JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY, NOIDA

INTGT-CSE SEMESTER II

REPORT FOR MINI PROJECT IN C++

TITLE OF PROJECT

'SCIENTIFIC CALCULATOR'

Supervision of:

Mrs Mradula Sharma

ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER

SCIENCE & ENGINEERING

JIIT, SECTOR-62, NOIDA

Submitted by:

Name	Enroll No.
ARYAN SINGH	22803008
SHIVAM CHAUDHARY	22803025
ABHINAV SINGH	22803010
SABEEH AHSAN	22803006

Introduction

Aim:

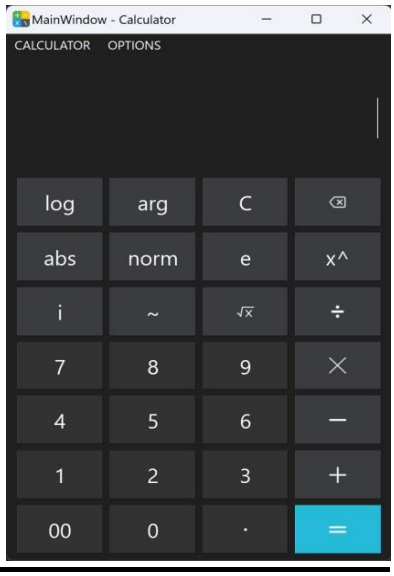
To develop a Working Calculator GUI using Qt framework for C++ Project.

About the project:

This project is made possible by Qt framework to make GUI for scientific calculator in C++. It can perform various operations like Basic arithmetic operations (add, subtract, multiply, divide). It can also perform complex calculations like calculating logarithm, to find trigonometric ratios. And at last, it can record the calculations performed by the user.

GLIMPSE OF CALCULATOR

- ***Dark Theme***



- ***Light Theme***

Features of the Project:

1. Dark and Light theme
2. Complex Calculations
3. Simple Calculations
4. Log and Trigonometric calculations

Operations used:

- Class
- Inheritance
- Maths functions
- Vector
- Try and catch
- Stack
- File Handling

CODE:-

//Calculator

1) Calculator.pro

```
QT       += core gui

greaterThan(QT_MAJOR_VERSION, 4): QT += widgets

CONFIG += c++17

# You can make your code fail to compile if it uses deprecated APIs.
# In order to do so, uncomment the following line.
#DEFINES += QT_DISABLE_DEPRECATED_BEFORE=0x060000    # disables all the
APIs deprecated before Qt 6.0.0

SOURCES += \
    about.cpp \
    complex.cpp \
    functions.cpp \
    main.cpp \
    record.cpp \
    standard.cpp

HEADERS += \
    about.h \
    complex.h \
    functions.h \
    record.h \
    standard.h

FORMS += \
    about.ui \
    complex.ui \
    functions.ui \
    record.ui \
    standard.ui
```

```
# Default rules for deployment.
qnx: target.path = /tmp/$${TARGET}/bin
else: unix:!android: target.path = /opt/$${TARGET}/bin
!isEmpty(target.path): INSTALLS += target

RESOURCES += \
    Resource.qrc

DISTFILES +=
```

//Headers

1) about.h

```
#ifndef ABOUT_H
#define ABOUT_H

#include <QDialog>

namespace Ui {
class about;
}

class about : public QDialog
{
    Q_OBJECT

public:
    explicit about(QWidget *parent = nullptr);
    ~about();

private:
    Ui::about *ui;
};

#endif // ABOUT_H
```

2) complex.h

```
#ifndef COMPLEX_H
#define COMPLEX_H

#include "record.h"
#include <QMainWindow>
#include <QApplication>
#include <about.h>
#include <record.h>
class MainWindow;
class MainWindow4;
namespace Ui {
class MainWindow2;
}

class MainWindow2 : public QMainWindow
{
    Q_OBJECT
```

```
public:
    explicit MainWindow2(QWidget *parent = nullptr);
    ~MainWindow2();

private slots:


    void on_actionSTANDARD_triggered();

    void on_actionLOG_TRIGO_triggered();

    void on_actionEXIT_triggered();

    void on_button_0_clicked();

    void on_button_1_clicked();

    void on_button_2_clicked();

    void on_button_3_clicked();

    void on_button_4_clicked();

    void on_button_5_clicked();

    void on_button_6_clicked();

    void on_button_7_clicked();

    void on_button_8_clicked();

    void on_button_9_clicked();

    void on_button_iota_clicked();

    void on_button_plus_clicked();

    void on_button_equal_clicked();
    void on_button_minus_clicked();

    void on_button_mul_clicked();

    void on_button_clear_2_clicked();

    void on_button_clear_clicked();

    void on_button_divide_clicked();

    void on_button_conjugate_clicked();

    void on_button_root_clicked();

    void on_button_power_clicked();

    void on_button_dzero_clicked();

    void on_button_decimal_clicked();
```

```
    void on_button_log_clicked();

    void on_actionABOUT_triggered();

    void on_button_argument_clicked();

    void on_actionBRIGHT_triggered();

    void on_actionDARK_triggered();

    void on_button_absolute_clicked();

    void on_button_normal_clicked();

    void on_button_exp_clicked();

    void fileEdit(QString a,QString b);

    void on_actionRECORD_triggered();

private:
    Ui::MainWindow2 *ui;
    MainWindow *standard;
    MainWindow4 *log;
    about *abt;
    record *rec;
};

#endif // COMPLEX_H
```

3) functions.h

```
#ifndef FUNCTIONS_H
#define FUNCTIONS_H

#include "record.h"
#include <QMainWindow>
#include <about.h>
#include <QApplication>
#include <record.h>
class MainWindow;
class MainWindow2;
class MainWindow3;
namespace Ui {
class MainWindow4;
}

class MainWindow4 : public QMainWindow
{
    Q_OBJECT

public:
    explicit MainWindow4(QWidget *parent = nullptr);
    ~MainWindow4();

private slots:
```

```
void on_actionSTANDARD_triggered();
void on_actionCOMPLEX_triggered();
void on_actionEXIT_triggered();
void on_button_0_clicked();
void on_button_1_clicked();
void on_button_2_clicked();
void on_button_3_clicked();
void on_button_4_clicked();
void on_button_5_clicked();
void on_button_6_clicked();
void on_button_7_clicked();
void on_button_8_clicked();
void on_button_9_clicked();
void on_button_equal_clicked();
void on_button_dot_clicked();
void on_button_sin_clicked();
void on_button_tan_clicked();
void on_button_cos_clicked();
void on_button_radian_clicked();
void on_button_deg_clicked();
void on_button_nlog_clicked();
void on_button_log_clicked();
void on_button_pi_clicked();
void on_button_clear_2_clicked();
void on_actionBRIGHT_triggered();
void on_actionDARK_triggered();
void on_button_pm_clicked();
void on_button_expo_2_clicked();
void on_button_tanin_clicked();
```

```
        void on_button_sinin_clicked();

        void on_button_cosin_clicked();

        void on_button_sinh_clicked();

        void on_button_cosh_clicked();

        void on_button_tanh_clicked();

        void on_button_expo_3_clicked();

        void on_button_factorial_clicked();


        void on_button_clear_clicked();

        void on_actionABOUT_triggered();

        void on_actionRECORD_triggered();

    private:
        Ui::MainWindow4 *ui;
        MainWindow *standard;
        MainWindow2 *complex;
        MainWindow3 *quadratic;
        about *abt;
        record *rec;
    };

    #endif // FUNCTIONS_H
```

4) standard.h

```
#ifndef STANDARD_H
#define STANDARD_H

#include <QMainWindow>
#include "complex.h"
#include "functions.h"
#include <QApplication>
#include <about.h>
#include <record.h>

QT_BEGIN_NAMESPACE
namespace Ui { class MainWindow; }
QT_END_NAMESPACE

class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    MainWindow(QWidget *parent = nullptr);

    ~MainWindow();
```

```
private slots:
    void on_actionEXIT_triggered();

    void on_button_0_clicked();

    void on_button_1_clicked();

    void on_button_2_clicked();

    void on_button_3_clicked();

    void on_button_4_clicked();

    void on_button_5_clicked();

    void on_button_6_clicked();

    void on_button_7_clicked();

    void on_button_8_clicked();

    void on_button_9_clicked();

    void on_button_open_clicked();

    void on_button_close_clicked();

    void on_button_root_clicked();

    void on_button_divide_clicked();

    void on_button_mul_clicked();

    void on_button_minus_clicked();

    void on_button_plus_clicked();

    void on_button_equal_clicked();

    void on_button_decimal_clicked();

    void on_button_clear_2_clicked();

    void on_button_square_clicked();

    void on_button_dzero_clicked();

    void on_actionDARK_triggered();

    void on_actionBRIGHT_triggered();

    void on_actionCOMPLEX_triggered();

    void on_actionLOG_TRIGO_triggered();

    void on_button_croot_clicked();

    void on_button_cube_clicked();
```

```
    void on_button_clear_clicked();

    void on_button_fraction_clicked();

    void on_button_conversion_clicked();

    void on_button_power_clicked();

    void on_actionABOUT_triggered();

    void on_actionRECORD_triggered();

private:
    Ui::MainWindow *ui;
    MainWindow2 *complex;
    MainWindow4 *log;
    about *abt;
    record *rec;
};
#endif // STANDARD_H
```

