CMPE 230

Project-3 Report

Students: Alperen Değirmenci – 2017400255

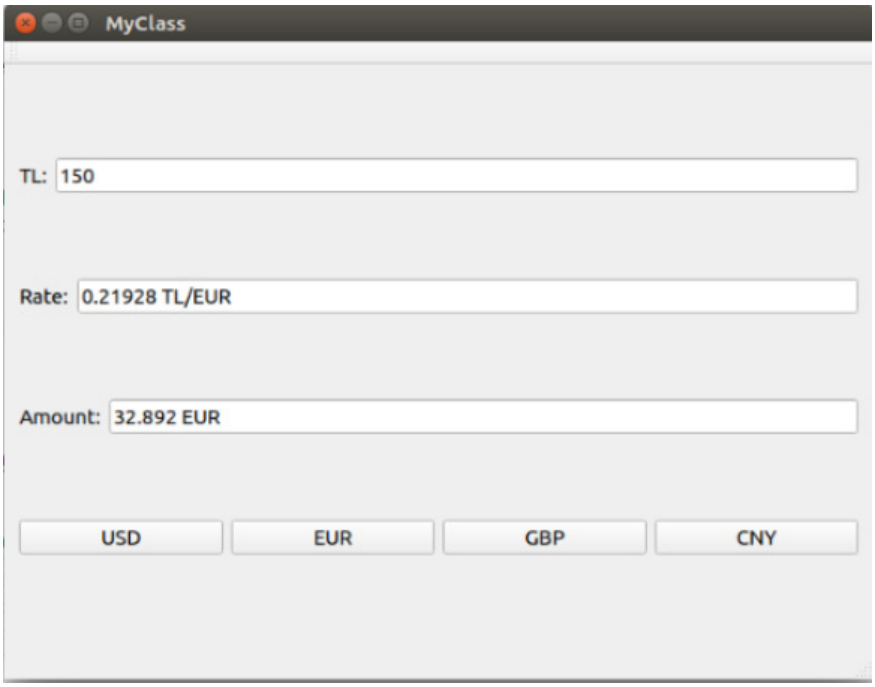
Hasan Öztürk – 2017400258

Instructor: Can Özturan

1- Description and the Solution to the Problem

In this project the task is to develop a QT program which converts given currencies. Four currency rates for **TRY / (USD, EUR, GBP** and **CNY)** are given and program should convert the amount of Turkish Lira input given by user to one of those currencies above. Currency rates are obtained from the **fixer.io** site.

Our project is based on a class named **MyClass** and the **user interface(.ui).** MyClass includes all the processes that make the program work. Our final user interface looks like this:



In **MyClass.h** file there is the **constructor**, **.ui** connection, **QMainWindow** initializer, **buttonClicked** functions, **network** access object and **currency rate** strings.

MyClass.cpp starts with the constructor and **user interface** sets up. Network access object gets the currency rates from the web page https://api.fixer.io/latest?base=TRY**.** Conversion of the currency rates to string objects are done in the **MyClass::replyFinished** function which is a parameter of the **SIGNAL – SLOT** process of the **network** access object. Currency rates are stored in the strings **strUSD, strEUR, strGBP** and **strCNY,** they are created and assigned via regular expressions like the following:

QRegExp rxUSD("((USD\":(\\d+\\.\\d+))) ");

This regex line finds the **USD** currency rate in the web page and **rxUSD.cap(3)** function takes only the **ratio** part (**e.g. : 0.25835**). Then these strings are assigned to ratio strings. For example **strUSD**.

In the next step, there are the button clicked functions. For example, let's take **EUR** rate and see how the **EUR** button works:

```
void MyClass::on_eurButton_clicked()
{
1    ui->rateLine->setText(strEUR + " TL/EUR" );
2    QRegExp rx1("[^tlTL ]+");
3    QString myString;
4    if ( rx1.indexIn(ui->tlLine->text(), 0) != -1 ) {
        myString = rx1.cap();    // rate found
    }
5    double tlValue = myString.toDouble();
6    double rate = strEUR.toDouble();
7    double b = (rate*tlValue);
8    QString abc = QString::number(b);
9    ui->amountLine->setText(abc + " EUR");
}
```

1st line puts the **TRY/EUR** ratio and the "TL/EUR" string to the **Rate LineEdit.**

2nd line defines the regular expression to take the correct number from the **TL LineEdit.**

3rd line defines a string and in the **4th** part, number from **TL LineEdit** is assigned to the string.

5th line casts the string to a double named **tlValue.**

6th line casts the **TRY/EUR** ratio string to a double named **rate.**

7th line multiplies the **tlValue** and **rate** and assigns it to the double **b** which is total amount in **EUR**.

8th line casts the double back to a **QString** variable.

9th line puts the total amount string and the **EUR** to the **amountLine LineEdit.**

In general this is how the code works.

2- How We Run the Program

- We wrote the program with the help of the codes sent by the instructor.
- But we didn't use **QLabel** instead, we used **QMainWindow** and **User Interface.**
- We wrote and tested the program in **QtCreator 5.10.0.**
- We are sending you the full project. Hope it also works fine in your testing PC.
- **Note:** We didn't try to run the project from the terminal.