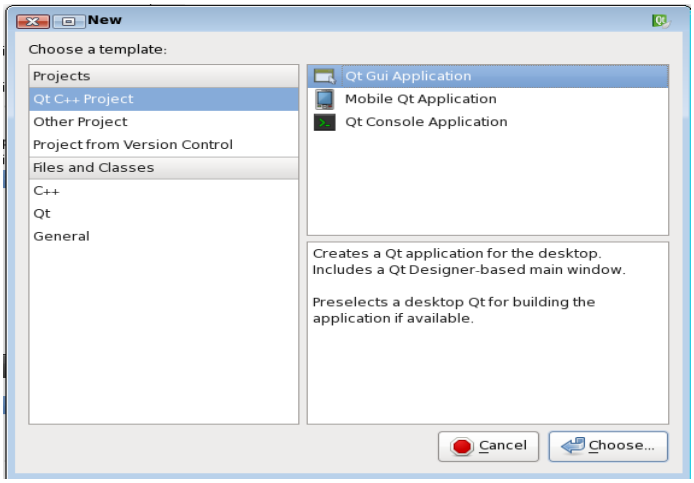
بسم الله الرحمن الرحيم و السلام على أشرف المرسلين محمد رسول الله صلى الله عليه و سلم
و السلام عليكم
تمثل هذه الوثيقة محاولة تطبيق أدوات qt و opengl معا لإنتاج تطبيق بسيط جدا و بأسهل الطرق المتوفرة في عالم الواجهات الرسومية.
التطبيق عبارة عن ثلاثة مربعات تختلف في العمق و اللون بحيث يقوم المستعمل بالنقر على أحد المربعات الثلاثة ليتغير لونه إلى اللون الأحمر , و سيتم إعداد هذا التطبيق باستعمال الواجهة الرسومية QT Gui
و سيثبت الشرح فقط على المفاهيم الخاصة بال QT لدعم الأوبن جي أل.
بالاعتماد على الصور فأنا أضن أنه سيكون الشرح واضح جدا

إصدارة الكيوتي المستخدمة في هذا التطبيق هي
qt-sdk-linux-x86-opensource-2010.04.bin
يمكن تحميلها من الموقع الرسمي لكيوتي

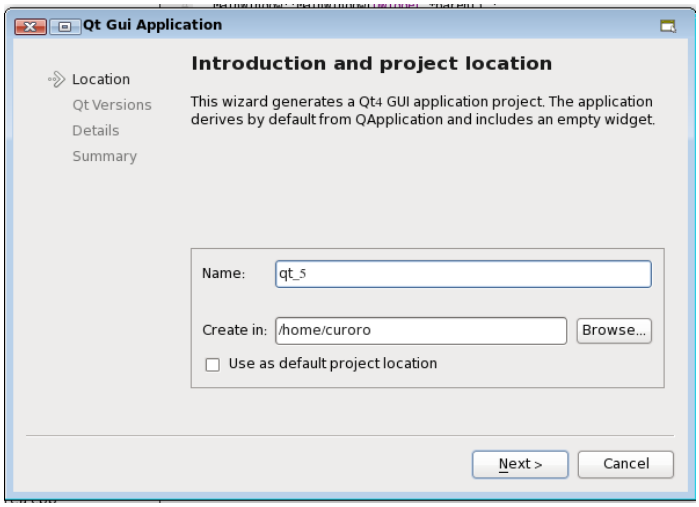
نظام التشغيل المستخدم هو أعجوبة 4
QT مدعوم بشكل كامل من قبل KDE من ناحية الواجهة الرسومية

و لنبدأ على بركة الله :

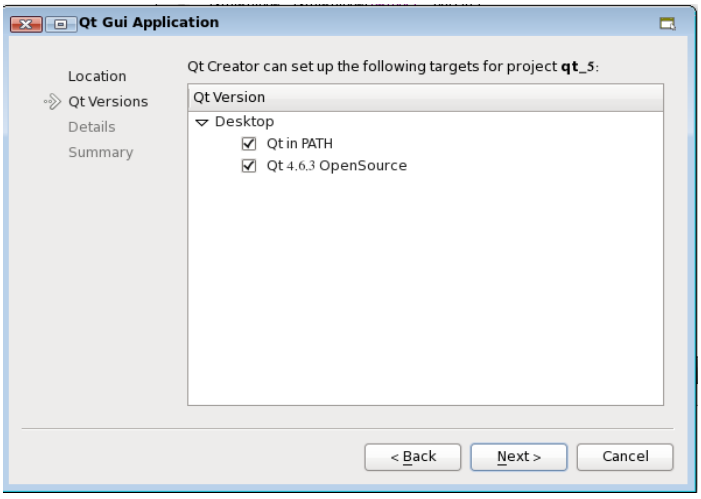
نذهب إلى : File ثم New File or Project... ثم Qt Gui Application



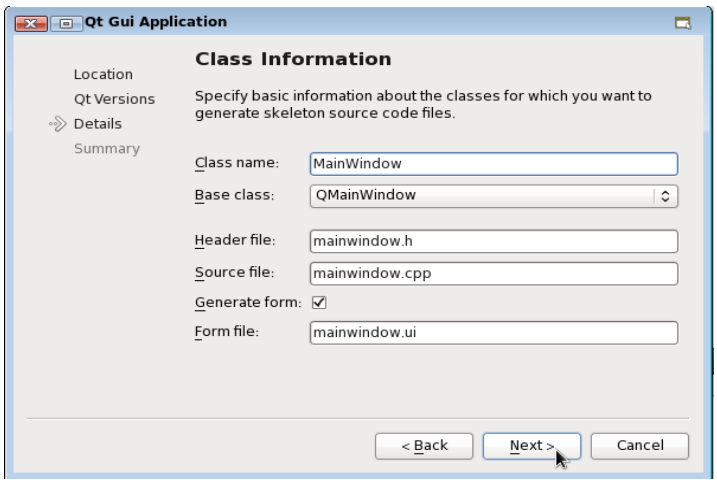
و نختار Qt Gui Application و من ثم choose