5) record.h

```
#ifndef RECORD_H
#define RECORD_H

#include "qabstractbutton.h"
#include <QDialog>

namespace Ui {
class record;
}

class record : public QDialog
{
    Q_OBJECT

public:
    explicit record(QWidget *parent = nullptr);
    ~record();

private slots:


    void on_buttonBox_rejected();

private:
    Ui::record *ui;
};

#endif // RECORD_H
```

//Sources

1) about.cpp

```
#include "about.h"
#include "ui_about.h"

about::about(QWidget *parent) :
    QDialog(parent),
    ui(new Ui::about)
{
    ui->setupUi(this);
}

about::~about()
{
    delete ui;
}
```

2) complex.cpp

```
#include "complex.h"
#include "ui_complex.h"
#include "standard.h"
#include "functions.h"
#include <QApplication>
#include <stack>
#include<complex>
#include<cmath>
#include <QFile>
#include <QTextStream>
#include <QDebug>
using namespace std;

extern bool dark,bright;
MainWindow2::MainWindow2(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::MainWindow2)
{
    ui->setupUi(this);
    if(dark)
    {

        ui->lineEdit->setAlignment(Qt::AlignRight);
        ui->label->setAlignment(Qt::AlignRight);
        ui->button_minus-
>setIcon(QIcon(":/images/dark_icon/minus.ico"));
        ui->button_plus->setIcon(QIcon(":/images/dark_icon/plus.ico"));
        ui->button_mul-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
        ui->button_root->setIcon(QIcon(":/images/dark_icon/root.ico"));
        ui->button_divide-
>setIcon(QIcon(":/images/dark_icon/divide.ico"));
        ui->button_clear_2-
>setIcon(QIcon(":/images/dark_icon/backspace.ico"));
        ui->button_equal-
>setIcon(QIcon(":/images/dark_icon/equal.ico"));
```

```
        ui->button_decimal-
>setIcon(QIcon(":/images/dark_icon/dot.ico"));
        ui->actionCOMPLEX-
>setIcon(QIcon(":/images/dark_icon/iota.ico"));
        ui->actionSTANDARD-
>setIcon(QIcon(":/images/dark_icon/complex.ico"));
        ui->actionQUADRATIC-
>setIcon(QIcon(":/images/dark_icon/quadractic.ico"));
        ui->actionEXIT-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
        ui->actionABOUT-
>setIcon(QIcon(":/images/dark_icon/about.ico"));
        ui->actionLOG_TRIGO-
>setIcon(QIcon(":/images/dark_icon/log.ico"));
        this->setStyleSheet("background-color:rgb(32, 32, 32);color:
rgb(255,255,255);");
        ui->button_0->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_1->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_2->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_3->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_4->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_5->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_6->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_7->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_8->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_9->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_decimal->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_dzero->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_clear_2->setStyleSheet("background-color: rgb(59,
60, 63)");
        ui->button_clear->setStyleSheet("background-color: rgb(59, 60,
63)");
        ui->button_power->setStyleSheet("background-color: rgb(59, 60,
63)");
        ui->button_minus->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_plus->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_root->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_divide->setStyleSheet(("background-color: rgb(59,
60, 63)"));
        ui->button_mul->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_exp->setStyleSheet(("background-color: rgb(59, 60,
63)"));
```

```
        ui->button_log->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_root->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_iota->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_conjugate->setStyleSheet(("background-color: rgb(59,
60, 63)"));
        ui->button_normal->setStyleSheet(("background-color: rgb(59,
60, 63)"));
        ui->button_absolute->setStyleSheet(("background-color: rgb(59,
60, 63)"));
        ui->button_argument->setStyleSheet(("background-color: rgb(59,
60, 63)"));
    }
    if(bright)
    {
        ui->lineEdit->setAlignment(Qt::AlignRight);
        ui->label->setAlignment(Qt::AlignRight);
        this->setStyleSheet("background-color: rgb(243, 243,
243);color: rgb(0,0,0)");
        ui->button_minus->setIcon(QIcon(":/images/icons/minus.ico"));
        ui->button_plus->setIcon(QIcon(":/images/icons/plus.ico"));
        ui->button_mul-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
        ui->button_root->setIcon(QIcon(":/images/icons/root.ico"));
        ui->button_divide-
>setIcon(QIcon(":/images/icons/divided.ico"));
        ui->button_clear_2-
>setIcon(QIcon(":/images/icons/cancel.ico"));
        ui->button_equal->setIcon(QIcon(":/images/icons/equal.ico"));
        ui->button_decimal->setIcon(QIcon(":/images/icons/full-
stop.ico"));
        ui->actionCOMPLEX->setIcon(QIcon(":/images/icons/iota.ico"));
        ui->actionSTANDARD-
>setIcon(QIcon(":/images/icons/simple.ico"));
        ui->actionQUADRATIC-
>setIcon(QIcon(":/images/icons/quadratic.ico"));
        ui->actionEXIT-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
        ui->actionABOUT->setIcon(QIcon(":/images/icons/about.ico"));
        ui->actionLOG_TRIGO-
>setIcon(QIcon(":/images/icons/logarithm.ico"));
        ui->button_0->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_1->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_2->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_3->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_4->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_5->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_6->setStyleSheet("background-color: rgb(252, 252,
252)");
```

```
        ui->button_7->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_8->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_9->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_decimal->setStyleSheet("background-color: rgb(252,
252, 252)");
        ui->button_dzero->setStyleSheet("background-color: rgb(252,
252, 252)");
        ui->button_clear_2->setStyleSheet("background-color:rgb(246,
246, 246)");
        ui->button_minus->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_plus->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_root->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_divide->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_mul->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_exp->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_log->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_root->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_iota->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_conjugate->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_normal->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_clear->setStyleSheet("background-color: rgb(246,
246, 246)");
        ui->button_power->setStyleSheet("background-color: rgb(246,
246, 246)");
        ui->button_argument->setStyleSheet("background-color: rgb(246,
246, 246)");
}
}

MainWindow2::~MainWindow2()
{
    delete ui;
}

void MainWindow2::on_actionBRIGHT_triggered()
{
    bright=true;
    ui->lineEdit->setAlignment(Qt::AlignRight);
    ui->label->setAlignment(Qt::AlignRight);
    this->setStyleSheet("background-color: rgb(243, 243, 243);color:
rgb(0,0,0)");
    ui->button_minus->setIcon(QIcon(":/images/icons/minus.ico"));
    ui->button_plus->setIcon(QIcon(":/images/icons/plus.ico"));
```

```
    ui->button_mul-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
    ui->button_root->setIcon(QIcon(":/images/icons/root.ico"));
    ui->button_divide->setIcon(QIcon(":/images/icons/divided.ico"));
    ui->button_clear_2->setIcon(QIcon(":/images/icons/cancel.ico"));
    ui->button_equal->setIcon(QIcon(":/images/icons/equal.ico"));
    ui->button_decimal->setIcon(QIcon(":/images/icons/full-stop.ico"));
    ui->actionCOMPLEX->setIcon(QIcon(":/images/icons/iota.ico"));
    ui->actionSTANDARD->setIcon(QIcon(":/images/icons/simple.ico"));
    ui->actionQUADRATIC-
>setIcon(QIcon(":/images/icons/quadratic.ico"));
    ui->actionEXIT-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
    ui->actionABOUT->setIcon(QIcon(":/images/icons/about.ico"));
    ui->actionLOG_TRIGO-
>setIcon(QIcon(":/images/icons/logarithm.ico"));
    ui->button_0->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_1->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_2->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_3->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_4->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_5->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_6->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_7->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_8->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_9->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_decimal->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_dzero->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_clear_2->setStyleSheet("background-color:rgb(246, 246,
246)");
    ui->button_minus->setStyleSheet(("background-color:rgb(246, 246,
246)"));
    ui->button_plus->setStyleSheet(("background-color:rgb(246, 246,
246)"));
    ui->button_root->setStyleSheet(("background-color:rgb(246, 246,
246)"));
    ui->button_divide->setStyleSheet(("background-color:rgb(246, 246,
246)"));
    ui->button_mul->setStyleSheet(("background-color:rgb(246, 246,
246)"));
    ui->button_exp->setStyleSheet(("background-color:rgb(246, 246,
246)"));
    ui->button_log->setStyleSheet(("background-color:rgb(246, 246,
246)"));
    ui->button_root->setStyleSheet(("background-color:rgb(246, 246,
246)"));
```

```
    ui->button_iota->setStyleSheet(("background-color:rgb(246, 246,
246)"));
    ui->button_conjugate->setStyleSheet(("background-color:rgb(246,
246, 246)"));
    ui->button_normal->setStyleSheet(("background-color:rgb(246, 246,
246)"));
    ui->button_clear->setStyleSheet("background-color: rgb(246, 246,
246)");
    ui->button_power->setStyleSheet("background-color: rgb(246, 246,
246)");
    ui->button_absolute->setStyleSheet(("background-color: rgb(246,
246, 246)"));
    ui->button_argument->setStyleSheet("background-color: rgb(246, 246,
246)");
    dark=false;
}

void MainWindow2::on_actionDARK_triggered()
{
    dark=true;
    ui->lineEdit->setAlignment(Qt::AlignRight);
    ui->label->setAlignment(Qt::AlignRight);
    ui->button_minus->setIcon(QIcon(":/images/dark_icon/minus.ico"));
    ui->button_plus->setIcon(QIcon(":/images/dark_icon/plus.ico"));
    ui->button_mul-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
    ui->button_root->setIcon(QIcon(":/images/dark_icon/root.ico"));
    ui->button_divide->setIcon(QIcon(":/images/dark_icon/divide.ico"));
    ui->button_clear_2-
>setIcon(QIcon(":/images/dark_icon/backspace.ico"));
    ui->button_equal->setIcon(QIcon(":/images/dark_icon/equal.ico"));
    ui->button_decimal->setIcon(QIcon(":/images/dark_icon/dot.ico"));
    ui->actionCOMPLEX->setIcon(QIcon(":/images/dark_icon/iota.ico"));
    ui->actionSTANDARD-
>setIcon(QIcon(":/images/dark_icon/complex.ico"));
    ui->actionQUADRATIC-
>setIcon(QIcon(":/images/dark_icon/quadractic.ico"));
    ui->actionEXIT-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
    ui->actionABOUT->setIcon(QIcon(":/images/dark_icon/about.ico"));
    ui->actionLOG_TRIGO->setIcon(QIcon(":/images/dark_icon/log.ico"));
    this->setStyleSheet("background-color:rgb(32, 32, 32);color:
rgb(255,255,255);");
    ui->button_0->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_1->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_2->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_3->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_4->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_5->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_6->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_7->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_8->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_9->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_decimal->setStyleSheet("background-color:rgb(50, 50,
