TO DO LIST in Cpp using Qt

I've made a TO DO LIST using C++ in Qt, which will hold four records of the work the user have to do. The logics are written in C++, but the UI is made using Qt.

==============================================================================================

**TO DO LIST in C++ using Qt**

**========================================================================================**

TO DO LIST is just a list of tasks you need to complete or things that you want to do.

**Benefits of Maintaining a TO DO LIST:**

1. Improves our memory

2. Increases productivity

3. Helps with motivation

**Steps to create a TO DO LIST in Qt:**

Step 1: Install Qt on your computer.

Step 2: Create a Qt Widget Application.

Step 3: Created 8 files named task1.txt, task2.txt, task3.txt, task4.txt, deadline1.txt, deadline2.txt, deadline3.txt and deadline4.txt. We'll store the task description and deadlines in these files. Then, we'll create a resources file and add all the text files in that resource file.

Step 4: Then in **mainwindow.ui**we'll add a **GroupBox**and add 8 textBrowsers, 5 labels, 8 push buttons inside the GroupBox.

Step 5: Then we'll display the contents of the files in the textBrowsers.

Here's the code for adding the first task:

QFile task1("E:\\Internships\\1. C++ - Codespeedy Technology\\Projects\\Project 2\\toDoList\\toDoList\\Files\\task1.txt");

if(!task1.open(QIODevice::ReadOnly))

QMessageBox::information(0,"No file like this","No file like this");

QTextStream tsk1(&task1);

ui->task1->setText(tsk1.readAll());

Step 6: Afterthat, we have to create 4 other windows to add tasks in the four different textBrowsers.

Step 7: When we'll press the **Add New Task**a new window will appear.

For this we have to write this code:

addTask task1;

task1.setModal(true);

task1.exec();

QFile task\_1("E:\\Internships\\1. C++ - Codespeedy Technology\\Projects\\Project 2\\toDoList\\toDoList\\Files\\task1.txt");

if(!task\_1.open(QIODevice::ReadOnly))

QMessageBox::information(0,"No file like this","No file like this");

QTextStream tsk\_1(&task\_1);

ui->task1->setText(tsk\_1.readAll());

QFile dline\_1("E:\\Internships\\1. C++ - Codespeedy Technology\\Projects\\Project 2\\toDoList\\toDoList\\Files\\deadline1.txt");

if(!dline\_1.open(QIODevice::ReadOnly))

QMessageBox::information(0,"No file like this","No file like this");

QTextStream dl\_1(&dline\_1);

ui->deadline1->setText(dl\_1.readAll());

First three lines of code will help to pop-up a new window on the screen. After the pop-up window's work is done and it is closed by the user the rest of the lines will do their work. The rest of the lines are used to show the immediate updated file contents on the screen; the task description as well as the deadlines.

Step 8: Now let's talk about the windows which will pop-up after the **Add New Task** Button is pressed. We'll make 4 separate windows for the four tasks. Here, I'm discussing about the first task. We have to repeat this same procedure for the rest of 3 windows.

Step 9: We'he to add 2 Labels, 2 textEdits and 1 push button. Then in the **void addTask::on\_set\_clicked(){}**function we've to write the following code.

QFile file1("E:\\Internships\\1. C++ - Codespeedy Technology\\Projects\\Project 2\\toDoList\\toDoList\\Files\\task1.txt");

if(!file1.open(QFile::WriteOnly|QFile::Text)){

QMessageBox::warning(this,"No file","No file like this");

}

QTextStream out(&file1);

QString text=ui->description->toPlainText();

out<<text;

file1.flush();

file1.close();

QFile dl1("E:\\Internships\\1. C++ - Codespeedy Technology\\Projects\\Project 2\\toDoList\\toDoList\\Files\\deadline1.txt");

if(!dl1.open(QFile::WriteOnly|QFile::Text)){

QMessageBox::warning(this,"No file","No file like this");

}

QTextStream out\_dl(&dl1);

QString text\_dl=ui->setDeadline->toPlainText();

out\_dl<<text\_dl;

dl1.flush();

dl1.close();

The first 9 lines of code is used to take input form the textEdits labelled **Description**and write it in the file **task1.txt**. And the rest lines of code is used to take input from the textEdits labelled **Deadline**and write it in the file **deadline1.txt**. We've to repeat the tasks in the other 3 windows as well.

Step 10: Now in the second last step, we'll discuss about the **Done** push button. We've to write the following lines of code in each of the **void MainWindow::on\_done\_clicked(){}** functions. This function will mainly delete the contents of the files.

QMessageBox::information(0,"Congratulations","Wohoo!!! You have completed the task.....");

QFile file("E:\\Internships\\1. C++ - Codespeedy Technology\\Projects\\Project 2\\toDoList\\toDoList\\Files\\task1.txt");

file.open(QIODevice::WriteOnly | QIODevice::Text);

QTextStream out(&file);

out << "";

ui->task1->setText(out.readAll());

QFile file\_("E:\\Internships\\1. C++ - Codespeedy Technology\\Projects\\Project 2\\toDoList\\toDoList\\Files\\deadline1.txt");

file\_.open(QIODevice::WriteOnly | QIODevice::Text);

QTextStream out\_(&file\_);

out\_ << "";

ui->deadline1->setText(out\_.readAll());

Step 11: Now **Build and Run** the project and use it to manage your time.