



Qt Tutorial

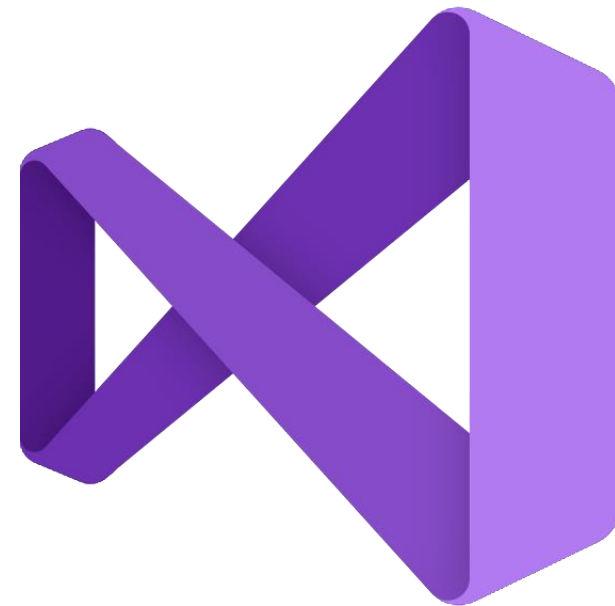
Object Oriented Programming



- Qt is cross-platform software for creating graphical user interfaces.

Tools

- Qt 5.15.2
- Visual Studio 2019 (Optional)



Authorization

- Create a Qt account
- Note: NTUST mail may not be verified
- [Link](#)



Qt Account

Your unified login to everything Qt

Sign in

[Forgot password?](#)

[Create Qt Account](#)

With Qt Account you get access to exclusive services, support and information. Use your Qt Account credentials to sign-in to all Qt services, including Qt Bugreports, Codereview, Forums & Wiki, and The Qt Company Support Center.

Authorization

- Verification

Congratulations!

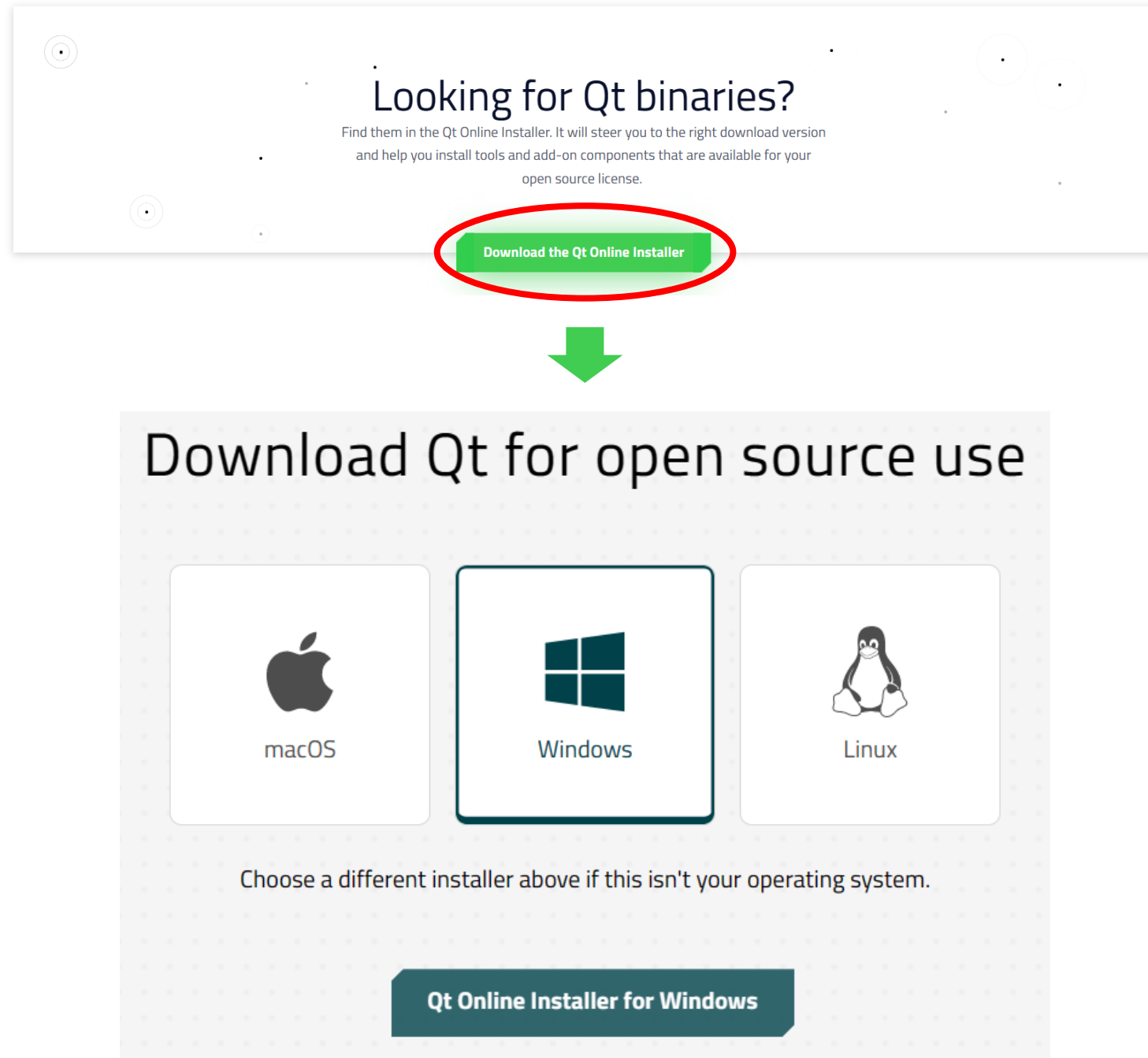
Your Qt Account has been created.

We have sent confirmation link to your email. Press continue to start using your account.