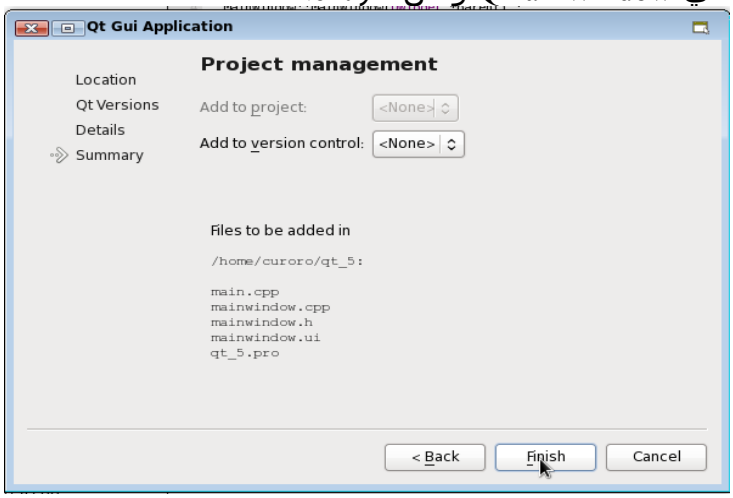
نسمي للمشروع و من ثم Next



و من ثم Next

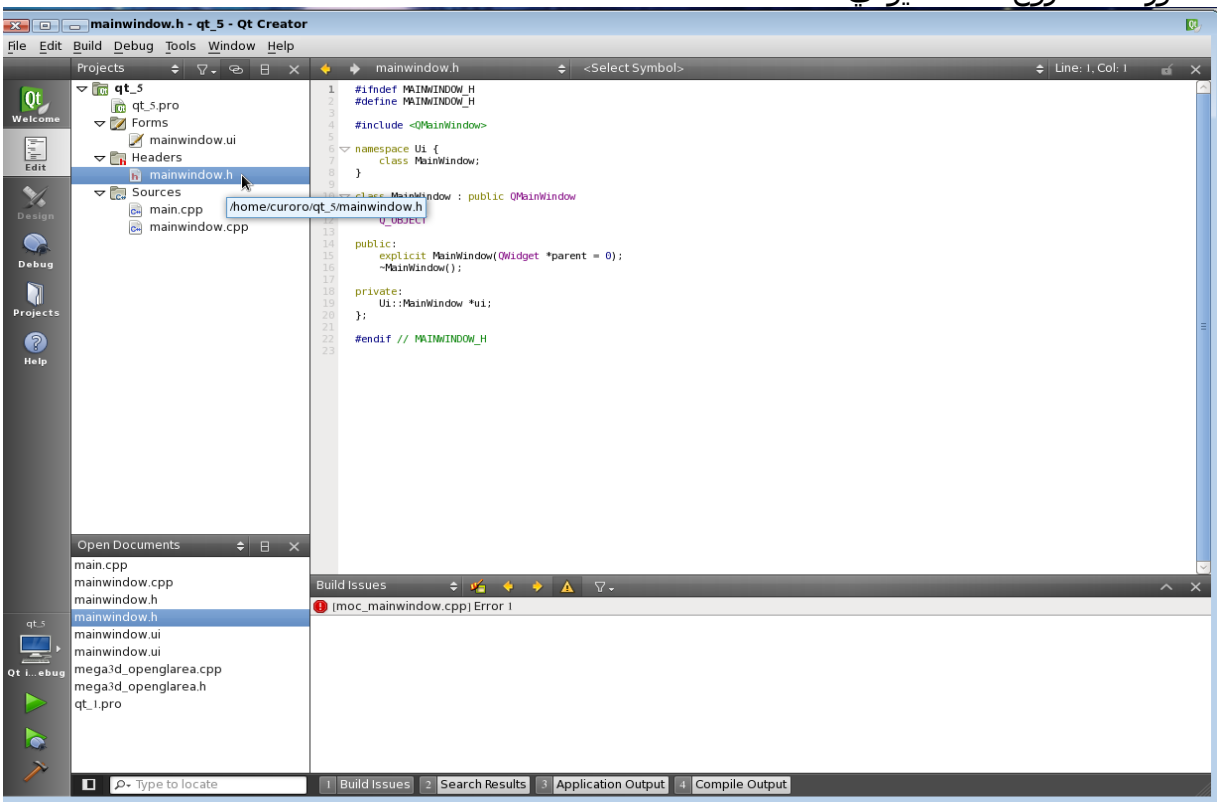


نتأكد من أن Base class هي QmainWindow و من ثم Next

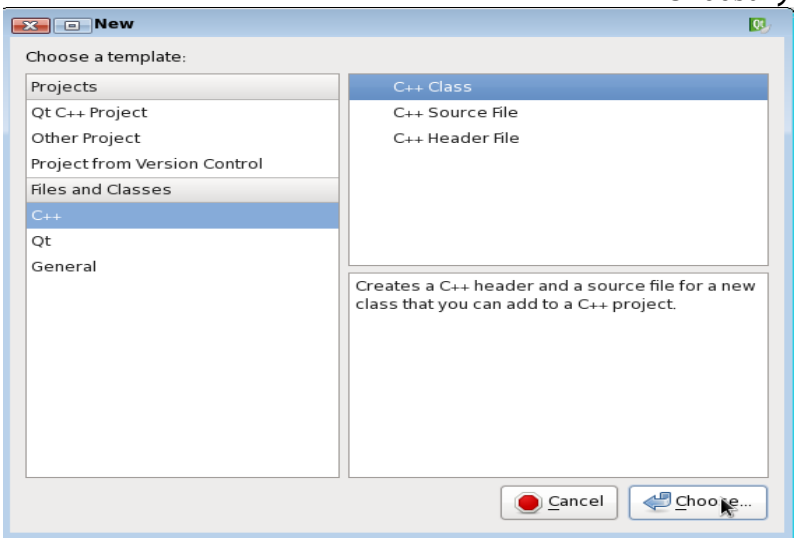


و من ثم Finish

و هذه صورة لمشروع نافذة كيوتي :

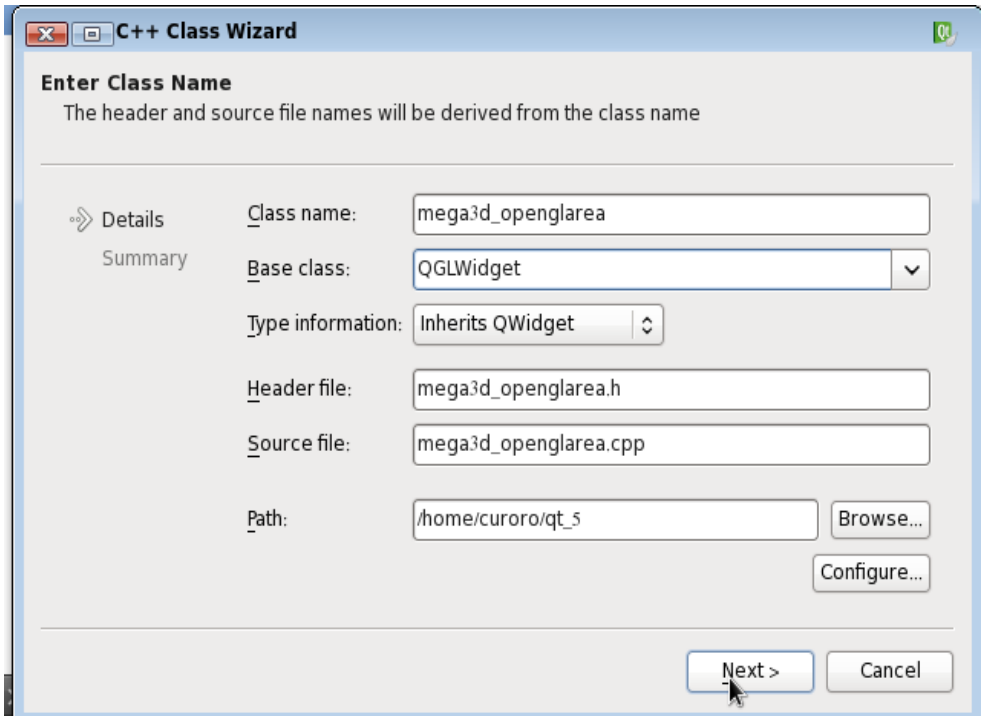


و الآن نضيف فئة للمشروع و نسميها mega3d_openglarea هي تسمية للفئة ليس إلا,
فنذهب إلى File و من ثم New File or Project.. و من ثم C++ من وحدة Files and Classes و من ثم نحدد C++ Class و ثم Choose..



