Archived:Como criar uma calculadora simples, em Qt

Aquivado: Este artigo foi arquivado, pois o conteúdo não é mais considerado relevante para se criar soluções comerciais atuais. Se você achar que este artigo ainda é importante, inclua o template {{ForArchiveReview|*escreva a sua justificativa*}}.

Qt Quick should be used for all UI development on mobile devices. The approach described in this article (based on QWidget) is deprecated.

Este trecho de código demonstra como criar uma simples calculadora usando Qt.

Arquivo de cabeçalho

Cal.h

```
#ifndef CAL_H
#define CAL_H

#include <QtGui/QWidget>
#include "ui_cal.h"
#include<QString>
#include<QPushButton>
#include<QLineEdit>
#include<QHBoxLayout>
#include<QVBoxLayout>
#include<QFormLayout>
class cal : public QWidget
{
    Q_OBJECT

public:
 cal(QWidget *parent = 0);
    ~cal();

private:
    QPushButton *but1;
    QPushButton *sub;
    QPushButton *mul;
    QPushButton *div;
    QPushButton *eq;
    QFormLayout *lay;
    QLineEdit *line;
    QString str;
    QString str1;
    QString ch;
    QString strResult;
private slots:
   void addi();
   void subs();
   void mult();
   void divi();
   void em();
public:
        float num;
        float num1;
        float ans;
```


(C) Copyright Nokia 2014. All rights reserved.

```
};

#endif // CAL_H
```

Código fonte

cal.cpp

```
#include "cal.h"

cal::cal(QWidget *parent)
    : QWidget(parent)
{
 lay=new QFormLayout(this);
        but1=new QPushButton("+",this);
        sub=new QPushButton("-",this);
        mul=new QPushButton("*",this);
        div=new QPushButton("/",this);
        eq=new QPushButton("=",this);
        line=new QLineEdit(this);
        line->setMaxLength(12);
        lay->addWidget(line);
        lay->addWidget(but1)
        lay->addWidget(sub);
        lay->addWidget(mul);
        lay->addWidget(div);
        lay->addWidget(eq);
        setLayout(lay);
        QObject::connect(but1,SIGNAL(clicked()),this,SLOT(addi()));
        QObject::connect(sub,SIGNAL(clicked()),this,SLOT(subs()));
        QObject::connect(mul,SIGNAL(clicked()),this,SLOT(mult()));
        QObject::connect(div,SIGNAL(clicked()),this,SLOT(divi()));
        QObject::connect(eq,SIGNAL(clicked()),this,SLOT(em()));
}

cal::~cal()
{
 // No need to delete any object that got a parent that is properly deleted.

}
void cal::addi()
{
                str=line->text();
                num=str.toFloat();
                ch=but1->text();
                line->clear();
}
void cal::subs()
{
                str=line->text();
                num=str.toFloat();
                ch=sub->text();
                line->clear();
}
void cal::mult()
```


(C) Copyright Nokia 2014. All rights reserved.