Continue

Download

- Download the installer
- [Link](#)



The screenshot shows the Qt website's download page. At the top, a section titled "Looking for Qt binaries?" explains that the Qt Online Installer will guide users to the correct download version and help with installation. Below this text, a green button labeled "Download the Qt Online Installer" is circled in red. A large green arrow points down from this button to a section titled "Download Qt for open source use". This section contains three buttons for different operating systems: macOS (with an Apple logo), Windows (with a Windows logo and a dark blue border), and Linux (with a Tux penguin logo). Below these buttons, a note says "Choose a different installer above if this isn't your operating system." At the bottom, a dark blue button labeled "Qt Online Installer for Windows" is visible.

Looking for Qt binaries?

Find them in the Qt Online Installer. It will steer you to the right download version and help you install tools and add-on components that are available for your open source license.

Download the Qt Online Installer

Download Qt for open source use

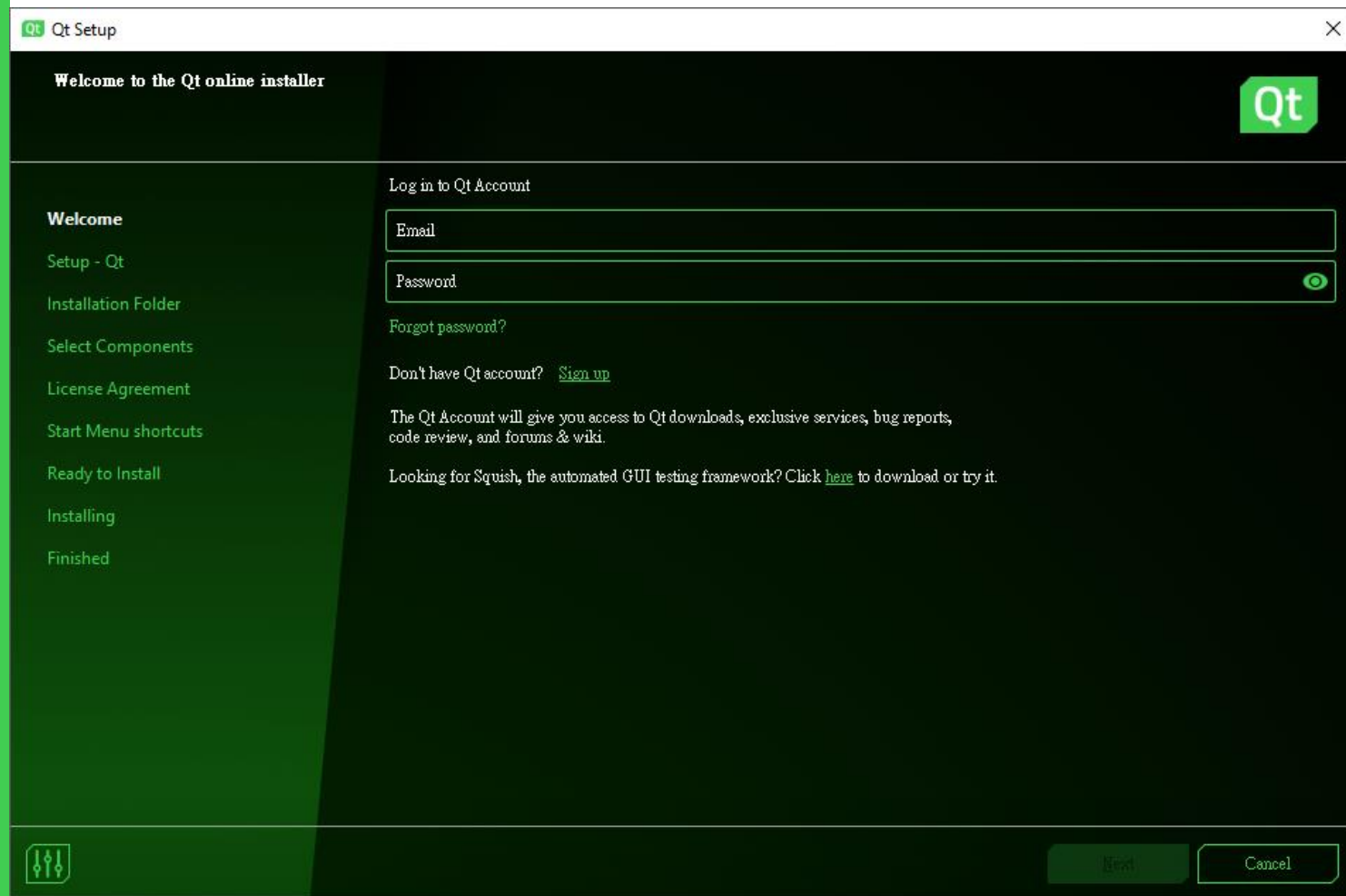
macOS Windows Linux

Choose a different installer above if this isn't your operating system.

