

Sofia University
Department of Mathematics and Informatics

Course : OO Programming Java

Date: December 13, 2018

Student Name:

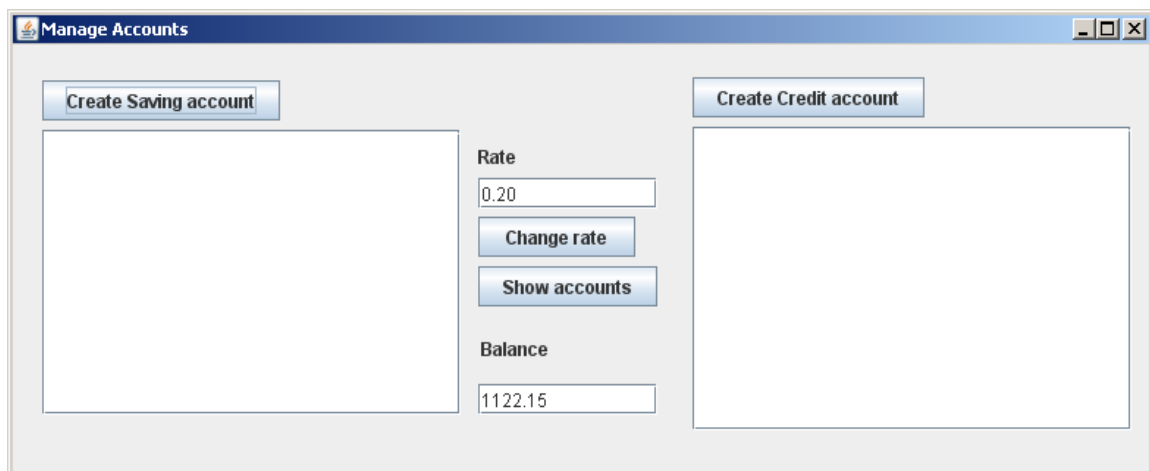
Lab No. 10b

Решете следните задачи като използвате JavaFX FXML приложение. Да се използват подходящи имена за всяка от променливите, рефериращи съответните графични контроли като се използват префикси в съответствие със следния стил за именуването им.

Контрола	Префикс
Button	<i>btn</i>
ComboBox	<i>cbo</i>
CheckBox	<i>chk</i>
Label	<i>lbl</i>
ListBox	<i>lst</i>
MainMenu	<i>mnu</i>
RadioButton	<i>rdb</i>
PictureBox	<i>pic</i>
TextBox	<i>txt</i>

Задача 1

Използвайте графичния редактор на SceneBuilder и създайте следния графичен интерфейс с JavaFX



Нека Контролера на приложението да има масив `arrAccounts` с 10 елемента от тип `Account` и има целочислена променлива `current`, чиято стойност е колко от

елементите на този масив са инициализирани в даден момент. В началото `current=0`.

- При натискане на бутона `Create Saving Account` се създава обект от клас `class SavingAcc` с баланс съответстващ на въведеното в текстовото поле `Balance`. Този обект се присвоява на масива от сметки `arrAccounts[current]` и `current` се увеличава с единица. Проверявайте за максимално валидната стойност на `current`.
- При натискане на бутона `Create Credit Account` се създава обект от клас `class CreditCardAcc` с баланс, съответстващ на въведеното в текстовото поле `Balance`. Този обект се присвоява на масива от сметки `arrAccounts[current]` и `current` се увеличава с единица. Проверявайте за максимално валидната стойност на `current`.
- При натискане на бутона `Change Rate` се променя лихвата на всички обекти от `class SavingAcc` в съответствие с въведената стойност в текстовото поле `Rate`
- При натискане на бутона `Show accounts` се обхождат в цикъл инициализираните елементи на масива `arrAccounts` и тези от елементите, които са `class SavingAcc` или `class CreditCardAcc` се извеждат с подробна информация (използва се `toString()` метода) съответно в лявата и дясната текстова област на графичния интерфейс

Задача 2

Напишете JavaFX приложение, което възпроизвежда следния графичен интерфейс

\$

1	2	3	Enter
4	5	6	
7	8	9	
0	.	Total	
			Delete
			Clear

Subtotal: \$

Tax: \$

Това приложение позволява на потребителя да въведе последователност от стойности на покупки като използват числовите бутони на графичния интерфейс.

- При натискане на бутона Enter текущо въведената стойност се прибавя към преди това въведените стойности и се извежда така получената сума в полето Subtotal
- При натискане на бутона Total се извежда сумата на всички въведени стойности в полето Subtotal, а полето Tax се извежда стойността на ДДС, което е 20% от общата сума на всички покупки

Задача 3

Напишете JavaFX приложение, което да преобразува от стойности от Fahrenheit temperature в Celsius еквивалентна стойност като използва графичен потребителски интерфейс за вход и изход на данните и изпълнение на преобразуванията на стойностите по начин аналогичен на изпълнението на задача