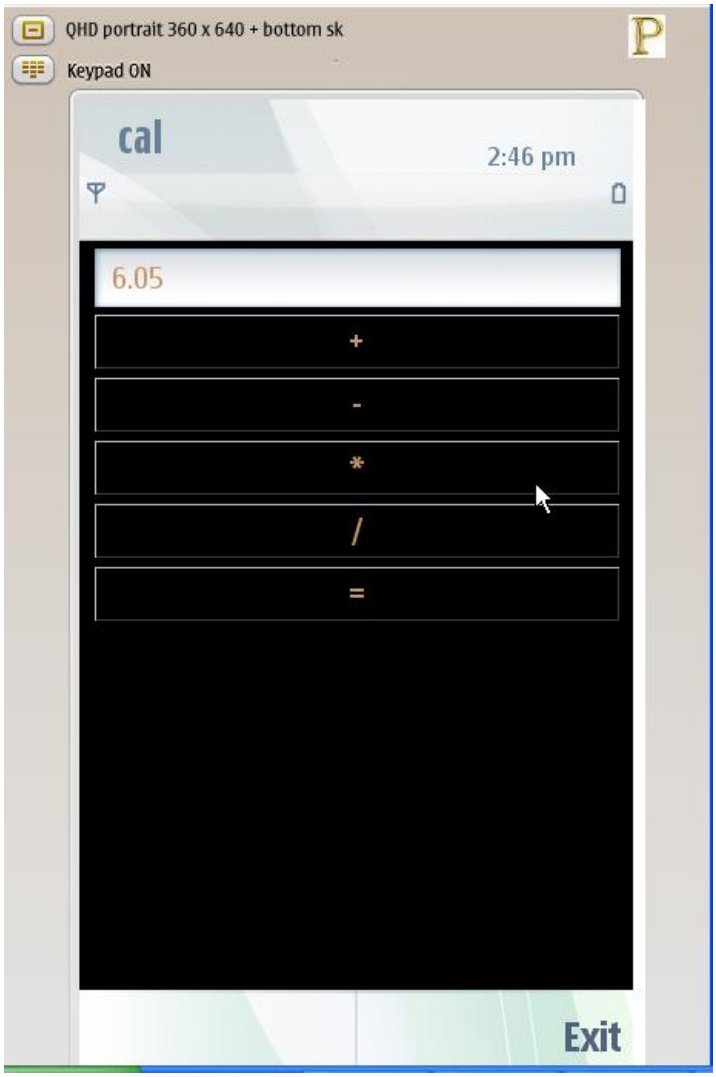
```
{
                str=line->text();
                num=str.toFloat();
                ch=mul->text();
                line->clear();
}
void cal::divi()
{
                str=line->text();
                num=str.toFloat();
                ch=div->text();
                line->clear();
}
void cal::em()
{


        str1 = line->text();
        num1 = str1.toFloat();
        if (ch=="+")
        {
                ans=num+num1;
        }
        else if(ch=="-")
        {
                ans=num-num1;
        }
        else if(ch=="*")
        {
                ans=num*num1;
        }
        else if(ch=="/")
        {
                ans=num/num1;
        }
        else
        {
               strResult = strResult.number(ans);
                line->setText(strResult);
        }
        strResult = strResult.number(ans);
        line->setText(strResult);
        ch="";
}
```

Captura de tela

- Entrada de algum valor na linha de edição.

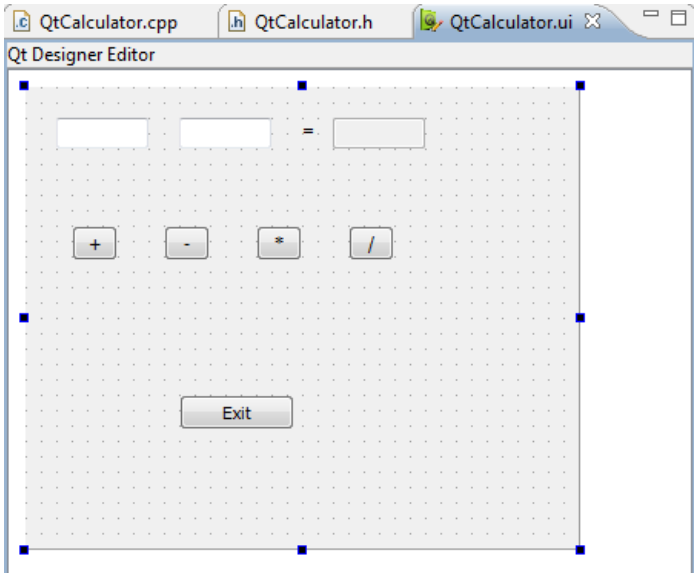
(C) Copyright Nokia 2014. All rights reserved.



- Operação decimal também é possível na calculadora.

Design da UI

QtCalculator.ui



Arquivo de cabeçalho

(C) Copyright Nokia 2014. All rights reserved.

QtCalculator.h

```
#ifndef QTCALCULATOR_H
#define QTCALCULATOR_H

#include <QtGui/QWidget>
#include "ui_QtCalculator.h"

class QtCalculator : public QWidget
{
    Q_OBJECT

public:
 QtCalculator(QWidget *parent = 0);
    ~QtCalculator();

public slots:
 void Addition();
 void Subtraction();
 void Multiplication();
 void Division();

private:
    Ui::QtCalculatorClass ui;
};

#endif // QTCALCULATOR_H
```