Qt Online Installer for Windows

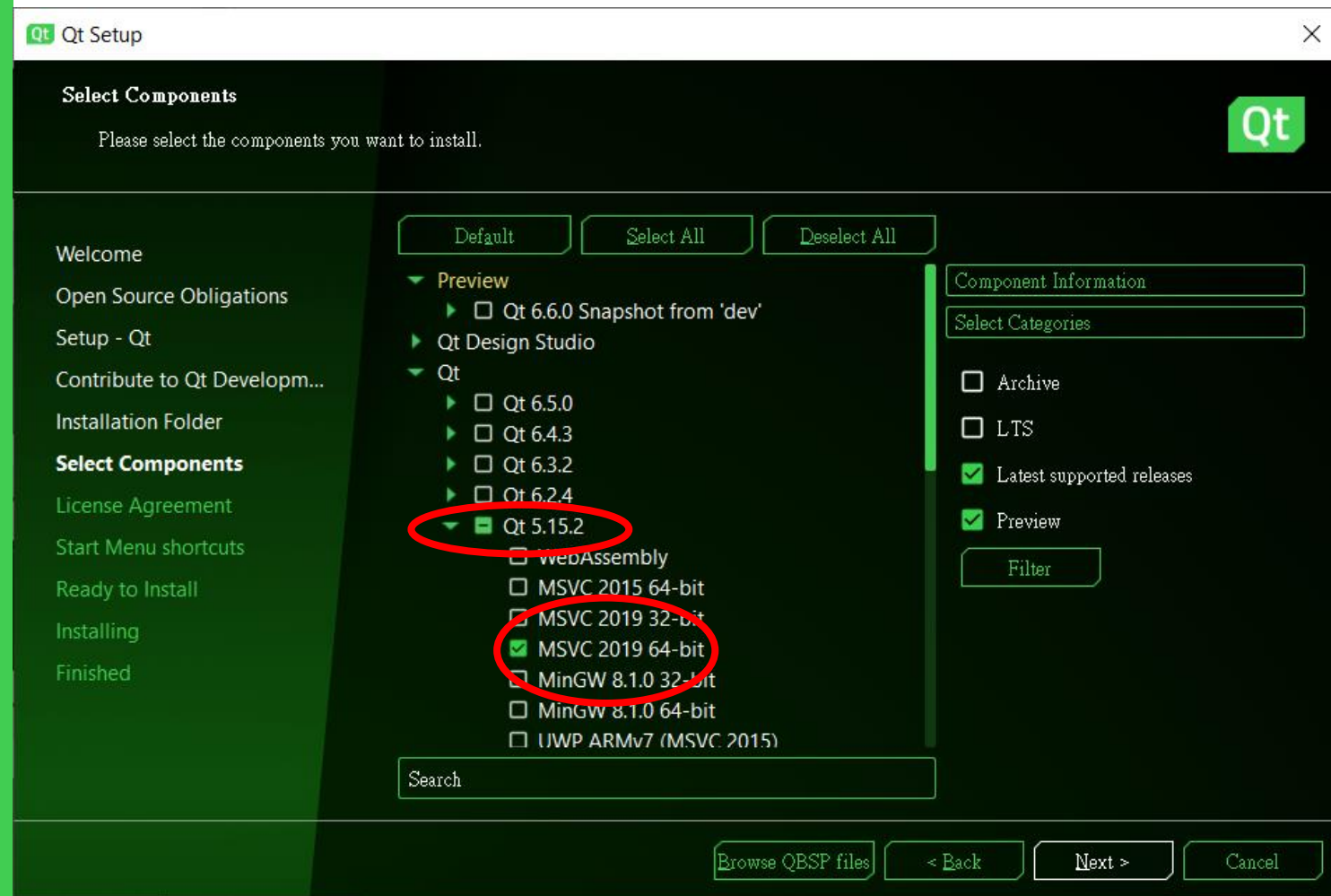
Setup

- Log in to Qt Account
- Next until next slide



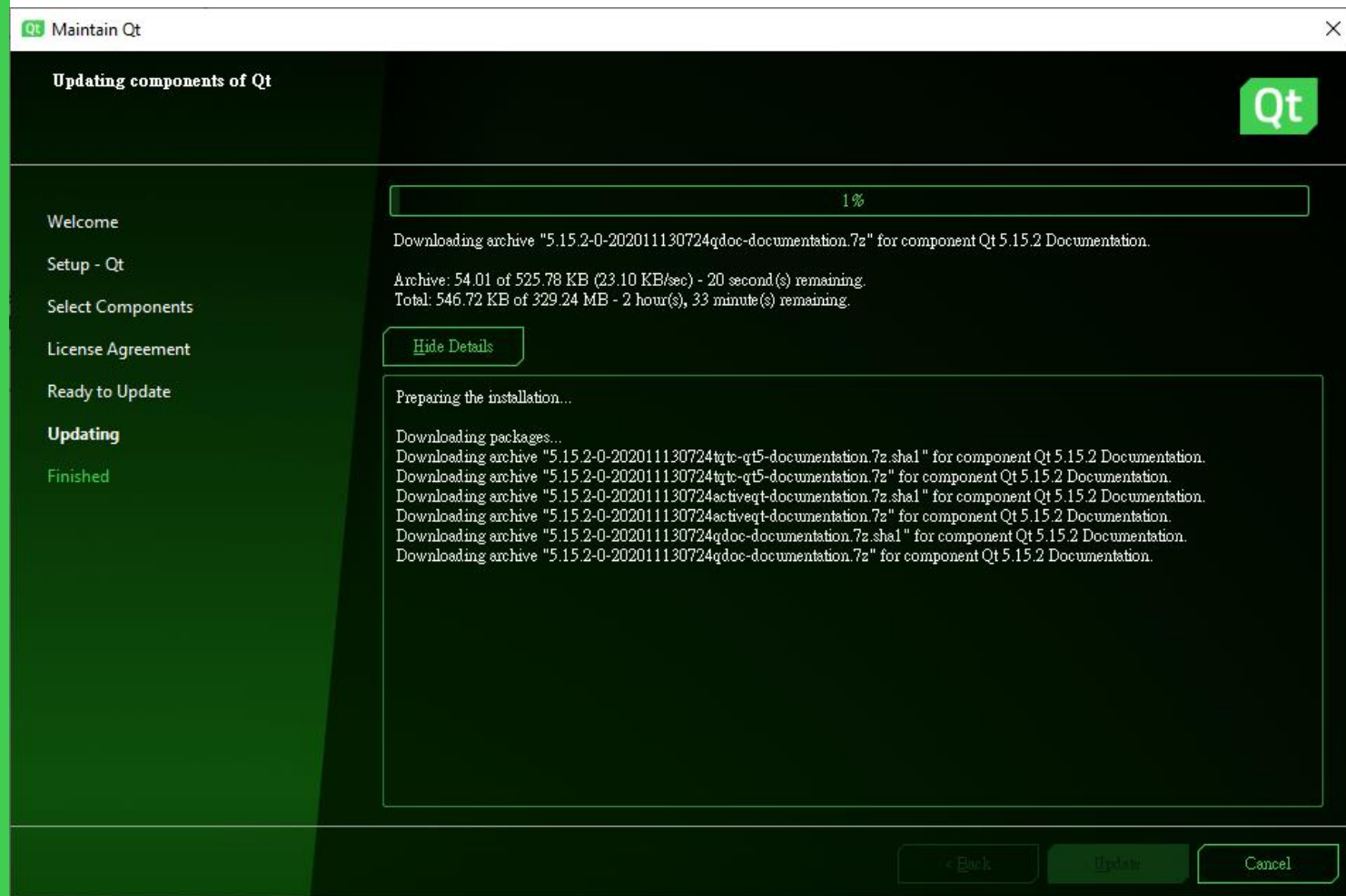
Setup

- Select MSVC 2019 64-bit
- Next



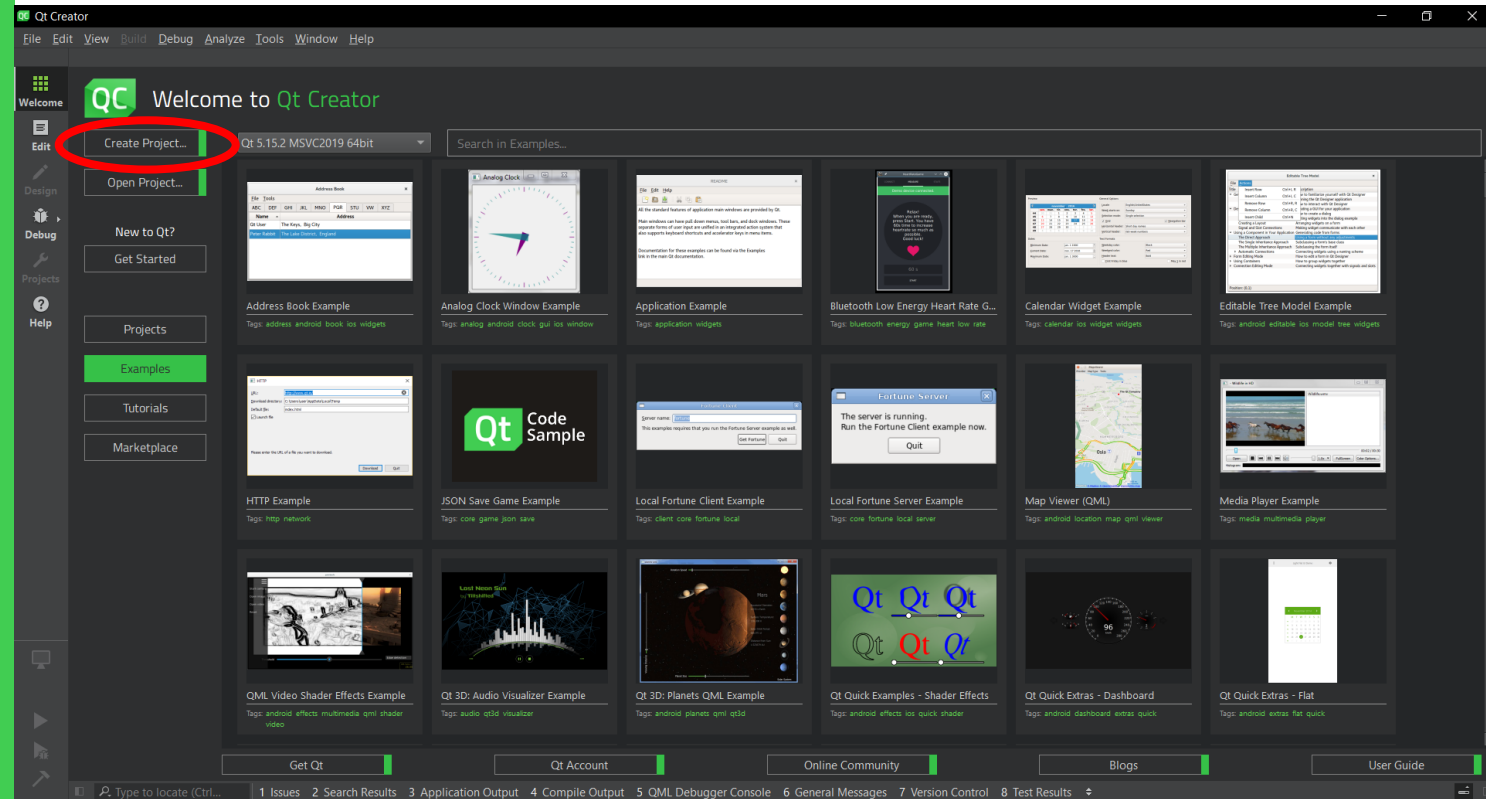
Setup

- Time: about 10 mins



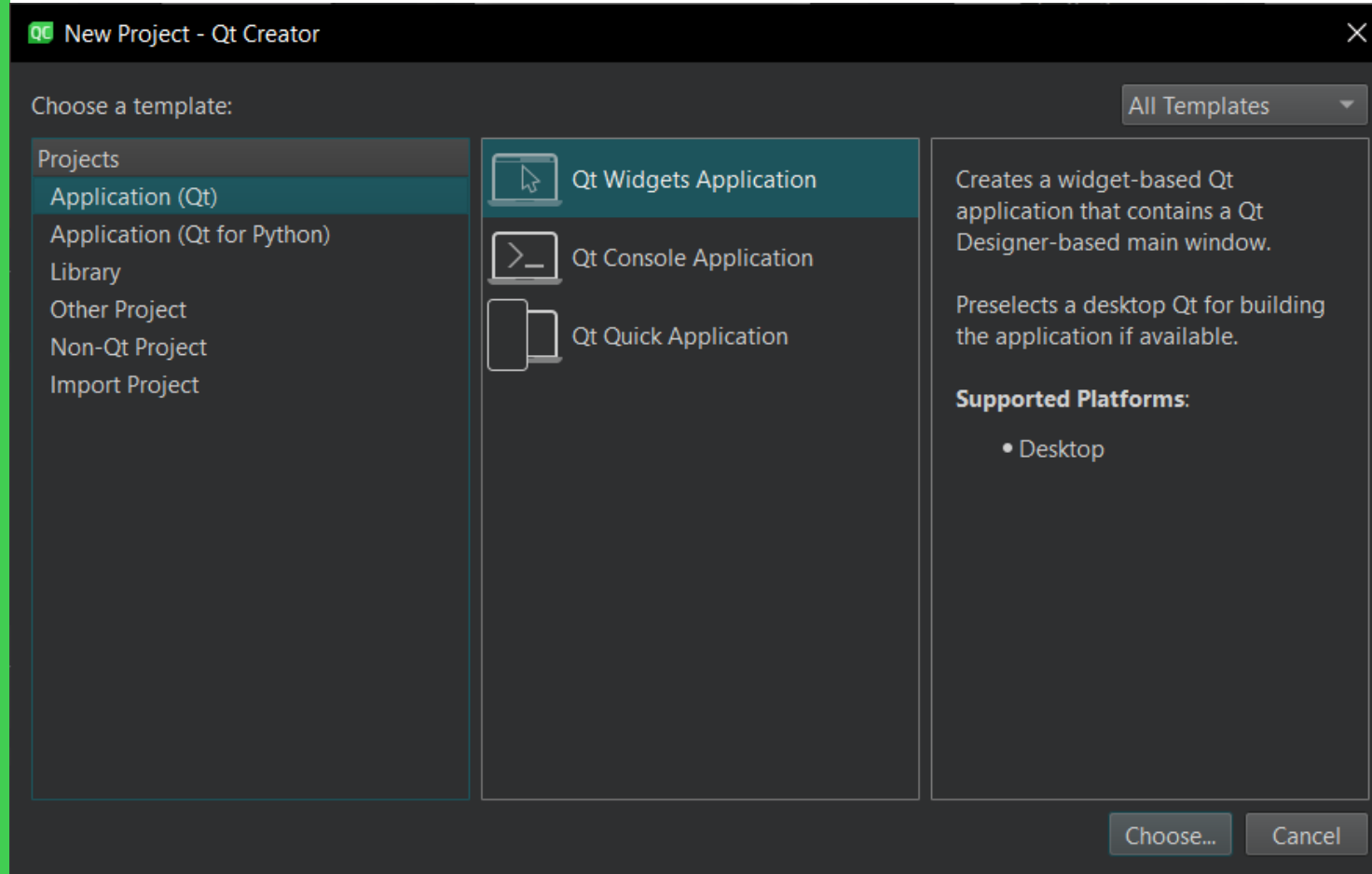
Qt Creator

- Create Project



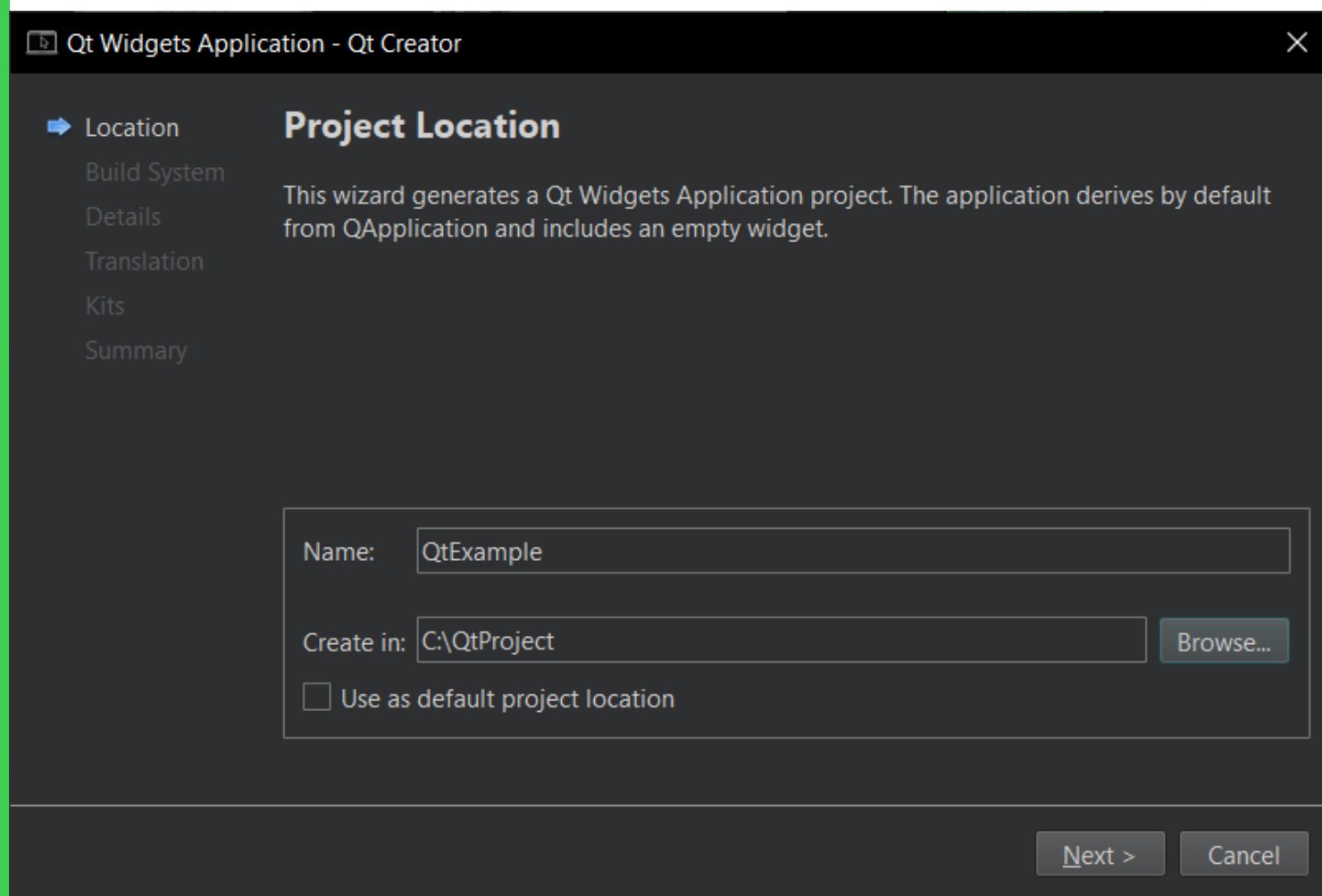