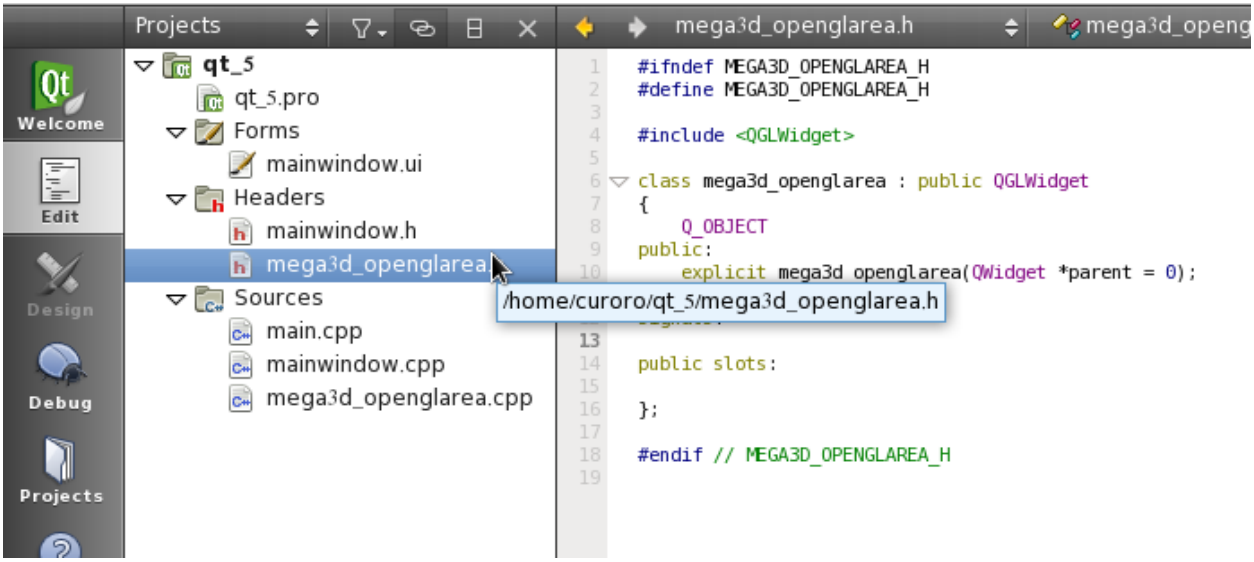
و من ثم نقوم بالإعدادت التالية :

Class name: mega3d_openglarea يمثل إسم الفئة أو الكلاس,
Base class: QGLWidget يمثل إسم الكلاس الذي سيرث منها بعض الأدوات,
Type information: inherits Qwidget يمثل الكلاس الأم لل QGLWidget.
path هو المجلد الحاوي على المشروع,



و من ثم Next و من ثم Finish.

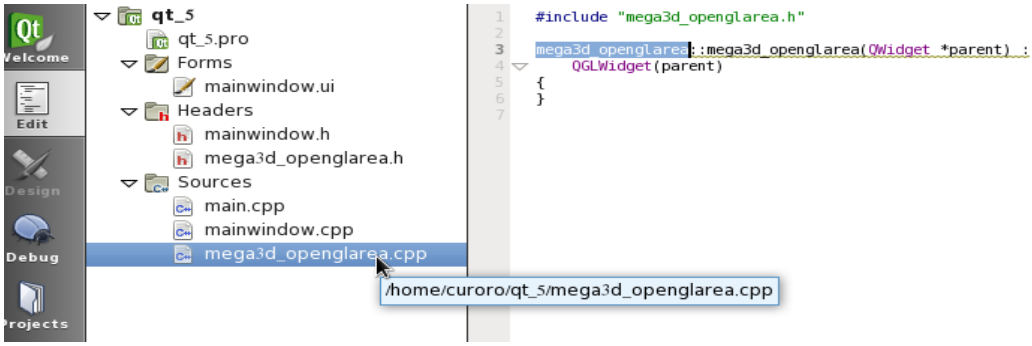
و الآن نذهب إلى فئتنا الرأسية الجديدة mega3d_openglarea.h :



ثم نقوم بإستبدال الشيفرة الإفتراضية بالشيفرة التالية :

```
#ifndef MEGA3D_OPENGLAREA_H
#define MEGA3D_OPENGLAREA_H

#include <QGLWidget>
#include <QtGui/QMouseEvent>
#include <stdio.h>


class mega3d_openglarea : public QGLWidget
{
    Q_OBJECT
public:
    explicit mega3d_openglarea(QWidget *parent = 0);

protected:
    void initializeGL();
    void resizeGL(int w, int h);
    void paintGL();
    void mousePressEvent(QMouseEvent *event);
    void mouseMoveEvent(QMouseEvent *event);
    void keyPressEvent(QKeyEvent* event);
    void drawRects(GLenum mode);
    void setname(int i) {  nameobject = i; }
    int  getname() { return nameobject; }

signals:

public slots:

private:
    int nameobject;
    void processHits(GLint hits, GLuint buffer[]);

};

#endif // MEGA3D_OPENGLAREA_H
```

و مرة ثانية نذهب إلى فئتنا المصدرية الجديدة mega3d_openglarea.cpp :



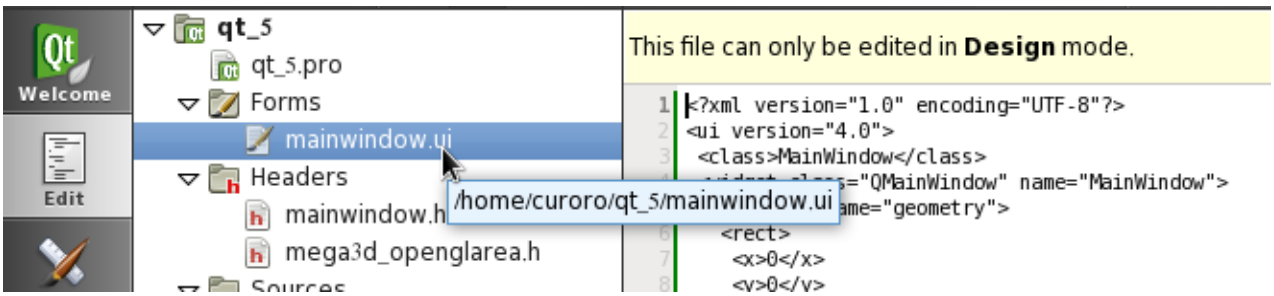
و من ثم نقوم بإستبدال الشيفرة الإفتراضية بالشيفرة التالية :

```
#include "mega3d_openglarea.h"

mega3d_openglarea::mega3d_openglarea(QWidget *parent) :
    QGLWidget(parent)
{
    nameobject = 0;
}
void mega3d_openglarea::initializeGL() {
    glClearColor(0.0, 0.0, 0.0, 0.0);
    glEnable(GL_DEPTH_TEST);
    glShadeModel(GL_FLAT);
    glDepthRange(0.0, 1.0);
}
void mega3d_openglarea::resizeGL(int w, int h) {
    glViewport(0, 0, (GLsizei) w, (GLsizei) h);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(0.0, 8.0, 0.0, 8.0, -0.5, 2.5);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
}
void mega3d_openglarea::paintGL() {
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    drawRects(GL_RENDER);
    glFlush();
}
void mega3d_openglarea::mousePressEvent(QMouseEvent *event) {
       GLuint selectBuf[512];
       GLint hits;
       GLint viewport[4];

       if (event->buttons() & Qt::RightButton)
          return;

       glGetIntegerv(GL_VIEWPORT, viewport);

       glSelectBuffer(512, selectBuf);
       (void) glRenderMode(GL_SELECT);

       glInitNames();
       glPushName(0);

       glMatrixMode(GL_PROJECTION);
       glPushMatrix();
       glLoadIdentity();
    /*  create 5x5 pixel picking region near cursor location */
       gluPickMatrix((GLdouble) event->x(), (GLdouble) (viewport[3] - event->y()),
                     5.0, 5.0, viewport);
       glOrtho(0.0, 8.0, 0.0, 8.0, -0.5, 2.5);
       drawRects(GL_SELECT);
       glPopMatrix();
       glFlush();

       hits = glRenderMode(GL_RENDER);
       processHits(hits, selectBuf);

}

void mega3d_openglarea::mouseMoveEvent(QMouseEvent *event) {
  printf("%d, %d\n", event->x(), event->y());
}
```

