Creating First Qt Form:

1. Open the Qt Creator
2. Click on the User Interface icon.
3. Click on File->New File or Project->Qt Widgets Application/Qt GUI Application
4. Click Choose.
5. Give a name to your project. Select a workspace.
6. Press Next->Next->Finish.
7. Your first Qt project is created.
8. Create a GUI as per the need.

➔ If you encounter No kits found error :

1. Click on the options link.
2. Select Build and Run.
3. Click on Desktop.
4. Change version from none to 4.8.5.

5. Restart Qt and create the Qt GUI Application Project.

Testing a small example

This example demonstrates to display window on screen using Qt.

We start with a very simple example.

```
#include <QApplication>
#include <QWidget>

int main(int argc, char *argv[])
{
    QApplication app(argc, argv);

    QWidget window;

    window.resize(250, 150); //resize draws window with width and height
    window.setWindowTitle("Simple example");  //add title on window
    window.show();  //shows the window

    return app.exec();
}
```

We will show a basic window on the screen.

```
#include <QApplication>
#include <QWidget>
```

We include necessary header files.

```
QApplication app(argc, argv);
```

This is the application object. Each application programmed in Qt4 must have this object. Except for console applications.

```
QWidget window;
```

This is our main widget.

```
window.resize(250, 150);
window.setWindowTitle("Simple example");
window.show();
```

Here we resize the widget. Set a title for our main window. In this case, the QWidget is our main window. And finally show the widget on the screen.

```
return app.exec();
```

We start the main loop of the application.

Figure: Simple example

This example demonstrates to display pixel on screen using qt

```
#include "mainwindow.h"
#include <QApplication>
#include<QtGui>
int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    MainWindow w;
    QImage image(300, 300, QImage::Format_RGB888);

    QRgb value;
    value=qRgb(0,255,0);   //set color of pixel as green
    image.setPixel(50,50,value);  //draws pixel with value

    image.setPixel(150,150,qRgb(255,255,255)); //draws pixel in white color at x,y
                                                //position.
    QLabel l;                       //shows pixel on screen
    l.setPixmap(QPixmap::fromImage(image));
    l.show();

                                                return a.exec();
}
```

This example demonstrates to draw a line with pixel using qt

include the following lines in main.cpp

```
QImage image(300, 300, QImage::Format_RGB888);

QRgb value;
value=qRgb(0,255,0);
//draws a line using setPixel() function
for(int x=50;x<250;++x){

        image.setPixel(x,100,value);

}
QLabel l;
l.setPixmap(QPixmap::fromImage(image));
l.show();
```

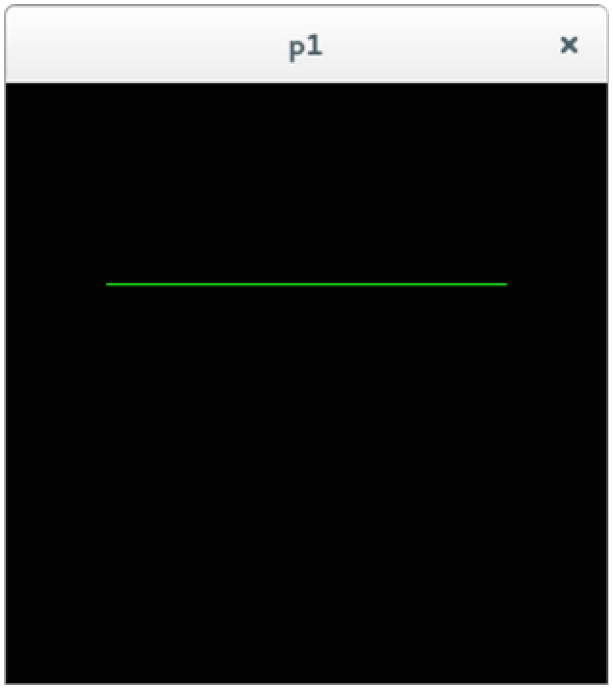

figure. Draws a line