Qt Creator

- Qt Widgets Application



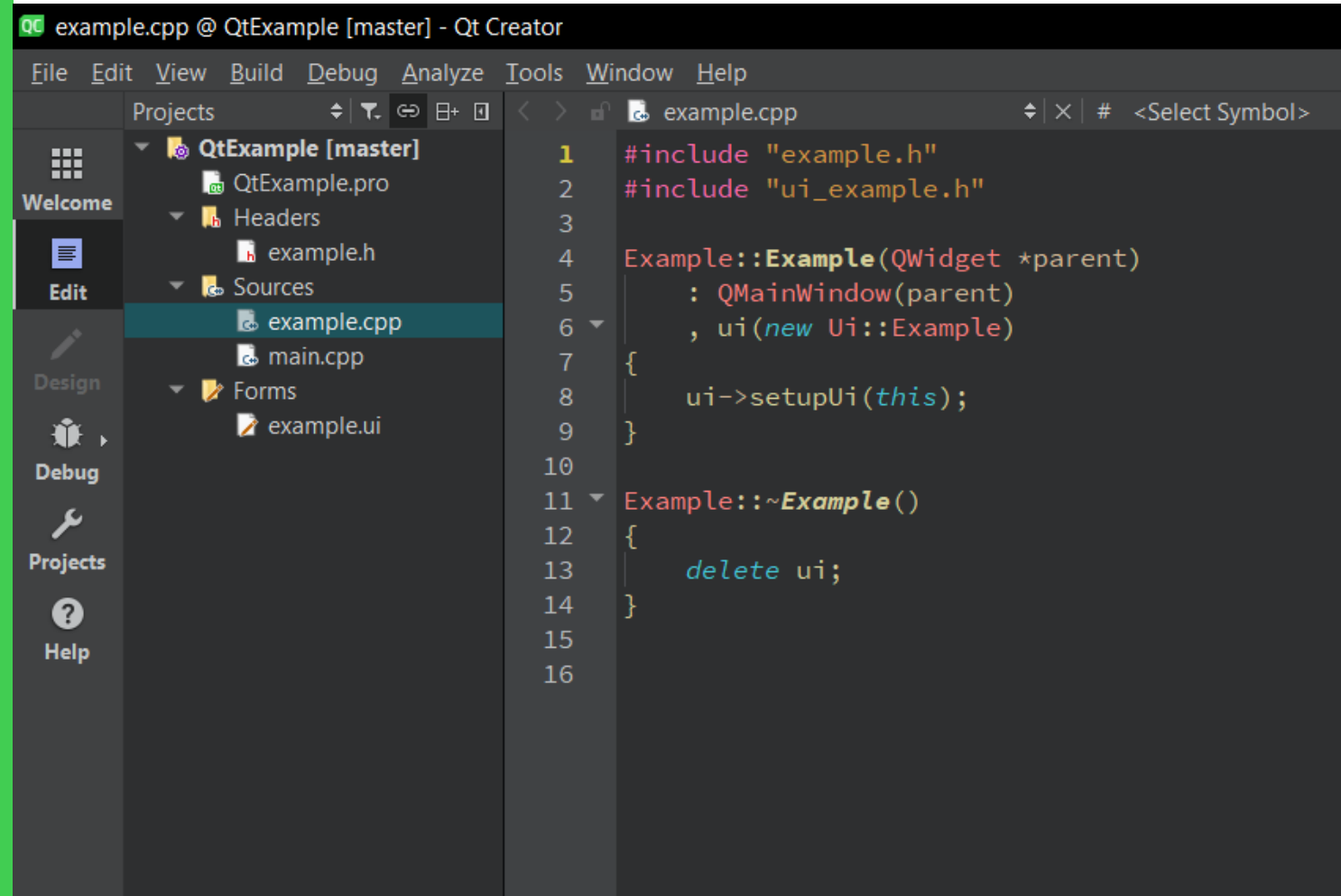
Qt Creator

- Attention!
Please **DO NOT** include
Chinese in the path
- Next




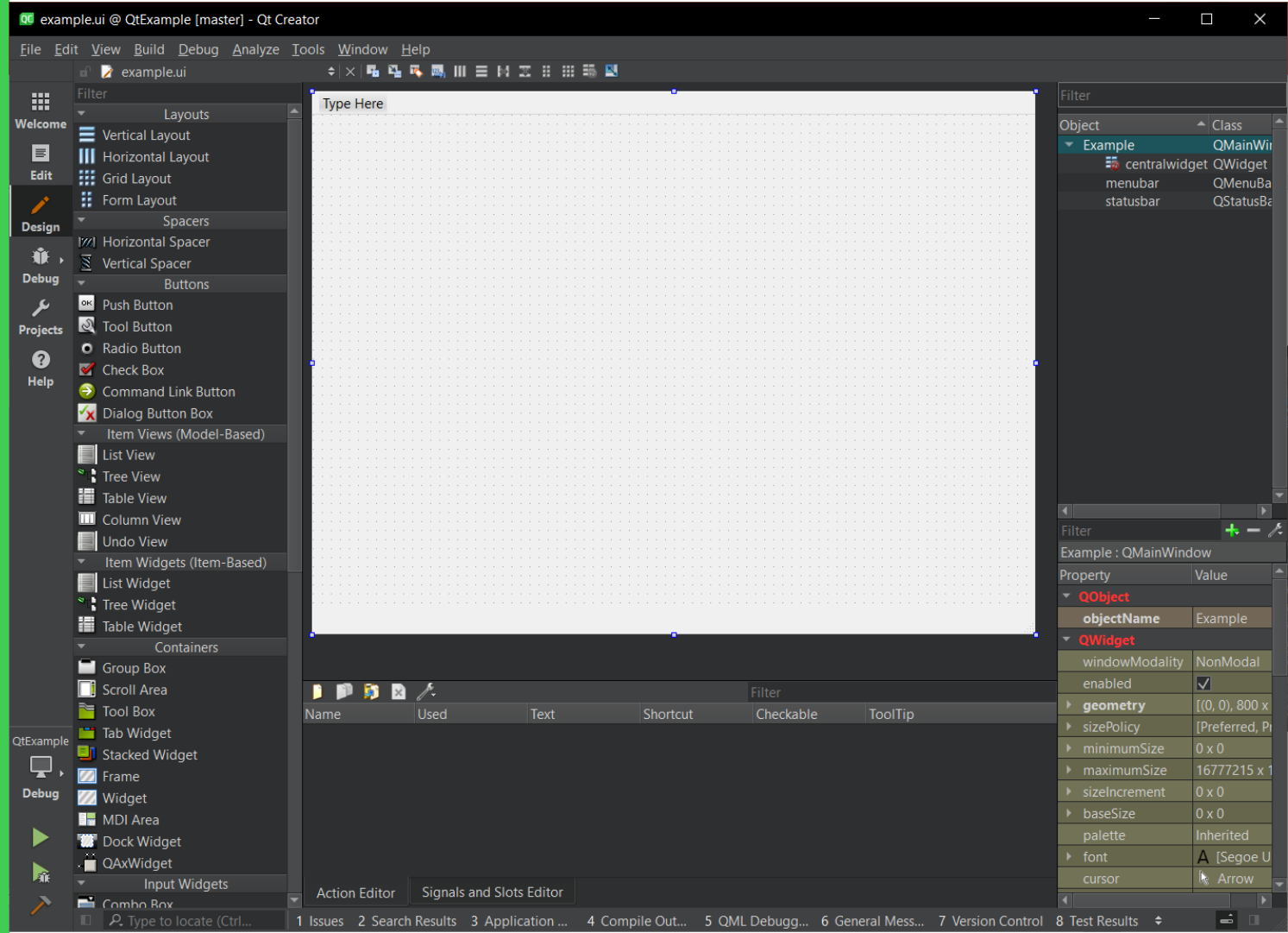
Qt Creator

- Implement in .cpp & .h
- Design UI in .ui



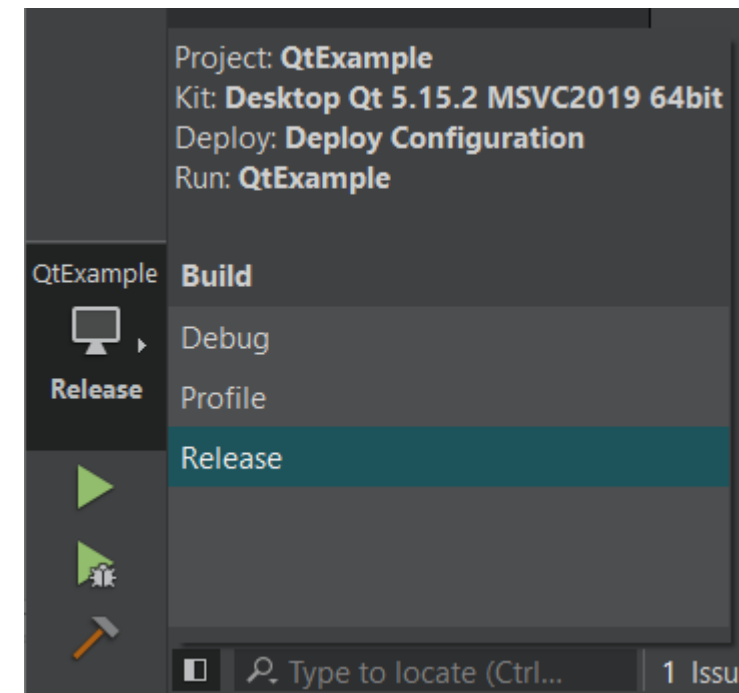
Qt Creator

- Drag components to use
- Common components:
 - Push Button
 - Check Box
 - Text Edit