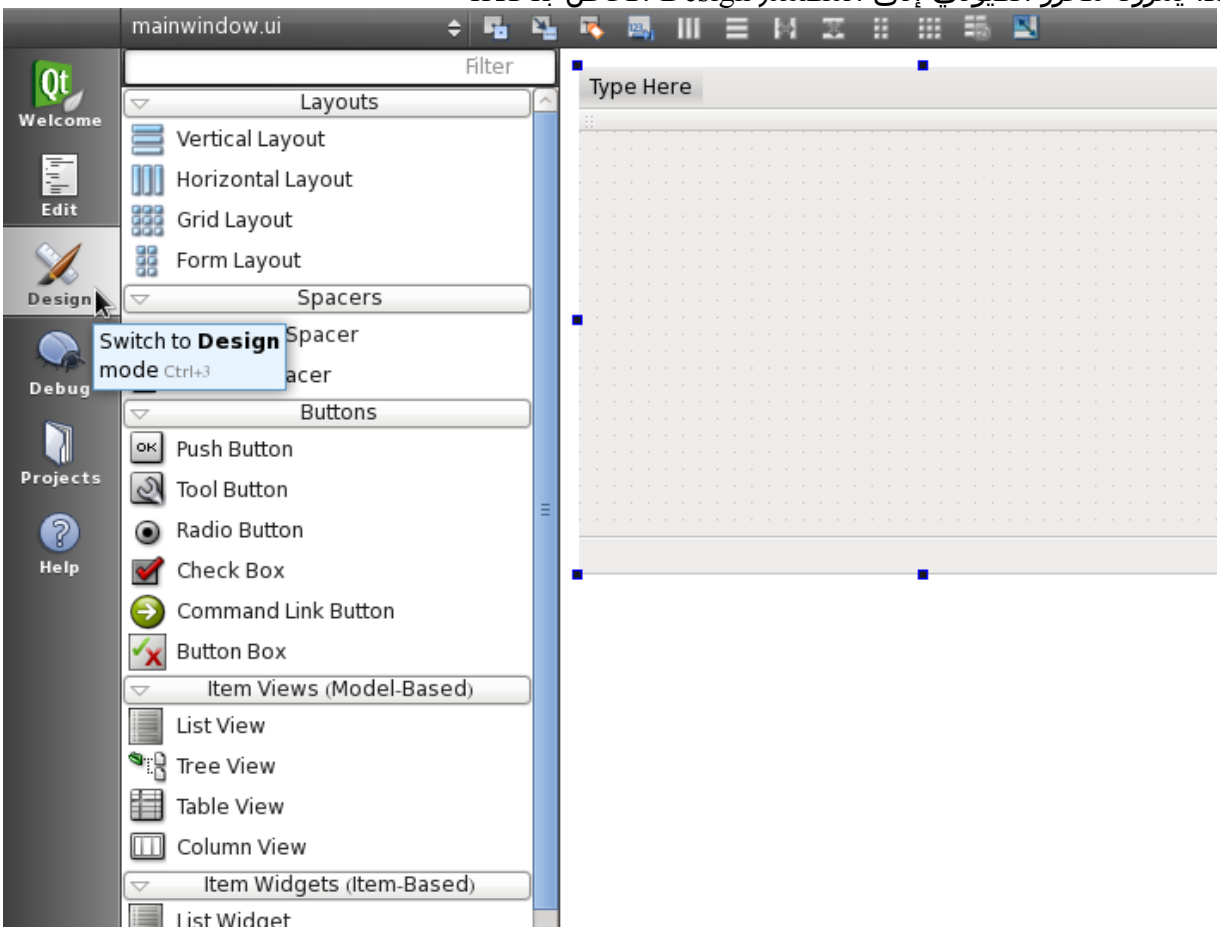
```
void mega3d_openglarea::keyPressEvent(QKeyEvent* event) {
  switch(event->key()) {
  case Qt::Key_Escape:
     close();
     break;
  default:
     event->ignore();
     break;
  }
}
void mega3d_openglarea::processHits(GLint hits, GLuint buffer[])
{
      GLuint *ptr;
      ptr = (GLuint *) buffer;
       if(hits == 0) setname(0); ;
       if(hits!=0)
       {
           ptr+=((3+(*ptr)))-1;
           printf("   the name is %d",*ptr);
           setname(*ptr);
           printf("\n");
       }

   updateGL();
}
void mega3d_openglarea::drawRects(GLenum mode)
{
   if (mode == GL_SELECT)
      glLoadName(1);
   glBegin(GL_QUADS);
   glColor3f(1.0, 1.0, 0.0);
   if(getname() == 1) glColor3f(1.0, 0.0, 0.0);
   glVertex3i(2, 0, 0);
   glVertex3i(2, 6, 0);
   glVertex3i(6, 6, 0);
   glVertex3i(6, 0, 0);
   glEnd();
   if (mode == GL_SELECT)
      glLoadName(2);
   glBegin(GL_QUADS);
   glColor3f(0.0, 1.0, 1.0);
   if(getname() == 2) glColor3f(1.0, 0.0, 0.0);
   glVertex3i(3, 2, -1);
   glVertex3i(3, 8, -1);
   glVertex3i(8, 8, -1);
   glVertex3i(8, 2, -1);
   glEnd();
   if (mode == GL_SELECT)
      glLoadName(3);
   glBegin(GL_QUADS);
   glColor3f(1.0, 0.0, 1.0);
   if(getname() == 3) glColor3f(1.0, 0.0, 0.0);
   glVertex3i(0, 2, -2);
   glVertex3i(0, 7, -2);
   glVertex3i(5, 7, -2);
   glVertex3i(5, 2, -2);
   glEnd();
}

```

و الآن بعد أن إنتهينا من إعداد شيفرة الفئة mega3d-openglarea نذهب لإعداد النافذة MainWindow.ui :

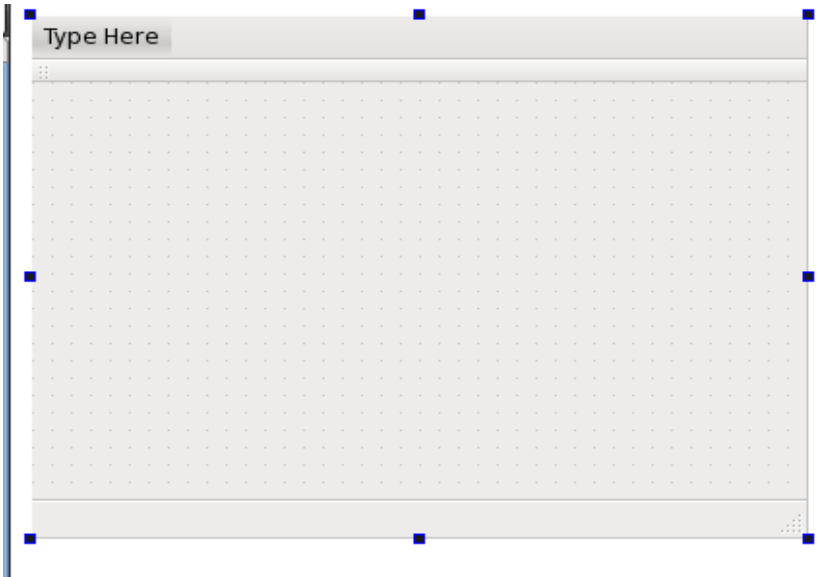
إذن نذهب إلى MainWindow.ui و نقوم بالنقر عليها مرتين