Código fonte

QtCalculator.cpp

```
#include "QtCalculator.h"

QtCalculator::QtCalculator(QWidget *parent)
    : QWidget(parent)
{
 ui.setupUi(this);
 QObject::connect(ui.plusButton, SIGNAL(clicked()), this, SLOT(Addition()));
 QObject::connect(ui.minusButton,SIGNAL(clicked()), this, SLOT(Subtraction()));
 QObject::connect(ui.multiplyButton,SIGNAL(clicked()), this, SLOT(Multiplication()));
 QObject::connect(ui.divisionButton,SIGNAL(clicked()), this, SLOT(Division()));
}

QtCalculator::~QtCalculator()
{
 QString strFisrt = ui.lineEditNumber1->text();
 QString strSecond = ui.lineEditNumber2->text();

 bool ok;
 int number1 = strFisrt.toInt(&ok);
 int number2 = strSecond.toInt(&ok);
 int result = number1 + number2;
```


(C) Copyright Nokia 2014. All rights reserved.

```
 QString strResult;
 strResult = strResult.number(result);
 ui.lineEditResult->setText(strResult);
}

void QtCalculator::Addition()
{
 QString strFisrt = ui.lineEditNumber1->text();
 QString strSecond = ui.lineEditNumber2->text();

 bool ok;
 int number1 = strFisrt.toInt(&ok);
 int number2 = strSecond.toInt(&ok);
 int result = number1 + number2;

 QString strResult;
 strResult = strResult.number(result);
 ui.lineEditResult->setText(strResult);
}

void QtCalculator::Subtraction()
{
 QString strFisrt = ui.lineEditNumber1->text();
 QString strSecond = ui.lineEditNumber2->text();

 bool ok;
 int number1 = strFisrt.toInt(&ok);
 int number2 = strSecond.toInt(&ok);
 int result = number1 - number2;

 QString strResult;
 strResult = strResult.number(result);
 ui.lineEditResult->setText(strResult);
}

void QtCalculator::Multiplication()
{
 QString strFisrt = ui.lineEditNumber1->text();
 QString strSecond = ui.lineEditNumber2->text();

 bool ok;
 int number1 = strFisrt.toInt(&ok);
 int number2 = strSecond.toInt(&ok);
 int result = number1 * number2;

 QString strResult;
 strResult = strResult.number(result);
 ui.lineEditResult->setText(strResult);
}

void QtCalculator::Division()
{
 QString strFisrt = ui.lineEditNumber1->text();
 QString strSecond = ui.lineEditNumber2->text();
```

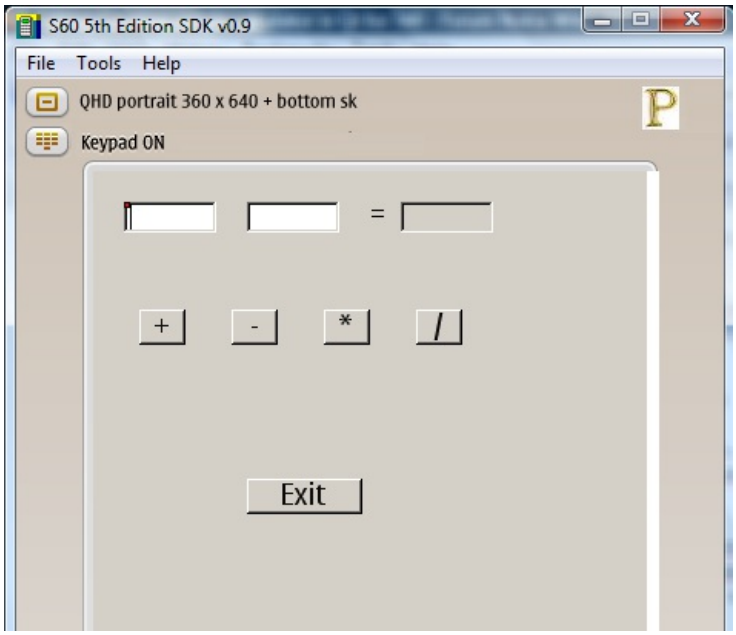

(C) Copyright Nokia 2014. All rights reserved.

```
  bool ok;
  int number1 = strFisrt.toInt(&ok);
  int number2 = strSecond.toInt(&ok);
  int result = number1 / number2;

  QString strResult;
  strResult = strResult.number(result);
  ui.lineEditResult->setText(strResult);
 }
```

Conclusões

- Será mostrada um simples calculadora na tela.



Exemplo

- Faça o download do exemplo trabalhado neste link: QtCalculator.zip

Feedback

- Se você econtrar algum problema na implementação desta simples calculadora, então por favor escreva seu *feedback* na página deste artigo Comment page (Inglês). Eu tentarei dá o máximo para tentar ajudar você.

Comentário

Código de erro resolvido.

(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.