1

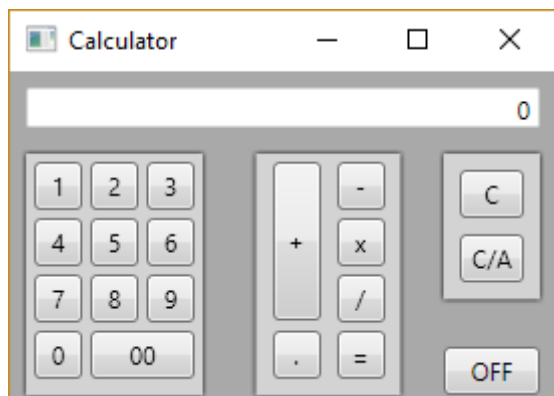
- Прочита цяло число от текстово поле на приложението, което е Fahrenheit temperature, при натискане на бутон "Compute Celsius"
- пресмята Celsius еквивалентна стойност по следната формула

$$\text{Celsius} = 5.0 / 9.0 * (\text{Fahrenheit} - 32)$$
- изобразява пресметнатата Celsius еквивалентна стойност в текстово поле на приложението
- При натискане на бутон "Compute Fahrenheit" пресмята Fahrenheit еквивалентна стойност на въведената в текстовото поле (с) като се използва формула обратна на б)
- Изобразява пресметнатата Fahrenheit еквивалентна стойност в текстово поле (а) на приложението
- Добавете възможност за преобразуване на температури по Келвин като използвате следната формула и графичен интерфейс

$$\text{Kelvin} = \text{Celsius} + 273.15$$

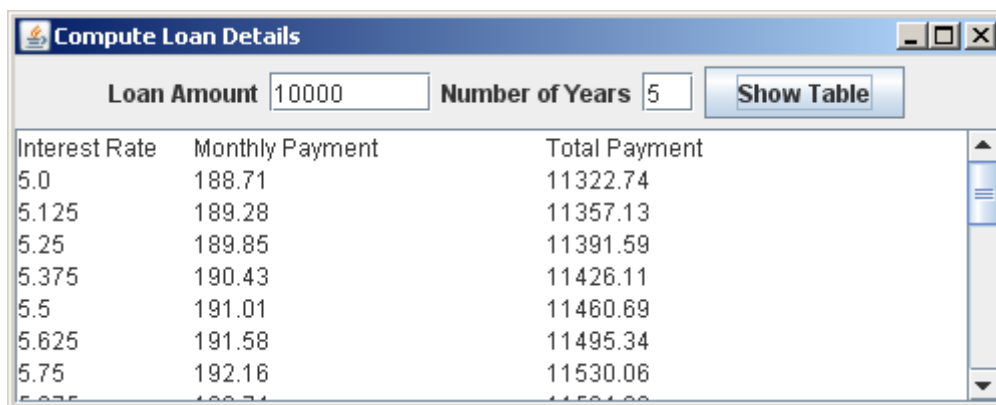
Задача 4

Да се създаде графично приложение на калкулатор следния графичен интерфейс



Задача 5

Да се създаде графично приложение на калкулатор следния графичен интерфейс



Напишете `class LoanComponent`, който има

```
annualInterestRate // годишен лихвен процент на заем
numberOfYears      // брой години за отпускане на заем
loanAmount         // отпусната парична сума на заема
```

Нека, освен изискваните **конструктори**, **SET/GET** методи да има и метод `computeLoan()`, който пресмята (връща) **общата стойност на заема** (отпуснат заем и обща стойност платени месечни вноски през периода на заема) за дадени стойности на `annualInterestRate`, `numberOfYears` и `loanAmount` като се използва **формулата**

$$\frac{\text{loanAmount} * \text{monthlyInterestRate}}{1 - \left(\frac{1}{1 + \text{monthlyInterestRate}} \right)^{\text{numberOfYears} * 12}}$$

Напишете обработка на събитието за бутона **Show Table**, при което се **извежда в таблица по горе указания формат стойността на заема, месечните вноски** за изменение на годишната лихва в интервала **5%- 8%** със стъпка **1/8**. Използвайте за тези пресмятания **данните въведени от потребителя за сума на заема (Loan Amount) и брой години за**

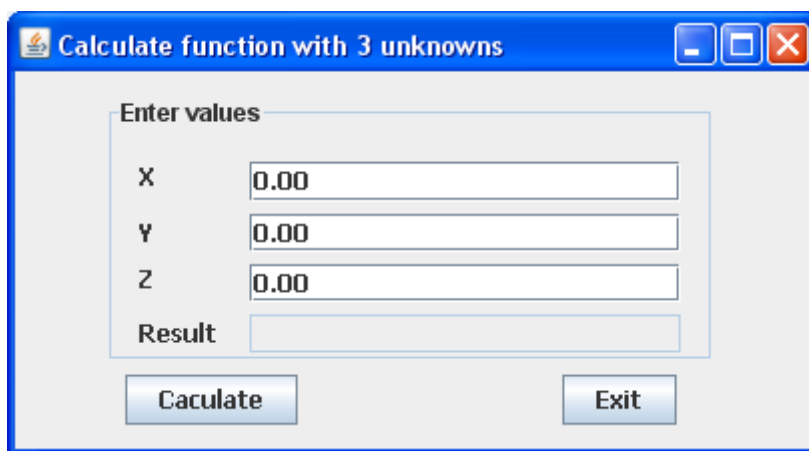
отпускане на заема (Number of Years) в съответните текстови полета на графичния интерфейс

Задача 6

Напишете JavaFX приложение, което въвежда в текстови полета на графичния интерфейс 3 числа в плаваща запетая от тип `BigDecimal`. Нека са означени чрез **x**, **y** и **z**.

При натискане на бутон `Compute` приложението да пресмята стойността на следния израз

$$4 \frac{x^2}{x^2 + 1} - 9(y + xz) + \frac{3 + y(2 + x)}{y^2 + 4}$$



и извежда въведените стойности за **x**, **y** и **z** в **Result**. Ако получената стойност е по- голяма от 10 тази стойност да се изведе в диалогов прозорец от тип `AlertType.ERROR`, съпътствана с подходящ текст. Заглавието на диалоговия прозорец да изписва показания примерен текст за заглавие на прозореца.

Забележка: Създайте обект от тип `NumberFormat` с `getInstance()` и използвайте метода `format()` на този обект за форматиране на числата от тип `BigDecimal`