و هكذا يمررنا محرر الكيوتي إلى المصمم Design الخاص بنافذتنا

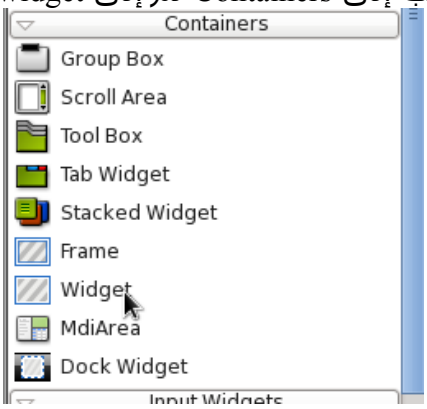


و هذه نافذتنا الفارغة :

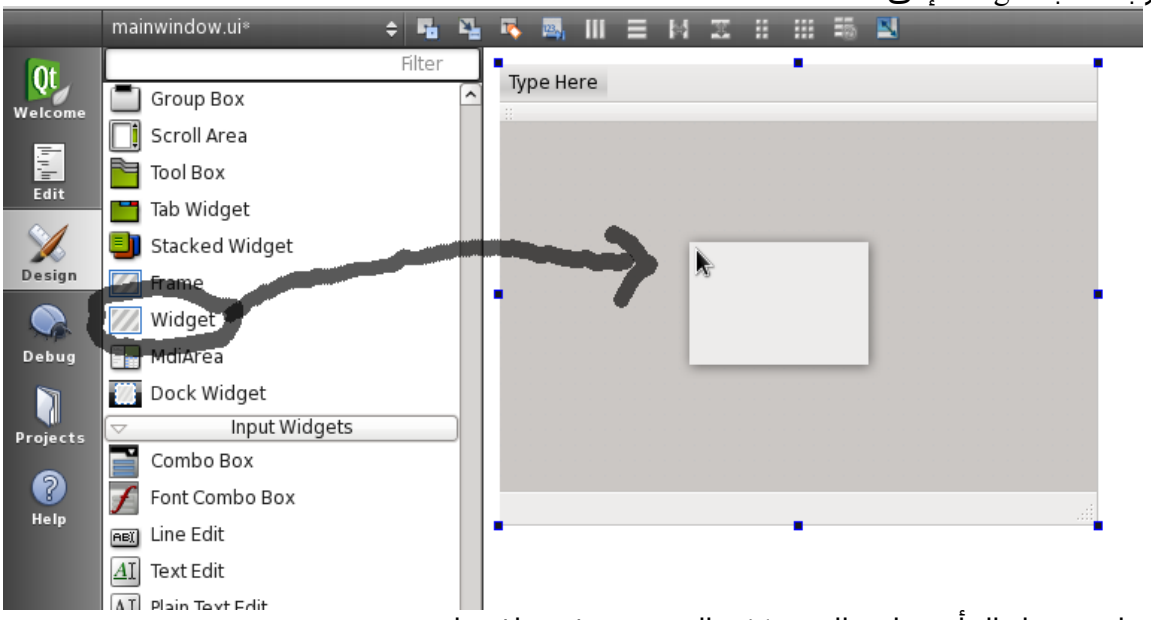


و الآن جاء دور إعداد النافذة بالمتطلبات الضرورية لتتفاعل مع الأوبن جي أل :

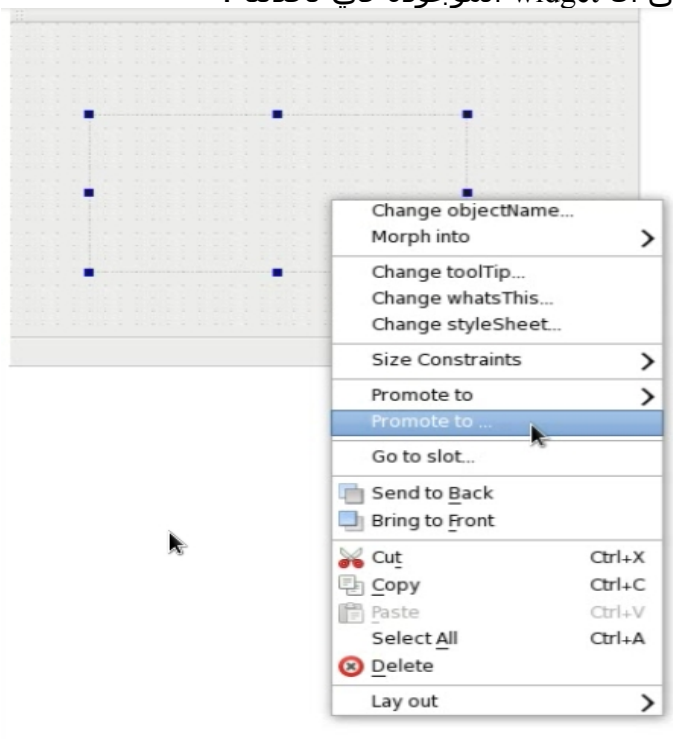
من اليسار أو من يمين الكيوتي نذهب إلى Containers ثم إلى widget



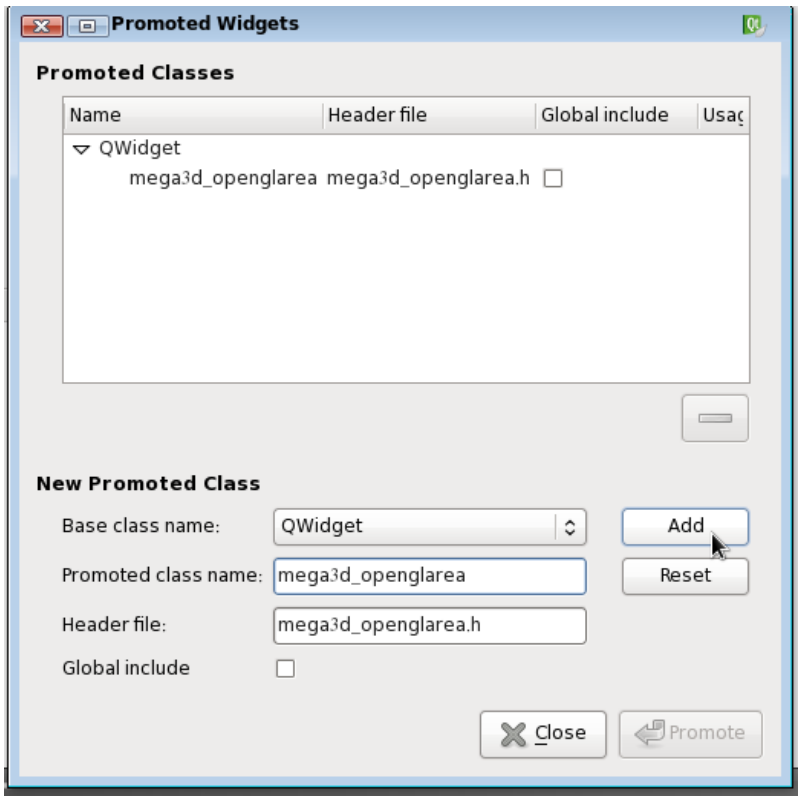
ثم نقوم بسحب widget إلى النافذة



ثم ننقر على يسار الفأرة على ال widget الموجودة في نافذتنا :



ثم نختار Promote to.. لنذهب إلى



و نقوم بإعدادها على الشكل :