 - Label
- Click  to run









Qt Creator

- Release for sharing
- Put all images or files in this directory



er (C:) > QtProject > build-QtExample-Desktop_Qt_5_15_2_MSVC2019_64bit-Release > release

Name	Date modified	Type	Size
 example.obj	3/3/2023 1:17 AM	3D Object	45 KB
 main.obj	3/3/2023 1:17 AM	3D Object	3 KB
 moc_example.cpp	3/3/2023 1:17 AM	CPP File	3 KB
 moc_example.obj	3/3/2023 1:17 AM	3D Object	3 KB
 moc_predefs.h	3/3/2023 1:17 AM	H File	1 KB
 QtExample.exe	3/3/2023 1:17 AM	Application	33 KB

Qt Creator

- Copy following .dll to .exe location:
Qt5Core.dll
Qt5Gui.dll
Qt5Widgets.dll
qwindows.dll (in platforms)

er (C:) > QtProject > build-QtExample-Desktop_Qt_5_15_2_MSVC2019_64bit-Release > release

Name	Date modified	Type	Size
example.obj	3/3/2023 1:17 AM	3D Object	45 KB
main.obj	3/3/2023 1:17 AM	3D Object	3 KB
moc_example.cpp	3/3/2023 1:17 AM	CPP File	3 KB
moc_example.obj	3/3/2023 1:17 AM	3D Object	3 KB
moc_predefs.h	3/3/2023 1:17 AM	H File	1 KB
Qt5Core.dll	11/6/2020 1:29 PM	Application extens...	5,883 KB
Qt5Gui.dll		extens...	6,844 KB
Qt5Widgets.dll	11/6/2020 1:30 PM	Application extens...	5,370 KB
QtExample.exe	3/3/2023 1:17 AM	Application	33 KB
platforms			