50)");
    ui->button_dzero->setStyleSheet("background-color:rgb(50, 50,
50)");
```

```
    ui->button_clear_2->setStyleSheet("background-color: rgb(59, 60,
63)");
    ui->button_clear->setStyleSheet("background-color: rgb(59, 60,
63)");
    ui->button_power->setStyleSheet("background-color: rgb(59, 60,
63)");
    ui->button_minus->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_plus->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_root->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_divide->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_mul->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_exp->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_log->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_root->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_iota->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_conjugate->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_normal->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_absolute->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_argument->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    bright=false;
}


//////////////////////////////////////////
////switching between different pages...
QString value2=" ",temp2=" ";
string num1,num2;
bool imaginary=false,sum=false,sub=false,mul=false,division=false,
npower=false,ln=false,arg_value=false,exp_bool=false,absolute=false,nor
m_bool=false;

void MainWindow2::on_actionSTANDARD_triggered()
{
    hide();
    standard = new MainWindow(this);
    standard->show();
}



void MainWindow2::on_actionLOG_TRIGO_triggered()
{
    hide();
    log=new MainWindow4(this);
    log->show();
```

```
}


void MainWindow2::on_actionABOUT_triggered()
{
    abt=new about(this);
    abt->show();
}

void MainWindow2::on_actionRECORD_triggered()
{
    rec=new record(this);
    rec->show();
}

///file handling
///
void MainWindow2::fileEdit(QString a, QString b)
{
    QFile
file("C:\\Users\\shiva\\Desktop\\sdf_project\\Calculator\\calrecord.txt
");
    if(!file.open(QIODevice::Append))
    {
        qDebug()<<"File open Failed";
    }
    else
    {
        qDebug()<<"file opened";
    }
    QTextStream out(&file);
    out<<"\n"<<a<<b<<"\n";

    file.flush();
    file.close();
}
//////////////////////////////////////////
/// \brief MainWindow2::on_actionEXIT_triggered
///quit application////
void MainWindow2::on_actionEXIT_triggered()
{
    QApplication::quit();
}

void MainWindow2::on_button_0_clicked()
{
    value2=value2+"0";
    temp2=temp2+"0";
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_1_clicked()
{
    value2=value2+"1";
    temp2=temp2+"1";
    ui->lineEdit->setText(temp2);
}
```

```
void MainWindow2::on_button_2_clicked()
{
    value2=value2+"2";
    temp2=temp2+"2";
    ui->lineEdit->setText(temp2);

}


void MainWindow2::on_button_3_clicked()
{
    value2=value2+"3";
    temp2=temp2+"3";
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_4_clicked()
{
    value2=value2+"4";
    temp2=temp2+"4";
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_5_clicked()
{
    value2=value2+"5";
    temp2=temp2+"5";
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_6_clicked()
{
    value2=value2+"6";
    temp2=temp2+"6";
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_7_clicked()
{
    value2=value2+"7";
    temp2=temp2+"7";
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_8_clicked()
{
    value2=value2+"8";
    temp2=temp2+"8";
    ui->lineEdit->setText(temp2);
}
```

```
void MainWindow2::on_button_9_clicked()
{
    value2=value2+"9";
    temp2=temp2+"9";
    ui->lineEdit->setText(temp2);
}

void MainWindow2::on_button_decimal_clicked()
{
    temp2=temp2+".";
    value2=value2+".";
    ui->lineEdit->setText(temp2);
}
void MainWindow2::on_button_dzero_clicked()
{
    temp2=temp2+"00";
    value2=value2+"00";
    ui->lineEdit->setText(temp2);
}
/////complex class function
///
///
class complexCal
{
protected:
    stack<float> values;
    float reall, img;

public:
    void Read(string str)
    {
        int i = 0, sign = 1;
        while (i < str.length() )
        {
            float val1 = 0;
            if (str[i] == '-')
            {
                sign = -1;
                i++;
            }
            if (isdigit(str[i]))
            {
                while (i < str.length() && isdigit(str[i]))
                {
                    val1 = (val1 * 10) + (str[i] - '0');
                    i++;
                }
                if (i < str.length() && str[i] == '.')
                {
                    i++;
                    float decimal = 0.1;
                    while (i < str.length() && isdigit(str[i]))
                    {
                        val1 += decimal * (str[i] - '0');
                        decimal *= 0.1;
                        i++;
                    }
```

```
            }
            values.push(val1 * sign);
        }
        else
        {
            i++;
        }
    }
    img = values.top();
    values.pop();
    reall = values.top();
    values.pop();
}
complexCal operator+(complexCal other)
{
    complexCal temp;
    temp.reall=reall+other.reall;
    temp.img=img+other.img;
    return temp;
}
complexCal operator-(complexCal other)
{
    complexCal temp;
    temp.reall=reall-other.reall;
    temp.img=img-other.img;
    return temp;
}
complexCal operator*(complexCal other)
{
    complexCal temp;
    temp.reall=(reall*other.reall)+(-1*(img*other.img));
    temp.img=(img*other.reall)+(reall*other.img);
     return temp;
}
complexCal operator~()
{
    complexCal temp;
    temp.reall=reall;
    temp.img=-1*img;
    return temp;
}
complexCal operator /(complexCal other)
{
    complexCal obj,temp,num,deno;
    obj= ~other;
    num=(*this)*obj;
    deno=other * obj;
    temp.reall=num.reall/deno.reall;
    temp.img=num.img/deno.reall;
    return temp;
}
complexCal root()
{
    complexCal temp;
    complex <float>in((*this).reall,(*this).img);
    complex<float> ans;
    ans=sqrt(in);
    temp.reall=real(ans);
```

```
    temp.img=imag(ans);
    return temp;
}
complexCal power(int n)
{
    complexCal temp;
    complex <float>in((*this).reall,(*this).img);
    complex<float> ans;
    ans=pow(in,n);
    temp.reall=real(ans);
    temp.img=imag(ans);
    return temp;
}
complexCal log_value()
{
    complexCal temp;
    complex <float>in((*this).reall,(*this).img);
    complex<float> ans;
    ans=log(in);
    temp.reall=real(ans);
    temp.img=imag(ans);
    return temp;
}
complexCal argument()
{
    complexCal temp;
    complex <float>in((*this).reall,(*this).img);
    complex<float> ans;
    ans=arg(in);
    temp.reall=real(ans);
    temp.img=imag(ans);
    return temp;
}
complexCal exp_value()
{
    complexCal temp;
    complex <float>in((*this).reall,(*this).img);
    complex<float> ans;
    ans=exp(in);
    temp.reall=real(ans);
    temp.img=imag(ans);
    return temp;
}
complexCal normal()
{
    complexCal temp;
    complex <float>in((*this).reall,(*this).img);
    complex<float> ans;
    ans=norm(in);
    temp.reall=real(ans);
    temp.img=imag(ans);
    return temp;
}
complexCal absolute()
{
    complexCal temp;
    complex <float>in((*this).reall,(*this).img);
    complex<float> ans;
```

```
        ans=abs(in);
        temp.reall=real(ans);
        temp.img=imag(ans);
        return temp;
    }
    QString display()
    {
        QString str;
        if(img>=0)
        str=QString::number(reall)+'+'+QString::number(img)+'i';
        else if(img<0)
        str=QString::number(reall)+QString::number(img)+'i';
        return str;
    }

}c1,c2,temp;

void MainWindow2::on_button_iota_clicked()
{
    imaginary=true;
    temp2=temp2+'i';
    value2=value2+'i';
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_plus_clicked()
{
    if(imaginary)
    {
        sum=true;
        num1=value2.toStdString();
        value2=" ";
    }
    else
    value2=value2+'+';
    temp2+='+';
    imaginary=false;
    ui->lineEdit->setText(temp2);
}

void MainWindow2::on_button_minus_clicked()
{
    if(imaginary)
    {
        sub=true;
        num1=value2.toStdString();
        value2=" ";
    }
    else
    value2=value2+'-';
    temp2+='-';
    imaginary=false;
    ui->lineEdit->setText(temp2);
}
void MainWindow2::on_button_mul_clicked()
{
    if(imaginary)
```

```
    {
        mul=true;
        num1=value2.toStdString();
        value2=" ";
    }
    else
    value2=value2+'*';
    temp2+='*';
    imaginary=false;
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_clear_2_clicked()
{
    value2=value2.remove(value2.length()-1,1);
    temp2=temp2.remove(temp2.length()-1,1);
    ui->lineEdit->setText(temp2);
}

void MainWindow2::on_button_clear_clicked()
{
    value2=" ";
    temp2=" ";
    ui->lineEdit->setText(temp2);
    imaginary=false;

sum=false,sub=false,mul=false,division=false,npower=false,ln=false,arg_
value=false;
}
void MainWindow2::on_button_divide_clicked()
{
    if(imaginary)
    {
        mul=true;
        num1=value2.toStdString();
        value2=" ";
    }
    else
    value2=value2+'/';
    temp2+='/';
    imaginary=false;
    ui->lineEdit->setText(temp2);
}
void MainWindow2::on_button_conjugate_clicked()
{
    c1.Read(value2.toStdString());
    temp=~c1;
    temp2=temp2+"=";
    ui->lineEdit->setText(temp.display());
    fileEdit(temp2,temp.display());
    temp2=temp.display();
    imaginary=true;
}


void MainWindow2::on_button_root_clicked()
{
```

```
    c1.Read(value2.toStdString());
    temp=c1.root();
    temp2=temp2+"=";
    ui->lineEdit->setText(temp.display());
    fileEdit(temp2,temp.display());
    temp2=temp.display();
    value2=temp.display();
    imaginary=true;
}


void MainWindow2::on_button_power_clicked()
{
    temp2=temp2+"^";
    if(imaginary)
    {
        npower=true;
        num1=value2.toStdString();
        value2=" ";
    }
    ui->lineEdit->setText(temp2);
    imaginary=false;
}

void MainWindow2::on_button_log_clicked()
{
    temp2=temp2+"log(";
    ln=true;
    ui->lineEdit->setText(temp2);
    num1=" ";
}

void MainWindow2::on_button_argument_clicked()
{
    temp2=temp2+"arg(";
    arg_value=true;
    ui->lineEdit->setText(temp2);
}
void MainWindow2::on_button_absolute_clicked()
{
    temp2=temp2+"abs(";
    absolute=true;
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_normal_clicked()
{
    temp2=temp2+"norm(";
    norm_bool=true;
    ui->lineEdit->setText(temp2);
}


void MainWindow2::on_button_exp_clicked()
{
    temp2=temp2+"exp(";
    exp_bool=true;
```

```
    ui->lineEdit->setText(temp2);
}

void MainWindow2::on_button_equal_clicked()
{
    temp2=temp2+")=";
    num2=value2.toStdString();
    if(sum)
    {
        c1.Read(num1);
        c2.Read(num2);
        temp=c1+c2;

    }
    if(sub)
    {
        c1.Read(num1);
        c2.Read(num2);
        temp=c1-c2;

    }
    if(mul)
    {
        c1.Read(num1);
        c2.Read(num2);
        temp=c1*c2;

    }
    if(division)
    {
        c1.Read(num1);
        c2.Read(num2);
        temp=c1/c2;

    }
    if(npower)
    {
        c1.Read(num1);
        int val=value2.toInt();
        temp=c1.power(val);

    }
    if(ln)
    {
        c2.Read(num2);
        temp=c2.log_value();
    }
    if(arg_value)
    {
        c2.Read(num2);
        temp=c2.argument();
    }
    if(exp_bool)
    {
        c2.Read(num2);
        temp=c2.exp_value();
    }
    if(absolute)
```

```
    {
        c2.Read(num2);
        temp=c2.absolute();
    }
    if(norm_bool)
    {
        c2.Read(num2);
        temp=c2.normal();
    }

    fileEdit(temp2,temp.display());
    ui->lineEdit->setText(temp.display());
    temp2=temp.display();
    value2=temp.display();
    imaginary=true;

sum=false,sub=false,mul=false,division=false,npower=false,ln=false,arg_
value=false,exp_bool=false,absolute=false,norm_bool=false;
}
```

