Extended Java: Lab Assignment 2

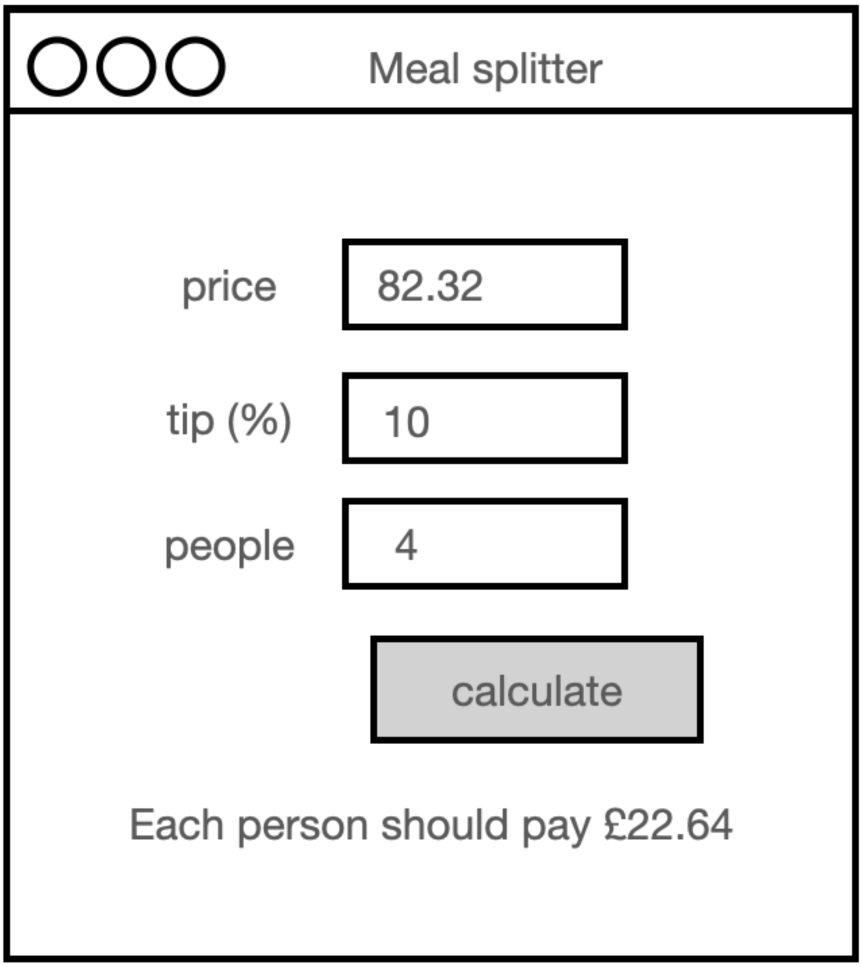
This assignment covers GUIs and enumerations. It is out of 50 marks, 30 for question 1 and 20 marks for question 2. You must complete your answer using the IntelliJ template provided. In both questions the style and quality of your code will be taken into consideration when allocating marks.

Question 1: Tip Calculator (30 marks)

Complete this question in the `TipCalculator` class. You may create other classes as part of your answer, although, you don't need to and this won't get you extra marks.

Write a class that implements a program to calculate each diner's share of a meal in a restaurant. The application should take the bill price, add a tip expressed as a percentage, and then divide by the people present. For full marks you should correctly handle erroneous input. You must use the Swing library as presented in class without using an interface builder or other tool.

When finished your GUI will resemble the following. You should make use of a single flow layout manager in your implementation, so resizing will change the layout.



Meal splitter

price: 82.32

tip (%): 10

people: 4

Each person should pay £22.64

Question 2: Change Calculator (20 marks)

Complete the change calculator. In this question you will write an Enumeration for the various coins and notes available in pounds sterling.

NotesAndCoins Enumeration (7 marks)

Complete the enumeration with the following constants, You also need to create a constructor to set the properties to match the table below and getterMethods for `valueInP` and `name`. These must be called getValueInP() and getName()

Constant	valueInP	name
PENCE1	1	“1p”
PENCE2	2	“2p”
PENCE5	5	“5p”
PENCE10	10	“10p”
PENCE20	20	“20p”
PENCE50	50	“50p”
POUND1	100	“£1”
POUND2	200	“£2”
POUND5	500	“£5”
POUND10	1000	“£10”
POUND20	2000	“£20"
POUND50	5000	“£50”

calcChange method (8 marks)

Complete the `calcChange` method it takes two arguments:

- A double called `price`, the amount the shopping costs
- A double called `paid`, How much money was handed over.

The method should return a `TreeMap`. The Tree map’s keys are the collection of constants in the `NotesAndCoins` enumeration and the values are how many of each denomination is required to make up the correct change.

For example, if the price is £28.99 and the paid amount is £40.00 the TreeMap returned would contain:

{pound50=0, pound20=0, pound10=1, pound5=0, pound2=0, pound1=1, pence50=0, pence20=0, pence10=0, pence5=0, pence2=0, pence1=1}

For the TreeMap you will need to create an implementation of `Comparator` to order the keys. It is recommended you order from largest denomination (£50.00) to the smallest (1p), as this will simplify your task.

Extension: ChangeCalculator GUI (5 marks)

Only attempt this part when you have completed everything else. It is worth an insubstantial number of marks relative to the required effort to complete.

Create a GUI for the ChangeCalculator. Use the `ChangeCalcGUI` class for your implementation. The design of your GUI is up to you. Marks will be awarded for the effective use of Swing complements, the quality of your code and having a usable layout.

Again, you must use the Swing library as presented in class without using an interface builder or other tool.