Base class name: Qwidget

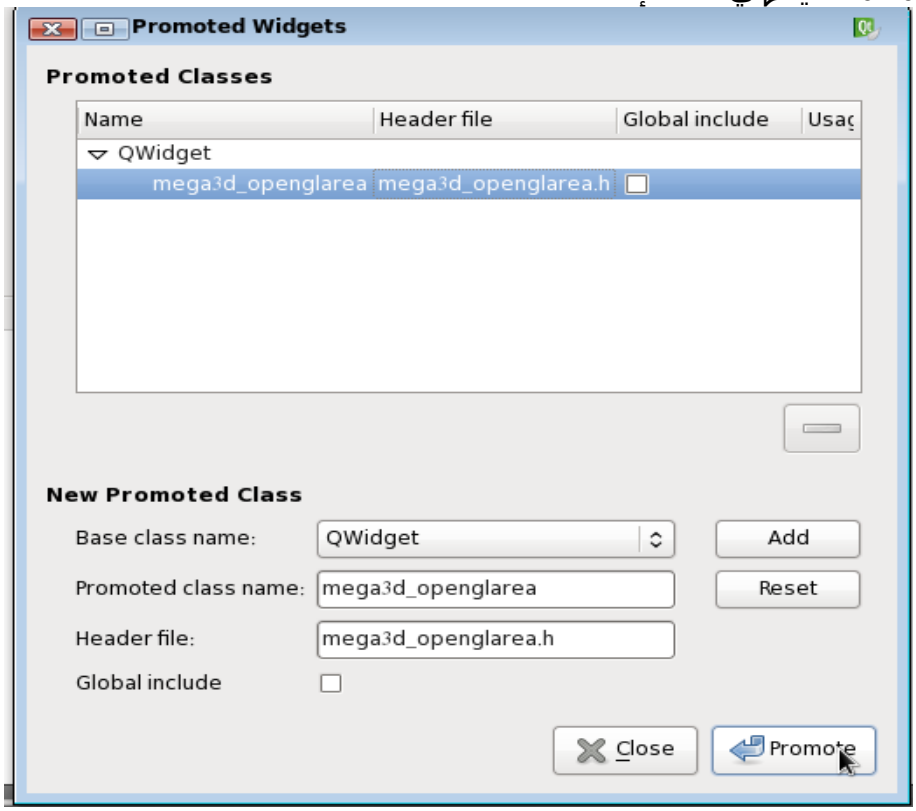
Promote class name: mega3d_openglarea و هو إسم الكلاس التي سترفى إلى الأدات widget .

Header file: mega3d_openglarea.h إسم الملف الرأسي الحاوي على الفئة mega3d_openglarea .

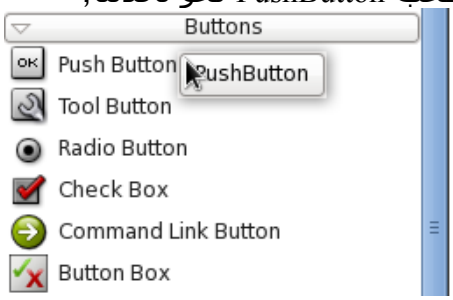
و من ثم نقوم بالنقر على Add لتظهر لنا الترقية على ال QWidget في الأعلى على Promoted Classes.

و من ثم نقوم بتحديد Qwidget [] mega3d_openglarea.h mega3d_openglarea

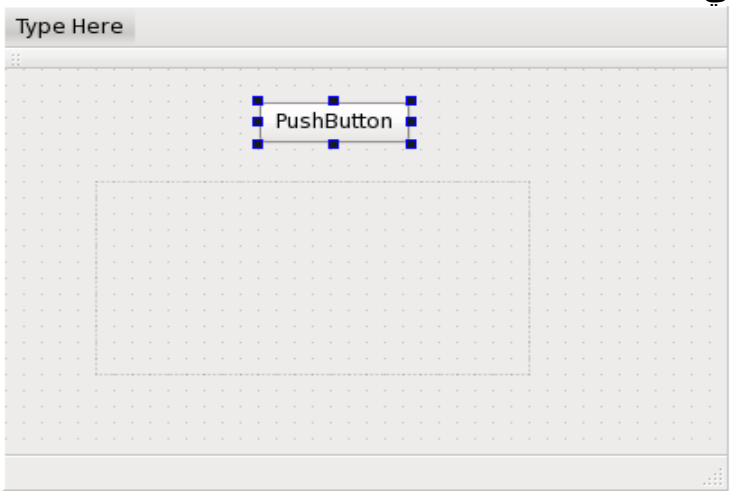
ثم ننقر على Promote لينتهي هذا الإعداد.



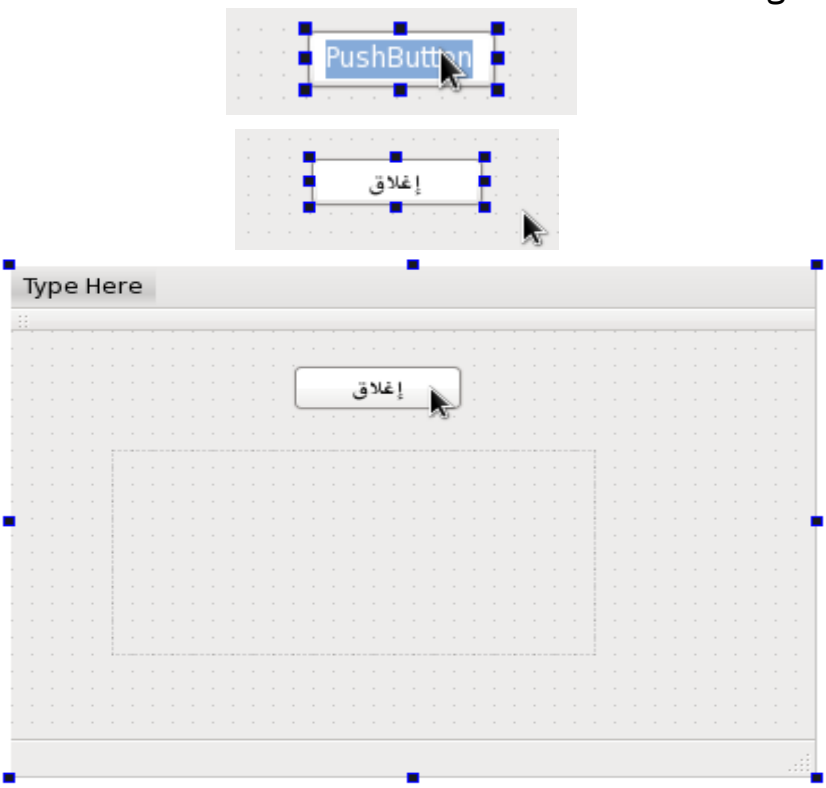
و الآن سنقوم ببعض الإضافات على النافذة :
نذهب إلى الخيارات Buttons ثم نسحب PushButton نحو نافذتنا,



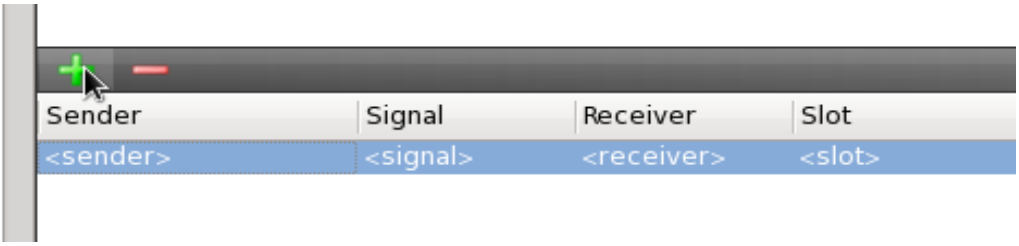
لتظهر على الشكل التالي :



نغير الإسم من PushButton نحو إغلاق و ذاك عبر النقر مرتين على الزر الظاهر بإسم PushButton
لتظهر النافذة على الشكل :



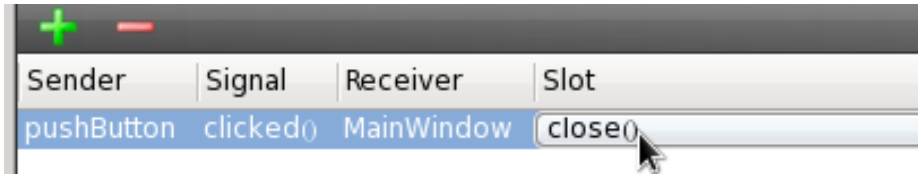
و الآن بقي إعدادت signals & slots :
نذهب إلى الإشارة + الظاهرة باللون الأخضر و نقوم بالنقر عليها ليظهر لنا signals & slots جديد في الأسفل, و يتكون من Sender و Receiver أي مرسل و مستقبل ,