3) functions.cpp

```
#include "functions.h"
#include "ui_functions.h"
#include "standard.h"
#include "complex.h"
#include <math.h>
#include <string.h>
#include <QApplication>
#include <QFile>
#include <QTextStream>
#include <QDebug>
using namespace std;
extern bool dark,bright;

MainWindow4::MainWindow4(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::MainWindow4)
{
    ui->setupUi(this);
    if(dark)
    {
        ui->lineEdit->setAlignment(Qt::AlignRight);
        ui->label->setAlignment(Qt::AlignRight);
        ui->button_equal-
>setIcon(QIcon(":/images/dark_icon/equal.ico"));
        ui->button_pi->setIcon(QIcon(":/images/dark_icon/pi.ico"));
        ui->button_clear_2-
>setIcon(QIcon(":/images/dark_icon/backspace.ico"));
        ui->actionCOMPLEX-
>setIcon(QIcon(":/images/dark_icon/iota.ico"));
        ui->actionSTANDARD-
>setIcon(QIcon(":/images/dark_icon/complex.ico"));
        ui->actionQUADRATIC-
>setIcon(QIcon(":/images/dark_icon/quadractic.ico"));
```

```
        ui->actionEXIT-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
        ui->actionABOUT-
>setIcon(QIcon(":/images/dark_icon/about.ico"));
        ui->actionLOG_TRIGO-
>setIcon(QIcon(":/images/dark_icon/log.ico"));
        ui->lineEdit->setStyleSheet("border:none; background-
color:rgb(32, 32, 32);font-size:30px");
        ui->button_dot->setIcon(QIcon(":/images/dark_icon/dot.ico"));
        this->setStyleSheet("background-color:rgb(32, 32, 32);color:
rgb(255,255,255);");
        ui->button_0->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_1->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_2->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_3->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_4->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_5->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_6->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_7->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_8->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_9->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_dot->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_pm->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_clear_2->setStyleSheet(" background-color: rgb(59,
60, 63);font-size:15px");
        ui->button_pi->setStyleSheet(" background-color: rgb(59, 60,
63;font-size:15px)");
        ui->button_tanin->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_sinin->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_cosin->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_sinh->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_cosh->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_tanh->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_expo_2->setStyleSheet(" background-color: rgb(59,
60, 63);font-size:15px");
        ui->button_expo_3->setStyleSheet(" background-color: rgb(59,
60, 63);font-size:15px");
        ui->button_clear->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
```

```
        ui->button_radian->setStyleSheet(" background-color: rgb(59,
60, 63);font-size:15px");
        ui->button_deg->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_tan->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_cos->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_sin->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_factorial->setStyleSheet(" background-color: rgb(59,
60, 63);font-size:15px");
        ui->button_log->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
        ui->button_nlog->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    }
    if(bright){
        ui->lineEdit->setAlignment(Qt::AlignRight);
        ui->label->setAlignment(Qt::AlignRight);
        ui->button_clear_2-
>setIcon(QIcon(":/images/icons/cancel.ico"));
        ui->button_equal->setIcon(QIcon(":/images/icons/equal.ico"));
        ui->button_dot->setIcon(QIcon(":/images/icons/full-stop.ico"));
        ui->actionCOMPLEX->setIcon(QIcon(":/images/icons/iota.ico"));
        ui->actionSTANDARD-
>setIcon(QIcon(":/images/icons/simple.ico"));
        ui->actionQUADRATIC-
>setIcon(QIcon(":/images/icons/quadratic.ico"));
        ui->actionEXIT-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
        ui->actionABOUT->setIcon(QIcon(":/images/icons/about.ico"));
        ui->button_pi->setIcon(QIcon(":/images/icons/pi.ico"));
        ui->actionLOG_TRIGO-
>setIcon(QIcon(":/images/icons/logarithm.ico"));
        ui->lineEdit->setStyleSheet("border:none; background-
color:rgb(243, 243, 243);font-size:30px");
        this->setStyleSheet("background-color: rgb(243, 243,
243);color: rgb(0,0,0)");
        ui->button_0->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_1->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_2->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_3->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_4->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_5->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_6->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_7->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_8->setStyleSheet("background-color: rgb(252, 252,
252)");
```

```
        ui->button_9->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_dot->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_pm->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_clear_2->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_pi->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
        ui->button_tanin->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_sinin->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_cosin->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_sinh->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_cosh->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_tanh->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_expo_2->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_expo_3->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_clear->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_radian->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
        ui->button_deg->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
        ui->button_tan->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
        ui->button_cos->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
        ui->button_sin->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
        ui->button_factorial->setStyleSheet(" background-color:
rgb(252, 252, 252);font-size:15px");
        ui->button_log->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
        ui->button_nlog->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
    }

}

MainWindow4::~MainWindow4()
{
    delete ui;
}

void MainWindow4::on_actionBRIGHT_triggered()
{
    bright=true;
    ui->lineEdit->setAlignment(Qt::AlignRight);
    ui->label->setAlignment(Qt::AlignRight);
```

```
    ui->button_clear_2->setIcon(QIcon(":/images/icons/cancel.ico"));
    ui->button_equal->setIcon(QIcon(":/images/icons/equal.ico"));
    ui->button_dot->setIcon(QIcon(":/images/icons/full-stop.ico"));
    ui->actionCOMPLEX->setIcon(QIcon(":/images/icons/iota.ico"));
    ui->actionSTANDARD->setIcon(QIcon(":/images/icons/simple.ico"));
    ui->actionQUADRATIC-
>setIcon(QIcon(":/images/icons/quadratic.ico"));
    ui->actionEXIT-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
    ui->actionABOUT->setIcon(QIcon(":/images/icons/about.ico"));
    ui->button_pi->setIcon(QIcon(":/images/icons/pi.ico"));
    ui->actionLOG_TRIGO-
>setIcon(QIcon(":/images/icons/logarithm.ico"));
    ui->lineEdit->setStyleSheet("border:none; background-color:rgb(243,
243, 243);font-size:30px");
    this->setStyleSheet("background-color: rgb(243, 243, 243);color:
rgb(0,0,0)");
    ui->button_0->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_1->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_2->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_3->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_4->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_5->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_6->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_7->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_8->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_9->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_dot->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_pm->setStyleSheet("background-color: rgb(252, 252,
252)");
    ui->button_clear_2->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_pi->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_tanin->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_sinin->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_cosin->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_sinh->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_cosh->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_tanh->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
```

```
    ui->button_expo_2->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_expo_3->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_clear->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_radian->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_deg->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_tan->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_cos->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_sin->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_factorial->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px");
    ui->button_log->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    ui->button_nlog->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px");
    dark=false;
}


void MainWindow4::on_actionDARK_triggered()
{
    dark=true;
    ui->lineEdit->setAlignment(Qt::AlignRight);
    ui->label->setAlignment(Qt::AlignRight);
    ui->button_equal->setIcon(QIcon(":/images/dark_icon/equal.ico"));
    ui->button_pi->setIcon(QIcon(":/images/dark_icon/pi.ico"));
    ui->button_clear_2-
>setIcon(QIcon(":/images/dark_icon/backspace.ico"));
    ui->actionCOMPLEX->setIcon(QIcon(":/images/dark_icon/iota.ico"));
    ui->actionSTANDARD-
>setIcon(QIcon(":/images/dark_icon/complex.ico"));
    ui->actionQUADRATIC-
>setIcon(QIcon(":/images/dark_icon/quadractic.ico"));
    ui->actionEXIT-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
    ui->actionABOUT->setIcon(QIcon(":/images/dark_icon/about.ico"));
    ui->actionLOG_TRIGO->setIcon(QIcon(":/images/dark_icon/log.ico"));
    ui->lineEdit->setStyleSheet("border:none; background-color:rgb(32,
32, 32);font-size:30px");
    ui->button_dot->setIcon(QIcon(":/images/dark_icon/dot.ico"));
    this->setStyleSheet("background-color:rgb(32, 32, 32);color:
rgb(255,255,255);");
    ui->button_0->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_1->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_2->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_3->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_4->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_5->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_6->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_7->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_8->setStyleSheet("background-color:rgb(50, 50, 50)");
```

```
