



UNIVERSITY OF THE GAMBIA  
PROGRAMMING 1 (JAVA)



---

*iGURU*

---

*Group Memembers:*

muhammed s BALDEH  
MAT No: 21516027

muhammed DANSO  
MAT No: 21526019

June 3, 2015

## **Abstract**

The iGURU application is a multi purpose application which help users to accomplice different tasks. The application provide five option for the user to choose from. The option are multiplication table drawer, a currency converter, a temperature unit converter, a length unit converter and finally a weight unit converter. So this report show the full implementation of the application, its features and also the user manual.

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Organization . . . . .	3
<b>2</b>	<b>Design</b>	<b>4</b>
<b>3</b>	<b>Implementation</b>	<b>6</b>
<b>4</b>	<b>Conclusion</b>	<b>7</b>
 Appendix:		
<b>A</b>	<b>Future Work</b>	<b>7</b>
<b>B</b>	<b>Contribution</b>	<b>8</b>
<b>C</b>	<b>Classes</b>	<b>8</b>
<b>D</b>	<b>User Manual</b>	<b>8</b>

# 1 Introduction

The main aim of this application is to help users with multiple solution with numbers, form which the name iGuru is derived from. So that users don't have to install many different application for doing different calculation, but with iGuru all this calculation can be done within one platform. The application also contain a feature which make the application user friendly and make sure that users enjoy using the application. This programming project which is part of the Java 1 programming course help to expose students to the Java library which developers can learn different Java methods for implementing certain features in a Java application, which we the iGuru developers do to implement features like back action, invalid entry detector and so on.

## 1.1 Organization

This report mainly serves as a documentation for the iGuru project. As the purpose of any documentation is to mainly serve as a reference, this report therefore serves to describe the implementation of the project. As such, the report is organize as follows:

1. The **introduction** serves to give a description of the project, for the purpose of introducing readers to the content of the report.
2. The **design** part presents the algorithm designed that were used guide users through the program.
3. We discuss how the design was transform to code in the implementation section of the report.
4. As testing is an integral component of software design, we test the implementation of our `renice_d` command and respectively discuss the results of the tests in the **testing** and **results** section of the report.