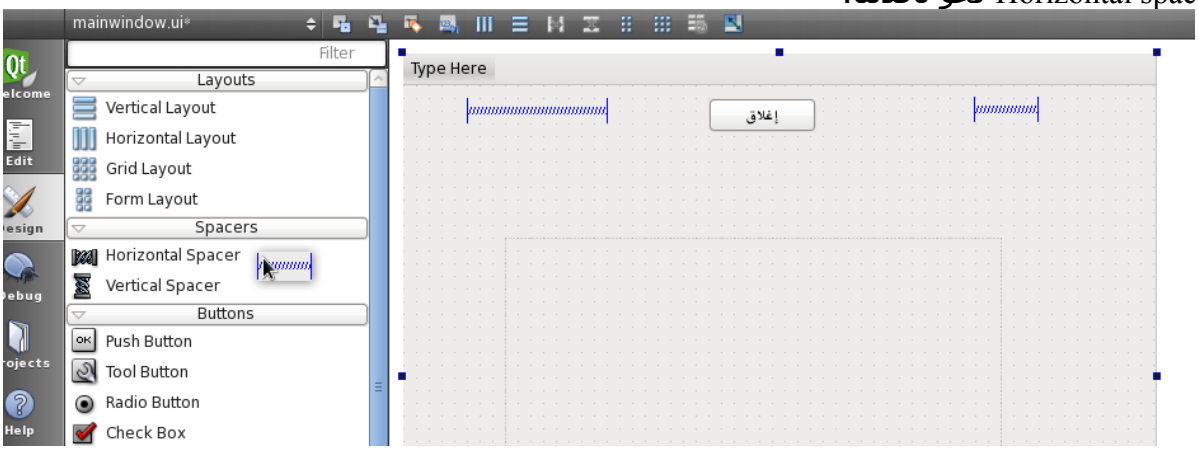
هنا سنجعل الزر "إغلاق" هو المرسل و نافذتنا MainWindow هي المستقبل, و سنختا لكليهما الحدث الخاص به.
إذن سنقوم بالإعدادات التالية:

Sender = pushButton
Signal = clicked
Receiver = MainWindow
Slot = close

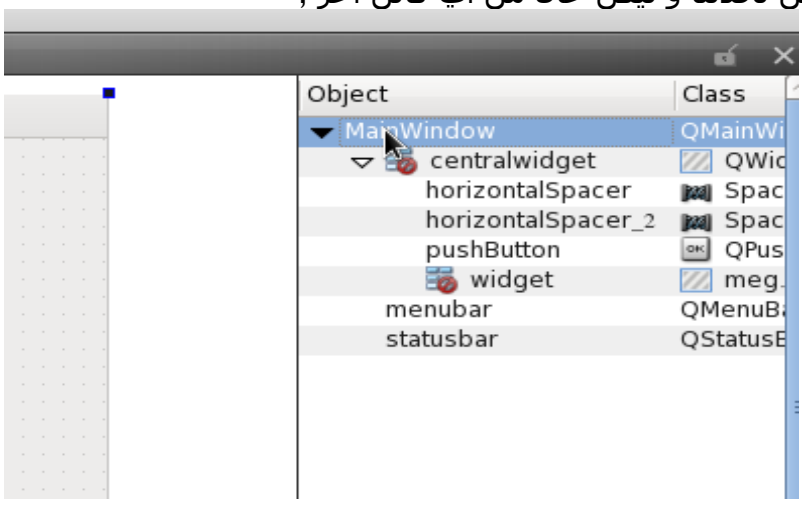
بهذا الشكل :



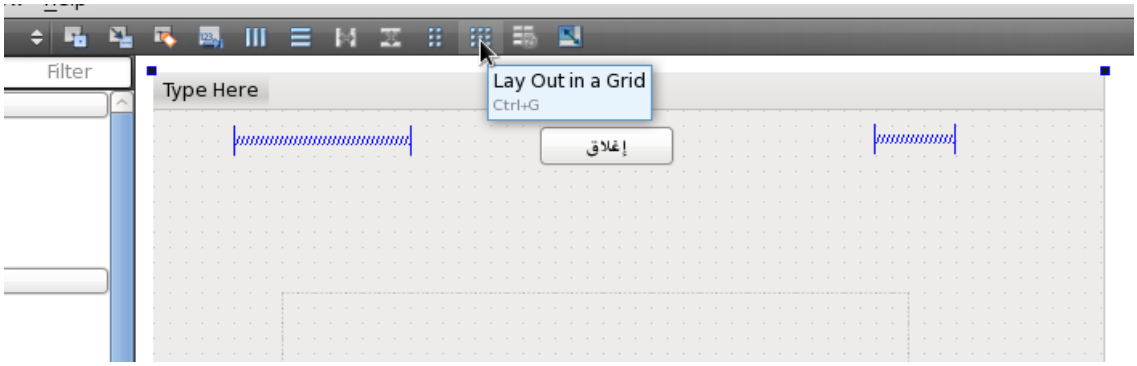
ثم نسحب الخيار Spacers :
من Horizontal spacer نحو نافذتنا.



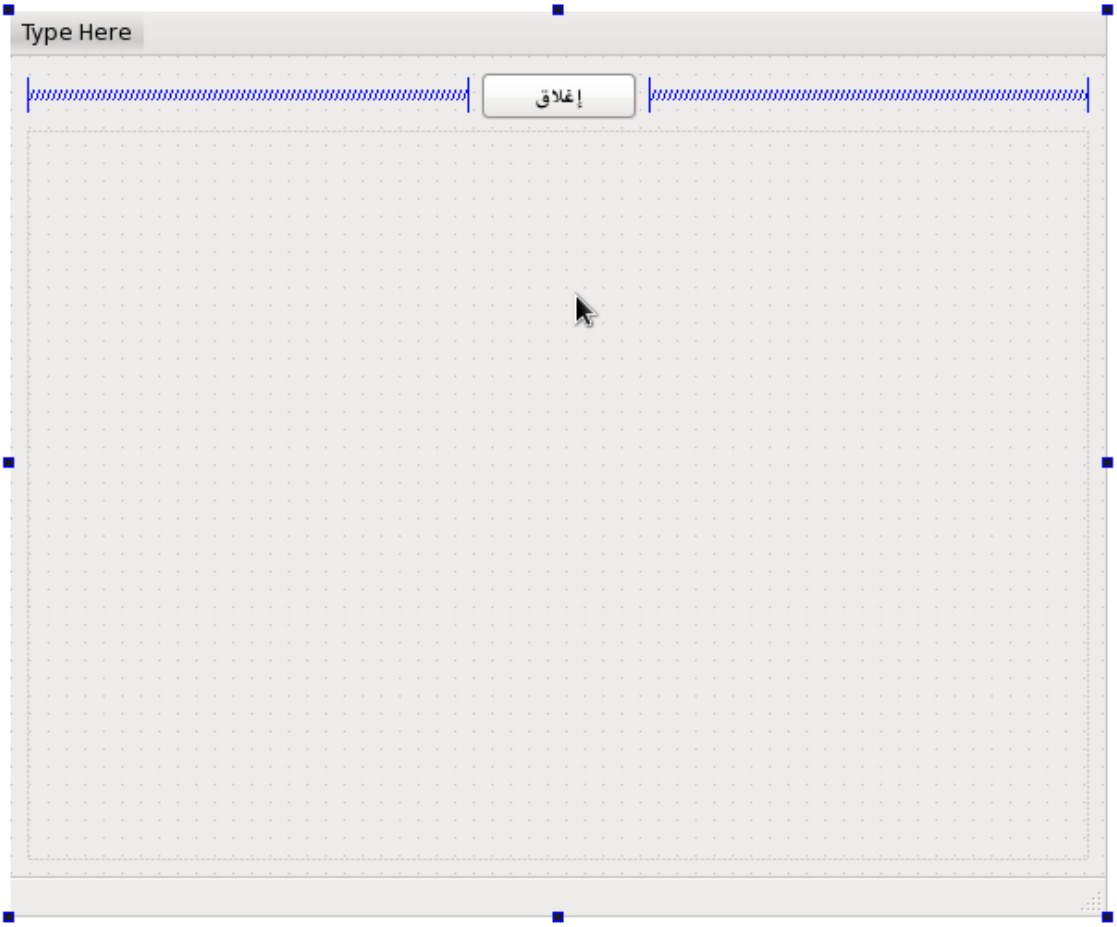
واحد على يمين الزر " إغلاق" و الآخر على يساره.
ثم نذهب إلى الكائن MainWindow و من ثم نحدد ليظهر باللون الأزرق , الغرض من هذا تحديد نافذتنا, أو النقر في أي جزء من نافذتنا و ليكن خال من أي كائن آخر ,