    ui->button_9->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_dot->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_pm->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_clear_2->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_pi->setStyleSheet(" background-color: rgb(59, 60,
63;font-size:15px)");
    ui->button_tanin->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_sinin->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_cosin->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_sinh->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_cosh->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_tanh->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_expo_2->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_expo_3->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_clear->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_radian->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_deg->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_tan->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_cos->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_sin->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_factorial->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_log->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    ui->button_nlog->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px");
    bright=false;
}

//////////////////////////////////////////
////switching between different pages...
void MainWindow4::on_actionSTANDARD_triggered()
{
    hide();
    standard= new MainWindow(this);
    standard->show();
}


void MainWindow4::on_actionCOMPLEX_triggered()
{
    hide();
    complex = new MainWindow2(this);
```

```
    complex->show();
}

void MainWindow4::on_actionABOUT_triggered()
{
    abt=new about(this);
    abt->show();
}


void MainWindow4::on_actionRECORD_triggered()
{

    rec=new record(this);
    rec->show();

}
////////////////////////////////////////////
/// \brief MainWindow2::on_actionEXIT_triggered
///quit application////

void MainWindow4::on_actionEXIT_triggered()
{
    QApplication::quit();
}


QString value4=" ",temp4=" ";
double tempValue;
bool tanbool=false,cosbool=false,fact=false,
sinbool=false,logbool=false,inverse=false,lnbool=false,
hyperbolic=false,trigo=false,exponent=false,conversion=true;



//////////////////
/// \brief The function_class class
///class for log and trigo funtions.....
///

class function_class{
private:
    double total;
public:
    function_class(QString value,bool conversion)
    {
        if(conversion)
        total=value.toDouble();
        else if(!conversion)
        total=value.toDouble()*0.01745329;
    }
    double logfuntion();
    double lnfuntion();
    double tanFuntion();
    double sinFuntion();
    double cosFuntion();
    double tanInverse();
    double cosInverse();
```

```
    double sinInverse();
    double tanhyper();
    double sinhyper();
    double coshyper();
    double exponential();
    double factorial();
    void reset();
};
double function_class::lnfuntion()
{
    total=log(total);
    return total;
}
double function_class::logfuntion()
{
    total=log10(total);
    return total;
}
double function_class::tanFuntion()
{
    total=tan(total);
    return total;
}
double function_class:: sinFuntion()
{
    total=sin(total);
    return total;
}
double function_class::cosFuntion()
{
    total=cos(total);
    return total;
}
double function_class::cosInverse()
{
    total=acos(total);
    return total;
}
double function_class::sinInverse()
{
    total=asin(total);
    return total;
}
double function_class::tanInverse()
{
    total=atan(total);
    return total;
}
double function_class::tanhyper()
{
    total=tanh(total);
    return total;
}
double function_class::sinhyper()
{
    total=sinh(total);
    return total;
}
```

```
double function_class::coshyper()
{
    total=cosh(total);
    return total;
}
double function_class::exponential()
{
    total=exp(total);
    return total;
}
double function_class::factorial()
{
    int i;
    double factorial=1;
    if(total==1||total==0)
    {
        return 1;
    }
    for(i=1;i<=total;i++)
    {
        factorial*=i;
    }
    return factorial;
}
void function_class::reset()
{
    total=0;
}

////////////////////////////////
///////button functionality code....


void MainWindow4::on_button_0_clicked()
{
    value4=value4+"0";
    temp4=temp4+"0";
    ui->lineEdit->setText(temp4);
}

void MainWindow4::on_button_1_clicked()
{
    value4=value4+"1";
    temp4=temp4+"1";
    ui->lineEdit->setText(temp4);
}


void MainWindow4::on_button_2_clicked()
{
    value4=value4+"2";
    temp4=temp4+"2";
    ui->lineEdit->setText(temp4);

}


void MainWindow4::on_button_3_clicked()
```

```
{
    value4=value4+"3";
    temp4=temp4+"3";
    ui->lineEdit->setText(temp4);
}


void MainWindow4::on_button_4_clicked()
{
    value4=value4+"4";
    temp4=temp4+"4";
    ui->lineEdit->setText(temp4);
}


void MainWindow4::on_button_5_clicked()
{
    value4=value4+"5";
    temp4=temp4+"5";
    ui->lineEdit->setText(temp4);
}


void MainWindow4::on_button_6_clicked()
{
    value4=value4+"6";
    temp4=temp4+"6";
    ui->lineEdit->setText(temp4);
}


void MainWindow4::on_button_7_clicked()
{
    value4=value4+"7";
    temp4=temp4+"7";
    ui->lineEdit->setText(temp4);
}


void MainWindow4::on_button_8_clicked()
{
    value4=value4+"8";
    temp4=temp4+"8";
    ui->lineEdit->setText(temp4);
}


void MainWindow4::on_button_9_clicked()
{
    value4=value4+"9";
    temp4=temp4+"9";
    ui->lineEdit->setText(temp4);
}

void MainWindow4::on_button_dot_clicked()
{
    value4=value4+".";
    temp4=temp4+".";
```

```
    ui->lineEdit->setText(temp4);
}

void MainWindow4::on_button_sin_clicked()
{
    temp4=temp4+"sin(";
    ui->lineEdit->setText(temp4);
    sinbool=true;
    trigo=true;
}


void MainWindow4::on_button_tan_clicked()
{
    temp4=temp4+"tan(";
    ui->lineEdit->setText(temp4);
    tanbool=true;
    trigo=true;
}


void MainWindow4::on_button_cos_clicked()
{
    temp4=temp4+"cos(";
    ui->lineEdit->setText(temp4);
    cosbool=true;
    trigo=true;
}


void MainWindow4::on_button_radian_clicked()
{
    conversion=true;
    ui->button_radian->setStyleSheet(" background-color: rgb(38, 186,
217)");
    if(dark)
    {
        ui->button_deg->setStyleSheet(" background-color: rgb(59, 60,
63);font-size:15px;");
    }
    else if(bright)
    {
        ui->button_deg->setStyleSheet(" background-color: rgb(252, 252,
252);font-size:15px;");
    }
}


void MainWindow4::on_button_deg_clicked()
{
    conversion=false;
    ui->button_deg->setStyleSheet(" background-color: rgb(38, 186,
217)");
    if(dark)
    {
        ui->button_radian->setStyleSheet(" background-color: rgb(59,
60, 63);font-size:15px;");
    }
```

```
    else if(bright)
    {
        ui->button_radian->setStyleSheet(" background-color: rgb(252,
252, 252);font-size:15px;");
    }
}


void MainWindow4::on_button_nlog_clicked()
{
    temp4=temp4+"ln(";
    ui->lineEdit->setText(temp4);
    lnbool=true;
}


void MainWindow4::on_button_log_clicked()
{
    temp4=temp4+"log(";
    ui->lineEdit->setText(temp4);
    logbool=true;
}


void MainWindow4::on_button_pi_clicked()
{
    if(value4==" ")
    {
        temp4=temp4+"π";
        value4+="3.14159265358979323846264 3";
    }
    else
    {
        temp4=temp4+"π";
        tempValue=value4.toDouble();
        tempValue*=3.14159265358979323846264 3;
        value4=QString::number(tempValue);
    }
    ui->lineEdit->setText(temp4);
    tempValue=0;
}


void MainWindow4::on_button_pm_clicked()
{
    tempValue=value4.toDouble();
    tempValue=-tempValue;
    temp4="-"+temp4;
    ui->lineEdit->setText(temp4);
    value4=QString::number(tempValue);
    tempValue=0;
}


void MainWindow4::on_button_expo_2_clicked()
{
    if(value4==" ")
    {
```

```
        temp4=temp4+"e";
        value4+="2.718281828459045235360";
    }
    else
    {
        temp4=temp4+"e";
        tempValue=value4.toDouble();
        tempValue*=2.718281828459045235360;
        value4=QString::number(tempValue);
    }
    ui->lineEdit->setText(temp4);
    tempValue=0;
}


void MainWindow4::on_button_tanin_clicked()
{

    temp4=temp4+"tan-1(";
    ui->lineEdit->setText(temp4);
    tanbool=true;
    inverse=true;
}


void MainWindow4::on_button_sinin_clicked()
{
    temp4=temp4+"sin-1(";
    ui->lineEdit->setText(temp4);
    sinbool=true;
    inverse=true;
}


void MainWindow4::on_button_cosin_clicked()
{
    temp4=temp4+"cos-1(";
    ui->lineEdit->setText(temp4);
    cosbool=true;
    inverse=true;
}


void MainWindow4::on_button_sinh_clicked()
{
    temp4=temp4+"sinh(";
    ui->lineEdit->setText(temp4);
    sinbool=true;
    hyperbolic=true;
}


void MainWindow4::on_button_cosh_clicked()
{

    temp4=temp4+"cosh(";
    ui->lineEdit->setText(temp4);
    cosbool=true;
```

```
    hyperbolic=true;
}


void MainWindow4::on_button_tanh_clicked()
{

    temp4=temp4+"tanh(";
    ui->lineEdit->setText(temp4);
    tanbool=true;
    hyperbolic=true;
}

void MainWindow4::on_button_expo_3_clicked()
{
    exponent=true;
    temp4=temp4+"exp(";
    ui->lineEdit->setText(temp4);
}
void MainWindow4::on_button_factorial_clicked()
{
    fact=true;
    temp4=temp4+"!";
    ui->lineEdit->setText(temp4);
}

void MainWindow4::on_button_equal_clicked()
{
    double tempNum=0;
    temp4=temp4+")=";
    ui->label->setText(temp4);
    function_class c(value4,conversion);
    if(logbool)
        tempNum=c.logfuntion();
    if(lnbool)
        tempNum=c.lnfuntion();
    if(exponent)
        tempNum=c.exponential();
    if(fact)
        tempNum=c.factorial();
     if(inverse)
        {
            if(tanbool)
                tempNum=c.tanInverse();
            if(sinbool)
                tempNum=c.sinInverse();
            if(cosbool)
                tempNum=c.cosInverse();
        }
      if(hyperbolic)
        {
            if(tanbool)
                tempNum=c.tanFuntion();
            if(sinbool)
                tempNum=c.sinFuntion();
            if(cosbool)
                tempNum=c.cosFuntion();
        }
```

```
      if(trigo)
      {
          if(tanbool)
              tempNum=c.tanFuntion();
          if(sinbool)
              tempNum=c.sinFuntion();
          if(cosbool)
              tempNum=c.cosFuntion();
      }

    c.reset();
    ui->lineEdit->setText(QString::number(tempNum));
    ///file handling
    ///
    QFile
file("C:\\Users\\shiva\\Desktop\\sdf_project\\Calculator\\calrecord.txt
");
    if(!file.open(QIODevice::Append))
    {
        qDebug()<<"File open Failed";
    }
    else
    {
        qDebug()<<"file opened";
    }
    QTextStream out(&file);
        out<<"\n"<<temp4<<tempNum<<"\n";

    file.flush();
    file.close();
    value4=" ";
    temp4=" ";

tanbool=false,cosbool=false,sinbool=false,logbool=false,inverse=false,l
nbool=false,hyperbolic=false,trigo=false,exponent=false,fact=false;
}

void MainWindow4::on_button_clear_2_clicked()
{
    value4=value4.remove(value4.length()-1,1);
    temp4=temp4.remove(temp4.length()-1,1);
    ui->lineEdit->setText(temp4);
}
void MainWindow4::on_button_clear_clicked()
{

    temp4=" ";
    value4=" ";
    ui->lineEdit->setText(temp4);
    ui->label->setText(temp4);

tanbool=false,cosbool=false,sinbool=false,logbool=false,inverse=false,l
nbool=false,hyperbolic=false,trigo=false,exponent=false,fact=false;
}
```