## 2 Design

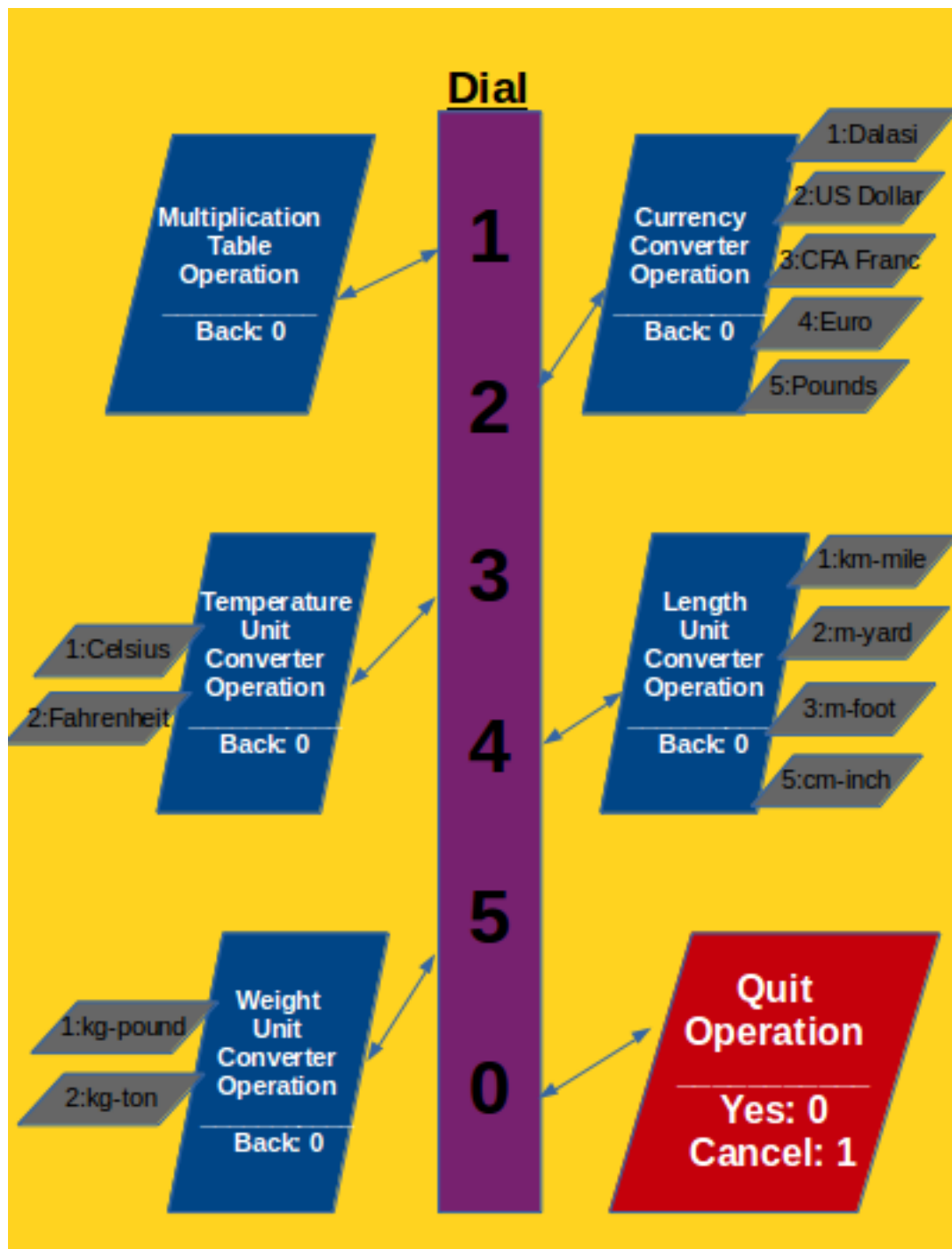


Figure 1: iGuru Options and Quit Flow Chart

The first step in the program is to print the five options in the program from which the users can choose what operation they want to do.

```

1 *****
2     Choose between the five option:
3 -----
4 To draw a multiplication Table;          Dial 1.
5 To convert Currency;                    Dial 2.
6 To convert Temperature Units;          Dial 3.
7 To convert Length Units;                Dial 4.
8 To convert Weight Units;                Dial 5.
9
10 [Quit: 0]
    [Option]:

```

Listing 1: Option Print

The program will then enter one of the loops which the users choose to do the operation in those loops. The next entry will then be asked. In any stage when 0 is entered this will close that present loop and return the program to the previous loop.

In some of the operation like the length and weight unit converter, when converting from between two units the user can either convert from the first printed unit or can enter 0 to convert from the next unit to the first unit.

Eg:

```

1 -----
2 M – FOOT CONVERTER
3
4
5 Enter amount in (m) unit:0
6 Enter amount in (foot) unit:34
7 -----
8 Result: 34.0 foot = 10.3632 m.
9

```

In the program code when a entry is correct and valid the boolean valid is set true to accomplish that loop else that boolean valid is set false. Which will make the while loop repeat and give the user another chance.

A while loop is also used to quit the program by setting all the necessary loops true to prevent them from repeating again. And finally running a (break;) method. The user can still cancel the quit process by dialling 1 to cancel quit.

```

1
2 ++++++
3     DO YOU WANT TO EXIT iGURU?
4     [ Ok: 0 ]          [ Cancel: 1 ]
5

```

```

6
7 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
8                                GOOD BYE
9 xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

```

### 3 Implementation

This application show a typical algorithm in order. The program first run the introduction codes and only proceed when the user enters the right entry between Option: 1, 2, 3, 4 and 5, or the quit option 0. This then set the boolean valid1 to true, to make sure that the program do not repeat that step.

```

1 System.out
2     .println("*****\n
3         WELCOME TO iGURU SERVICES");
4
5     // loop to check if dial option is in range
6     boolean valid1 = false;
7     while (!valid1) {
8
9         // The three main options lister
10        System.out
11            .print("\n*****\n
12                + "\n-----"
13                + "\nTo draw a multiplication Table;          Dial 1."
14                + "\nTo convert Currency;                      Dial 2."
15                + "\nTo convert Temperature Units;           Dial 3."
16                + "\nTo convert Length Units;                Dial 4."
17                + "\nTo convert Weight Units;                Dial 5."
18                + "\n
19
20                [Quit: 0]
21                + "\n[Option:]");
22
23        // main Scanner for options
24        @SuppressWarnings("resource")
25        Scanner dialScanner = new Scanner(System.in);
26
27        String dial = dialScanner.next();
28
29        // if dial in range if condition
30        if (dial.equals("1") || dial.equals("2") || dial.equals("3")
31            || dial.equals("4") || dial.equals("5")
32            || dial.equals("0")) {
33            valid1 = true;
34        } else {
35            System.out

```

```

35         .println("\n*Wrong entry! Please try again by choosing
36         between the five options.*");
    }

```

Listing 2: The iGuru Multiplication Table Class

The algorithm will then proceed and enter the loop which the user direct to. In some of the operations loops the users also have to choose in one of the loops present in that operation loop. After completing a loop the boolean valid at the beginning of that operational loop is set false so that the program does not return to the initial option printer loop, this is because the user might be interested in doing a similar calculation to the previous calculation.

