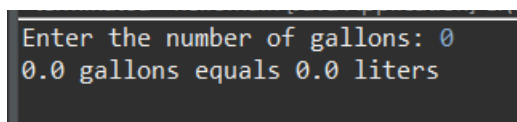
```

Console x Problems Debug Shell
<terminated> hellomain [Java Application] C:\Users\HP
Enter the number of gallons: 10
10.0 gallons equals 37.85 liters

```

Input: 0 gallons

Expected Output: 0 gallons equals 0.0 liters



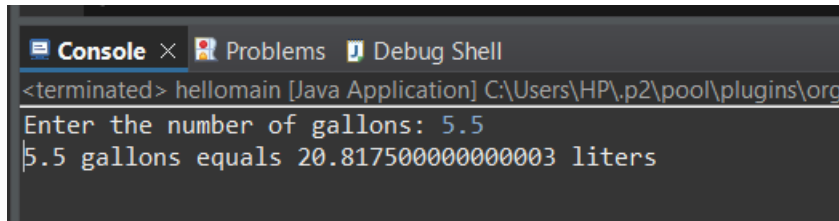
```

Enter the number of gallons: 0
0.0 gallons equals 0.0 liters

```

Input: 5.5 gallons

Expected Output: 5.5 gallons equals 20.8275 liters



```
Console x Problems Debug Shell
<terminated> hellomain [Java Application] C:\Users\HP\p2\pool\plugins\org
Enter the number of gallons: 5.5
5.5 gallons equals 20.81750000000003 liters
```

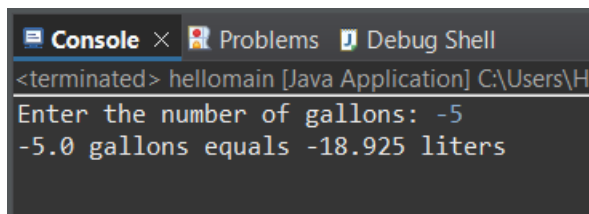
**Verification:** Compare the program's output against the expected results to ensure accuracy.

**Rationale:** This tests the core functionality of the conversion algorithm with a range of valid inputs.

### Example Edge Cases:

**Negative Input:** Input: -5 gallons

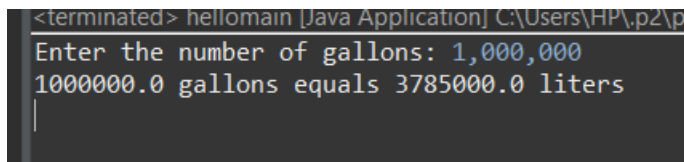
Expected Output: -5 gallons equals -18.925 liters (Verify the program handles negative values appropriately, or decide if negative inputs should be restricted.)



```
Console x Problems Debug Shell
<terminated> hellomain [Java Application] C:\Users\H
Enter the number of gallons: -5
-5.0 gallons equals -18.925 liters
```

**Large Input:** Input: 1,000,000 gallons

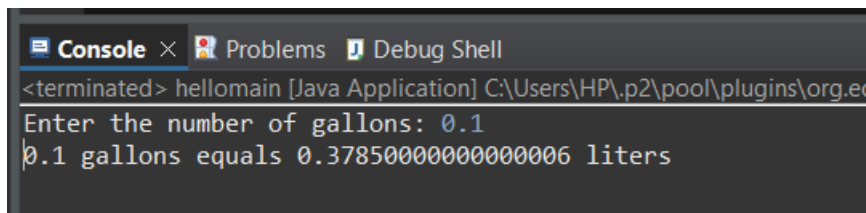
Expected Output: 1,000,000 gallons equals 3,785,000 liters (Check if the program can handle very large numbers without performance issues or overflow.)



```
<terminated> hellomain [Java Application] C:\Users\HP\p2\p
Enter the number of gallons: 1,000,000
1000000.0 gallons equals 3785000.0 liters
```

**Decimal Input:** Input: 0.1 gallons

Expected Output: 0.1 gallons equals 0.3785 liters (Ensure the program handles small decimal values accurately.)



```
Console x Problems Debug Shell
<terminated> hellomain [Java Application] C:\Users\HP\p2\pool\plugins\org.ec
Enter the number of gallons: 0.1
0.1 gallons equals 0.3785000000000006 liters
```

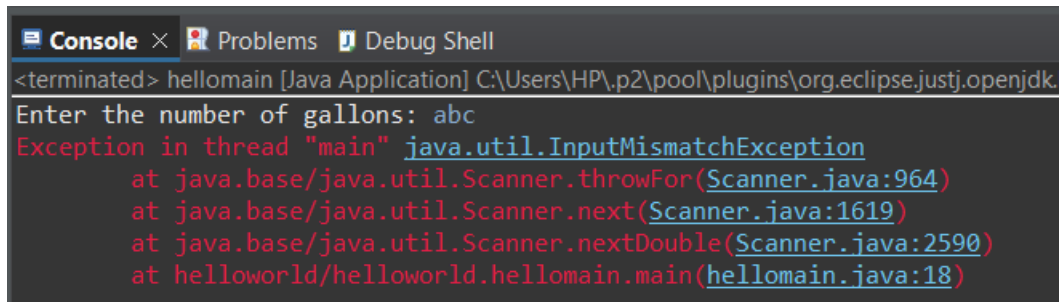
**Rationale:** Testing edge cases helps ensure the program is robust and behaves as expected in less common scenarios.

### Test for Invalid Inputs

**Example Invalid Inputs:**

### Non-Numeric Input: Input: abc

Expected Behavior: The program should either handle the error gracefully (e.g., with an error message) or prompt the user to enter a valid number.



```
Console × Problems Debug Shell
<terminated> hellomain [Java Application] C:\Users\HP\p2\pool\plugins\org.eclipse.justj.openjdk
Enter the number of gallons: abc
Exception in thread "main" java.util.InputMismatchException
    at java.base/java.util.Scanner.throwFor(Scanner.java:964)
    at java.base/java.util.Scanner.next(Scanner.java:1619)
    at java.base/java.util.Scanner.nextDouble(Scanner.java:2590)
    at helloworld/helloworld.hellomain.main(hellomain.java:18)
```