4) main.cpp

```
#include "standard.h"
#include <QApplication>

bool dark=true,bright=false;

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    MainWindow w;
    a.setWindowIcon(QIcon(":/images/icons/calculator.ico"));
    a.setApplicationDisplayName("Calculator");
    w.show();
    return a.exec();
}
```

5) standard.cpp

```
#include "standard.h"
#include "ui_standard.h"
#include <string.h>
#include <stack>
#include <QApplication>
#include <math.h>
#include <QFile>
#include <QTextStream>
#include <QDebug>

using namespace std;
extern bool dark,bright;
MainWindow::MainWindow(QWidget *parent)
    : QMainWindow(parent)
    , ui(new Ui::MainWindow)
{
    ui->setupUi(this);
    if(dark)
    {

        ui->lineEdit->setAlignment(Qt::AlignRight);
        ui->label->setAlignment(Qt::AlignRight);
        ui->button_minus-
>setIcon(QIcon(":/images/dark_icon/minus.ico"));
        ui->button_plus->setIcon(QIcon(":/images/dark_icon/plus.ico"));
        ui->button_mul-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
        ui->button_root->setIcon(QIcon(":/images/dark_icon/root.ico"));
        ui->button_divide-
>setIcon(QIcon(":/images/dark_icon/divide.ico"));
        ui->button_clear_2-
>setIcon(QIcon(":/images/dark_icon/backspace.ico"));
        ui->button_equal-
>setIcon(QIcon(":/images/dark_icon/equal.ico"));
```

```
        ui->button_decimal-
>setIcon(QIcon(":/images/dark_icon/dot.ico"));
        ui->actionCOMPLEX-
>setIcon(QIcon(":/images/dark_icon/iota.ico"));
        ui->actionSTANDARD-
>setIcon(QIcon(":/images/dark_icon/complex.ico"));
        ui->actionQUADRATIC-
>setIcon(QIcon(":/images/dark_icon/quadractic.ico"));
        ui->actionEXIT-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
        ui->actionABOUT-
>setIcon(QIcon(":/images/dark_icon/about.ico"));
        ui->actionLOG_TRIGO-
>setIcon(QIcon(":/images/dark_icon/log.ico"));
        this->setStyleSheet("background-color:rgb(32, 32, 32);color:
rgb(255,255,255);");
        ui->button_0->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_1->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_2->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_3->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_4->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_5->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_6->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_7->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_8->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_9->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_decimal->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_dzero->setStyleSheet("background-color:rgb(50, 50,
50)");
        ui->button_clear_2->setStyleSheet("background-color: rgb(59,
60, 63)");
        ui->button_clear->setStyleSheet("background-color: rgb(59, 60,
63)");
        ui->button_power->setStyleSheet("background-color: rgb(59, 60,
63)");
        ui->button_minus->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_plus->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_root->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_divide->setStyleSheet(("background-color: rgb(59,
60, 63)"));
        ui->button_mul->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_croot->setStyleSheet(("background-color: rgb(59, 60,
63)"));
```

```
        ui->button_square->setStyleSheet(("background-color: rgb(59,
60, 63)"));
        ui->button_root->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_open->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_close->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_cube->setStyleSheet(("background-color: rgb(59, 60,
63)"));
        ui->button_fraction->setStyleSheet(("background-color: rgb(59,
60, 63)"));
        ui->button_conversion->setStyleSheet(("background-color:
rgb(59, 60, 63)"));
    }
    if(bright)
    {
        ui->lineEdit->setAlignment(Qt::AlignRight);
        ui->label->setAlignment(Qt::AlignRight);
        this->setStyleSheet("background-color: rgb(243, 243,
243);color: rgb(0,0,0)");
        ui->button_minus->setIcon(QIcon(":/images/icons/minus.ico"));
        ui->button_plus->setIcon(QIcon(":/images/icons/plus.ico"));
        ui->button_mul-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
        ui->button_root->setIcon(QIcon(":/images/icons/root.ico"));
        ui->button_divide-
>setIcon(QIcon(":/images/icons/divided.ico"));
        ui->button_clear_2-
>setIcon(QIcon(":/images/icons/cancel.ico"));
        ui->button_equal->setIcon(QIcon(":/images/icons/equal.ico"));
        ui->button_decimal->setIcon(QIcon(":/images/icons/full-
stop.ico"));
        ui->actionCOMPLEX->setIcon(QIcon(":/images/icons/iota.ico"));
        ui->actionSTANDARD-
>setIcon(QIcon(":/images/icons/simple.ico"));
        ui->actionQUADRATIC-
>setIcon(QIcon(":/images/icons/quadratic.ico"));
        ui->actionEXIT-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
        ui->actionABOUT->setIcon(QIcon(":/images/icons/about.ico"));
        ui->actionLOG_TRIGO-
>setIcon(QIcon(":/images/icons/logarithm.ico"));
        ui->button_0->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_1->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_2->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_3->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_4->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_5->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_6->setStyleSheet("background-color: rgb(252, 252,
252)");
```

```
        ui->button_7->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_8->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_9->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_decimal->setStyleSheet("background-color: rgb(252,
252, 252)");
        ui->button_dzero->setStyleSheet("background-color: rgb(252,
252, 252)");
        ui->button_clear_2->setStyleSheet("background-color:rgb(246,
246, 246)");
        ui->button_minus->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_plus->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_root->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_divide->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_mul->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_croot->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_square->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_root->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_open->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_close->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_cube->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_clear->setStyleSheet("background-color: rgb(246,
246, 246)");
        ui->button_power->setStyleSheet("background-color: rgb(246,
246, 246)");
        ui->button_conversion->setStyleSheet("background-color:
rgb(246, 246, 246)");
}
}

MainWindow::~MainWindow()
{
    delete ui;
}


////////////////////////////////////////////
/// \brief MainWindow::on_actionDARK_triggered
///dark theme ........///
void MainWindow::on_actionDARK_triggered()
{
    dark=true;
    ui->lineEdit->setAlignment(Qt::AlignRight);
    ui->label->setAlignment(Qt::AlignRight);
    ui->button_minus->setIcon(QIcon(":/images/dark_icon/minus.ico"));
```

```
    ui->button_plus->setIcon(QIcon(":/images/dark_icon/plus.ico"));
    ui->button_mul-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
    ui->button_root->setIcon(QIcon(":/images/dark_icon/root.ico"));
    ui->button_divide->setIcon(QIcon(":/images/dark_icon/divide.ico"));
    ui->button_clear_2-
>setIcon(QIcon(":/images/dark_icon/backspace.ico"));
    ui->button_equal->setIcon(QIcon(":/images/dark_icon/equal.ico"));
    ui->button_decimal->setIcon(QIcon(":/images/dark_icon/dot.ico"));
    ui->actionCOMPLEX->setIcon(QIcon(":/images/dark_icon/iota.ico"));
    ui->actionSTANDARD-
>setIcon(QIcon(":/images/dark_icon/complex.ico"));
    ui->actionQUADRATIC-
>setIcon(QIcon(":/images/dark_icon/quadractic.ico"));
    ui->actionEXIT-
>setIcon(QIcon(":/images/dark_icon/multiplication.ico"));
    ui->actionABOUT->setIcon(QIcon(":/images/dark_icon/about.ico"));
    ui->actionLOG_TRIGO->setIcon(QIcon(":/images/dark_icon/log.ico"));
    this->setStyleSheet("background-color:rgb(32, 32, 32);color:
rgb(255,255,255);");
    ui->button_0->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_1->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_2->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_3->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_4->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_5->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_6->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_7->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_8->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_9->setStyleSheet("background-color:rgb(50, 50, 50)");
    ui->button_decimal->setStyleSheet("background-color:rgb(50, 50,
50)");
    ui->button_dzero->setStyleSheet("background-color:rgb(50, 50,
50)");
    ui->button_clear_2->setStyleSheet("background-color: rgb(59, 60,
63)");
    ui->button_minus->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_plus->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_root->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_divide->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_mul->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_clear->setStyleSheet("background-color: rgb(59, 60,
63)");
    ui->button_power->setStyleSheet("background-color: rgb(59, 60,
63)");
    ui->button_croot->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_square->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_root->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_open->setStyleSheet(("background-color: rgb(59, 60,
63)"));
```

```
    ui->button_close->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_cube->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_fraction->setStyleSheet(("background-color: rgb(59, 60,
63)"));
    ui->button_conversion->setStyleSheet(("background-color: rgb(59,
60, 63)"));
   bright=false;
}

////////////////////////////////////////////////////////////////////////
/
/// \brief MainWindow::on_actionBRIGHT_triggered
///stylesheet for bright theme.....................
void MainWindow::on_actionBRIGHT_triggered()
{
    bright=true;
        ui->lineEdit->setAlignment(Qt::AlignRight);
        ui->label->setAlignment(Qt::AlignRight);
        this->setStyleSheet("background-color: rgb(243, 243,
243);color: rgb(0,0,0)");
        ui->button_minus->setIcon(QIcon(":/images/icons/minus.ico"));
        ui->button_plus->setIcon(QIcon(":/images/icons/plus.ico"));
        ui->button_mul-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
        ui->button_root->setIcon(QIcon(":/images/icons/root.ico"));
        ui->button_divide-
>setIcon(QIcon(":/images/icons/divided.ico"));
        ui->button_clear_2-
>setIcon(QIcon(":/images/icons/cancel.ico"));
        ui->button_equal->setIcon(QIcon(":/images/icons/equal.ico"));
        ui->button_decimal->setIcon(QIcon(":/images/icons/full-
stop.ico"));
        ui->actionCOMPLEX->setIcon(QIcon(":/images/icons/iota.ico"));
        ui->actionSTANDARD-
>setIcon(QIcon(":/images/icons/simple.ico"));
        ui->actionQUADRATIC-
>setIcon(QIcon(":/images/icons/quadratic.ico"));
        ui->actionEXIT-
>setIcon(QIcon(":/images/icons/multiplication.ico"));
        ui->actionABOUT->setIcon(QIcon(":/images/icons/about.ico"));
        ui->actionLOG_TRIGO-
>setIcon(QIcon(":/images/icons/logarithm.ico"));
        ui->button_0->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_1->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_2->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_3->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_4->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_5->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_6->setStyleSheet("background-color: rgb(252, 252,
252)");
```

```
        ui->button_7->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_8->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_9->setStyleSheet("background-color: rgb(252, 252,
252)");
        ui->button_decimal->setStyleSheet("background-color: rgb(252,