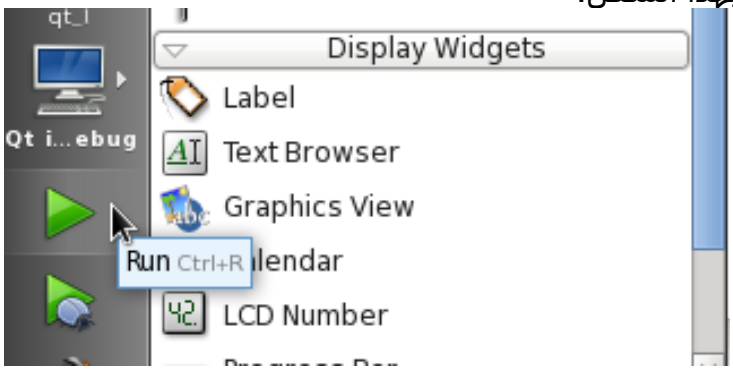
ثم نذهب إلى الخيار Lay Out in a Grid و من ثم بقوم بالنقر عليه



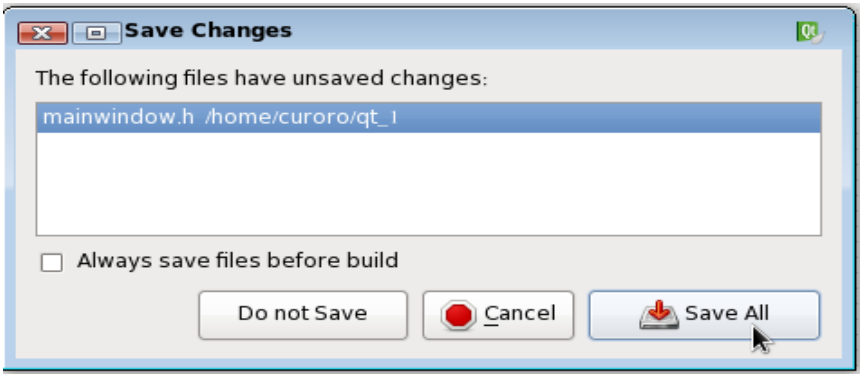
و النتيجة هي :



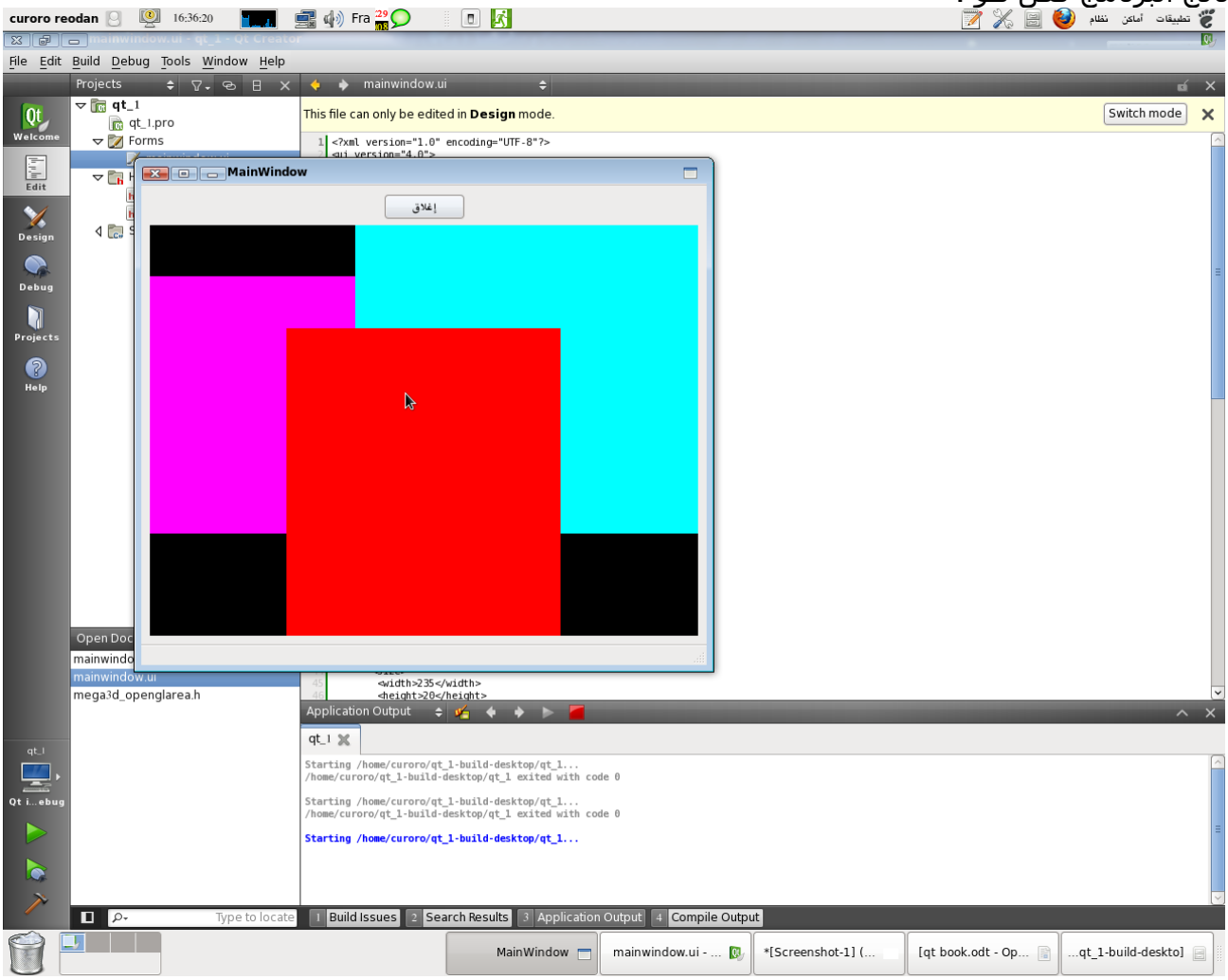
و الآن جاء دور تنفيذ البرنامج :
من القائمة نذهب إلى Build و من ثم Run
أو
نقوم بالنقر على Run , بهذا الشكل:



و سيطلب منك المحرر كيوتي تثبيت حفظ المشروع عندها ننقر على الزر Save All .



و ناتج البرنامج ككل هو :



إنتهى,