In the length and weight operational loops, when the user choose the units to use to convert to and from. The program always ask the user to enter the value of the first unit printed. But when the user don't want to convert from the first printed unit, he or she should enter 0 as the value of the first printed unit. The program will then detect and ask the user to enter value in the second unit. The user can still enter 0 as the value of the second unit, this will return the program to the previous loop as a normal back action.

In all the stages a while loop is used to facilitate the back action, wrong entry retrial and also repeating operation. As every while loop is only executed when the boolean valid related to it is false and not executed when it is true.

## 4 Conclusion

iGuru is a very simple to use application when the option and instructions are carefully followed. This is made possible by the algorithm design used in the codes. And the iGuru team are working to make the program more simple, useful and fun to operate in the upcoming versions.

## A Future Work

In the future this application will be developed into a GUI program in become more user interactive. We are planning to also include a simple calculator into the program.



## B Contribution

The iGuruTester and MultiplicationTable class is designed by muhammed Baldeh and also prepare the project report. And muhammed Danso design the Currency, temperature, length and weight unit converter classes.

## C Classes

There are six classes in this project. A main tester class and other five classes doing the calculation.

Class Name	Description
iGuruTester	Print and scan option selection, then connect to other classes.
MultiplicationTable	Compute multiplication table calculation to the required limit.
CurrencyUnit	Compute currency conversion calculation from required currency.
TemperatureUnit	Compute temperature unit conversion calculation.
LengthUnit	Compute length unit conversion conversion calculation.
WeightUnit	Compute weight unit conversion conversion calculation.

Table 1: List of iGuru classes

## D User Manual

iGuru have five uses which options are displayed first after iGuru header or after restart. Users are to choose one of the options by dialing the number corresponding to that option; which are 1 for Multiplication Table option, 2 for Currency Converter option, 3 for Temperature Unit Converter option, 4 for Length Unit Converter option and then 5 for Weight Unit Converter option. The application will then move to the next level and do the processing.

-Multiplication Table.

-Currency Converter.

-Temperature Unit Converter.

-Length Unit Converter.

-Weight Unit Converter.

- **MULTIPLICATION TABLE** This option draw a multiplication table based on your choice of limit. So enter a whole number value ( preferably a small value less than 100 ) as your limit. The program wont process any other value other than a whole number. The program will then calculate and print out the table.

- **CURRENCY CONVERTER** This option convert currency values of five main currencies;

1. Gambian Dalasi
2. United State Dollar
3. West African CFA FRANC
4. European Euro
5. British Pounds Starling

Choose one of the five currencies from which you want to convert from and then enter the amount. After choosing from which currency to convert from, users then have to enter the amount of money they want to convert. The program will then calculate and print out the result from the currency you choose to the other remaining currencies.

- **TEMPERATURE UNIT CONVERTER** This option convert temperature units in Celsius or Fahrenheits. After choosing from which unit to convert from then enter the amount. The program will then calculate and print out the result.

- **LENGTH UNIT CONVERTER** This option convert length units among four major groups;

1. km – mile
2. m – yard
3. m – foot
4. cm – inch

First choose between the four option, then enter the amount in km, m or cm to convert to the corresponding units. If you want to convert to the other units, then enter amount of the first unit as 0, and then enter amount in mile, yard, foot or inch. The program will then calculate and print out the result.

- **WEIGHT UNIT CONVERTER** This option convert weight units among two major groups;

1. kg – pound
2. kg - ton

First choose between the two option, then enter the amount in kg to convert to the corresponding units. If you want to convert to the other units, then enter amount of the first unit as 0, and then enter amount in pound or ton. The program will then calculate and print out the result.

At any stage of the program user can enter 0 to step back or quit if at the main option stage. The program can be quieted by entering 0 till the quit verification appears. Please read instructions carefully while using the program to avoid waste of time.