252, 252)");
        ui->button_dzero->setStyleSheet("background-color: rgb(252,
252, 252)");
        ui->button_clear_2->setStyleSheet("background-color:rgb(246,
246, 246)");
        ui->button_minus->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_plus->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_root->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_divide->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_mul->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_croot->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_square->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_root->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_open->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_close->setStyleSheet(("background-color:rgb(246,
246, 246)"));
        ui->button_cube->setStyleSheet(("background-color:rgb(246, 246,
246)"));
        ui->button_clear->setStyleSheet("background-color: rgb(246,
246, 246)");
        ui->button_power->setStyleSheet("background-color: rgb(246,
246, 246)");
        ui->button_fraction->setStyleSheet(("background-color: rgb(246,
246, 246)"));
        ui->button_conversion->setStyleSheet("background-color:
rgb(246, 246, 246)");
dark =false;
}

//////////////////////////////////////////
////switching between different pages...

void MainWindow::on_actionCOMPLEX_triggered()
{
    hide();
    complex= new MainWindow2(this);
    complex->show();
}

void MainWindow::on_actionLOG_TRIGO_triggered()
{
    hide();
```

```
    log=new MainWindow4(this);
    log->show();
}

void MainWindow::on_actionABOUT_triggered()
{
    abt=new about(this);
    abt->show();
}

void MainWindow::on_actionRECORD_triggered()
{
    rec=new record(this);
    rec->show();

}

//////////////////////////////////////////
//variable declaration../////////
QString value=" ",temp=" ";
bool
simple=true,sroot=false,croot=false,cube=false,square=false,power=false
;

void MainWindow::on_actionEXIT_triggered()
{
    QApplication::quit();
}
////////////////////////////////////////////////
//standard calculator code//
int precedence(char op)
{
    if (op == '+' || op == '-')
        return 1;
    if (op == '*' || op == '/')
        return 2;
    return 0;
}

float applyOp(float a, float b, char op)
{
    switch (op)
    {
    case '+':
        return a + b;
    case '-':
        return a - b;
    case '*':
        return a * b;
    case '/':
        return a / b;
    default:
        return 0;
    }
}
float evaluate(string tokens)
{
    int i;
```

```
    stack<float> values;

    stack<char> ops;
    for (i = 0; i < tokens.length(); i++)
    {

        if (tokens[i] == ' ')
            continue;
        else if (tokens[i] == '(')
        {
            ops.push(tokens[i]);
        }

        else if (isdigit(tokens[i]))
        {
            float val = 0;
            while (i < tokens.length() &&
                   isdigit(tokens[i]))
            {
                val = (val * 10) + (tokens[i] - '0');
                i++;
            }
            if(tokens[i]=='.'){
                            i++;
                            int power=-1;

                            while (i < tokens.length() &&
                                   isdigit(tokens[i]))
                            {
                                val = (val) + (tokens[i] -
'0')*pow(10,power);
                                power--;
                                i++;
                            }
                        }
            values.push(val);

            i--;
        }

        else if (tokens[i] == ')')
        {
            while (!ops.empty() && ops.top() != '(')
            {
                float val2 = values.top();
                values.pop();

                float val1 = values.top();
                values.pop();

                char op = ops.top();
                ops.pop();

                values.push(applyOp(val1, val2, op));
            }
            if (!ops.empty())
                ops.pop();
```

```
        }

        else
        {

            while (!ops.empty() && precedence(ops.top()) >=
precedence(tokens[i]))
            {
                float val2 = values.top();
                values.pop();

                float val1 = values.top();
                values.pop();

                float op = ops.top();
                ops.pop();

                values.push(applyOp(val1, val2, op));
            }

            ops.push(tokens[i]);
        }
    }

    while (!ops.empty())
    {
        float val2 = values.top();
        values.pop();

        float val1 = values.top();
        values.pop();

        float op = ops.top();
        ops.pop();

        values.push(applyOp(val1, val2, op));
    }
    return values.top();
}
float squareNum(QString value)
{
    float temp;
    temp=value.toFloat();
    temp=temp*temp;
    return temp;
}

float cubeRoot(QString value)
{
    float temp;
    temp=value.toFloat();
    temp=cbrt(temp);
    return temp;
}

float squareRoot(QString value)
{
    float temp;
```

```
    temp=value.toFloat();
    temp=sqrt(temp);
    return temp;
}
float cubeNum(QString value)
{
    float temp;
    temp=value.toFloat();
    temp=temp*temp*temp;
    return temp;
}


class Power
{
private:
    float base;
    float exponent;

public:
    void Read(string str)
    {
        int i = 0;
        stack<float> values;
        int decimal_places = 0;
        while (i < str.length())
        {
            if (isdigit(str[i]) || (str[i] == '.'))
            {
                float val = 0;
                while (i < str.length() && (isdigit(str[i]) || str[i]
== '.'))
                {
                    if (str[i] == '.')
                    {
                        decimal_places++;
                        i++;
                    }
                    else
                    {
                        val = (val * 10) + (str[i] - '0');
                        i++;
                    }
                }
                values.push(val / pow(10, decimal_places));
            }
            else if (str[i] == '^')
            {
                i++;
                decimal_places=0;
            }
            else
            {
                i++;
            }
        }
        exponent = values.top();
        values.pop();
```

```
        base = values.top();
    }

    float PowerNum()
    {
        return pow(base, exponent);
    }
};
////////////////////////////////////////////
/// decimal to fraction conversion
///

class fraction{
private:
    double input;
    long numerator,denominator;
public:
    fraction(QString fvalue)
    {
        input = fvalue.toDouble();
    }
    long gcd(long a, long b)
    {
        if (a == 0)
            return b;
        else if (b == 0)
            return a;

        if (a < b)
            return gcd(a, b % a);
        else
            return gcd(b, a % b);
    }
    void conversion()
    {
        double integral = floor(input);
        double frac = input - integral;

        const long precision = 1000000000;

        long gcd_ = gcd(round(frac * precision), precision);
        denominator = precision / gcd_;
        numerator = round(frac * precision) / gcd_;
    }
    QString display()
    {
        QString tempString=
QString::number(numerator)+"/"+QString::number(denominator);
        return tempString;
    }
};




////////////////////////////////////////
//button functionality/////////
```

```
void MainWindow::on_button_0_clicked()
{
    value=value+"0";
    temp=temp+"0";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_dzero_clicked()
{
    value=value+"00";
    temp=temp+"00";
    ui->lineEdit->setText(temp);
}

void MainWindow::on_button_1_clicked()
{
    value=value+"1";
    temp=temp+"1";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_2_clicked()
{
    value=value+"2";
    temp=temp+"2";
    ui->lineEdit->setText(temp);

}


void MainWindow::on_button_3_clicked()
{
    value=value+"3";
    temp=temp+"3";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_4_clicked()
{
    value=value+"4";
    temp=temp+"4";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_5_clicked()
{
    value=value+"5";
    temp=temp+"5";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_6_clicked()
```

```
{
    value=value+"6";
    temp=temp+"6";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_7_clicked()
{
    value=value+"7";
    temp=temp+"7";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_8_clicked()
{
    value=value+"8";
    temp=temp+"8";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_9_clicked()
{
    value=value+"9";
    temp=temp+"9";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_open_clicked()
{
    value=value+"(";
    temp=temp+"(";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_close_clicked()
{
    value=value+")";
    temp=temp+")";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_divide_clicked()
{
    value=value+"/";
    temp=temp+"/";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_mul_clicked()
{
    value=value+"*";
```

```
    temp=temp+"*";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_minus_clicked()
{
    value=value+"-";
    temp=temp+"-";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_plus_clicked()
{
    value=value+"+";
    temp=temp+"+";
    ui->lineEdit->setText(temp);
}

void MainWindow::on_button_decimal_clicked()
{
    value=value+".";
    temp=temp+".";
    ui->lineEdit->setText(temp);
}


void MainWindow::on_button_clear_2_clicked()
{
    value=value.remove(value.length()-1,1);
    temp=temp.remove(temp.length()-1,1);
    ui->lineEdit->setText(temp);
}

void MainWindow::on_button_square_clicked()
{
    temp=temp+"*"+temp;
    simple=false;
    square=true;
    ui->lineEdit->setText(temp);
}

void MainWindow::on_button_croot_clicked()
{
    temp="∛"+temp;
    simple=false;
    croot=true;
    ui->lineEdit->setText(temp);
}

void MainWindow::on_button_root_clicked()
{
    temp="√"+temp;
    simple=false;
    sroot=true;
    ui->lineEdit->setText(temp);
}
```

```
void MainWindow::on_button_cube_clicked()
{
    temp=temp+"*"+temp+"*"+temp;
    simple=false;
    cube=true;
    ui->lineEdit->setText(temp);
}
void MainWindow::on_button_clear_clicked()
{
    temp=" ";
    value=" ";
    ui->lineEdit->setText(temp);
    ui->label->setText(temp);
}


void MainWindow::on_button_fraction_clicked()
{
    value=value+"1/";
    temp=temp+"1/";
    ui->lineEdit->setText(temp);
}

void MainWindow::on_button_equal_clicked()
{
    temp=temp+"=";
    if(simple)
    {
        float total;
        QString fvalue;
        fvalue=ui->lineEdit->text();
        total=evaluate(fvalue.toStdString());
        ui->lineEdit->setText(QString::number(total));
        value=QString::number(total);
    }
    if(sroot)
    {
        ui->lineEdit->setText(QString::number(squareRoot(value)));
        value=QString::number(squareRoot(value));
    }
    if(croot)
    {
        ui->lineEdit->setText(QString::number(cubeRoot(value)));
        value=QString::number(cubeRoot(value));
    }
    if(square)
    {
        ui->lineEdit->setText(QString::number(squareNum(value)));
        value=QString::number(squareNum(value));
    }
    if(cube)
    {
        ui->lineEdit->setText(QString::number(cubeNum(value)));
        value=QString::number(cubeNum(value));
    }
    if(power)
    {
```

```
        Power p;
        p.Read(temp.toStdString());
        ui->lineEdit->setText(QString::number(p.PowerNum()));
        value=QString::number(p.PowerNum());
    }

    ///file handling
    ///
    QFile
file("C:\\Users\\shiva\\Desktop\\sdf_project\\Calculator\\calrecord.txt
");
    if(!file.open(QIODevice::Append))
    {
        qDebug()<<"File open Failed";
    }
    else
    {
        qDebug()<<"file opened";
    }
    QTextStream out(&file);
        out<<"\n"<<temp<<value<<"\n";

    file.flush();
    file.close();
    ui->label->setText(temp);
    simple=true;
    temp=" ";
    temp=value;
}


void MainWindow::on_button_conversion_clicked()
{
    fraction f(temp);
    f.conversion();
    ui->lineEdit->setText(f.display());
}


void MainWindow::on_button_power_clicked()
{
    temp=temp+"^";
    value=value+"^";
    power=true;
    ui->lineEdit->setText(temp);
}
```

6) record.cpp

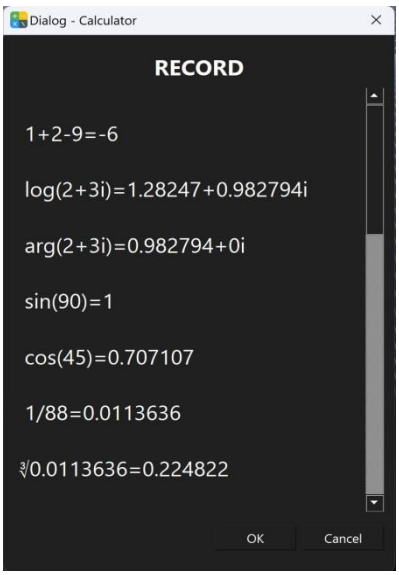
```
#include "record.h"
#include "ui_record.h"
#include <QFile>
#include <QTextStream>
#include <QDebug>
#include <QMessageBox>
#include <QDialog>
```

```
record::record(QWidget *parent) :
    QDialog(parent),
    ui(new Ui::record)
{
    ui->setupUi(this);
    ui->textBox->setStyleSheet("border:none;font-size: 20px;");
    QFile
file("C:\\Users\\shiva\\Desktop\\sdf_project\\Calculator\\calrecord.txt
");
    if(!file.open(QIODevice::ReadOnly))
    {
        QMessageBox::information(0,"info",file.errorString());
    }
    else
    {
        qDebug()<<"file opened";
    }
    QTextStream in(&file);
    ui->textBox->setText(in.readAll());


    file.close();
}

record::~record()
{
    delete ui;
}


void record::on_buttonBox_rejected()
{
    QFile
file("C:\\Users\\shiva\\Desktop\\sdf_project\\Calculator\\calrecord.txt
");
    file.open(QIODevice::WriteOnly);
    QTextStream out(&file);
    out<<" ";
}
```



References

- https://www.geeksforgeeks.org/
- https://www.programiz.com/cpp-programming
- https://stackoverflow.com/