They will be found in C:\Qt\5.15.2\msvc2019_64\bin

It will be found in C:\Qt\5.15.2\msvc2019_64\plugins\platforms



er (C:) > QtProject > build-QtExample-Desktop_Qt_5_15_2_MSVC2019_64bit-Release > release > platforms

Name	Date modified	Type	Size
qwindows.dll	11/6/2020 1:30 PM	Application extens...	1,443 KB

Qt Creator

- Better UI (optional)
- Copy following .dll to .exe location:
qwindowsvistastyle.dll (in styles)

er (C:) > QtProject > build-QtExample-Desktop_Qt_5_15_2_MSVC2019_64bit-Release > release

Name	Date modified	Type	Size
platforms	3/3/2023 1:27 AM	File folder	
styles			
example.obj	3/3/2023 1:17 AM	3D Object	45 KB
main.obj	3/3/2023 1:17 AM	3D Object	3 KB
moc_example.cpp	3/3/2023 1:17 AM	CPP File	3 KB
moc_example.obj	3/3/2023 1:17 AM	3D Object	3 KB
moc_predefs.h	3/3/2023 1:17 AM	H File	1 KB
Qt5Core.dll	11/6/2020 1:29 PM	Application extens...	5,883 KB
Qt5Gui.dll	11/6/2020 1:29 PM	Application extens...	6,844 KB
Qt5Widgets.dll	11/6/2020 1:30 PM	Application extens...	5,370 KB
QtExample.exe	3/3/2023 1:17 AM	Application	33 KB

It will be found in C:\Qt\5.15.2\msvc2019_64\plugins\styles

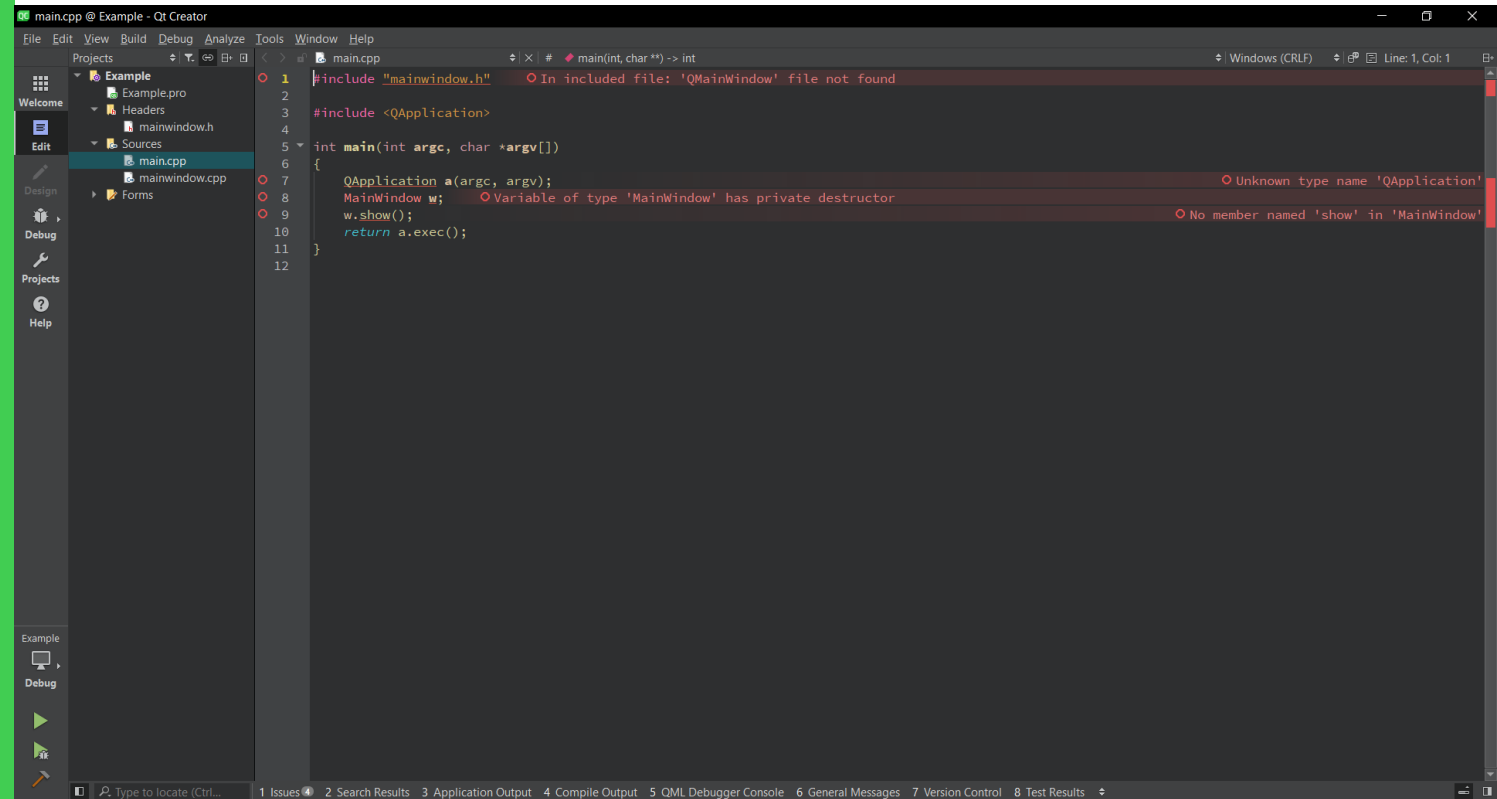


er (C:) > QtProject > build-QtExample-Desktop_Qt_5_15_2_MSVC2019_64bit-Release > release > styles

Name	Date modified	Type	Size
qwindowsvistastyle.dll	11/6/2020 1:30 PM	Application extens...	141 KB

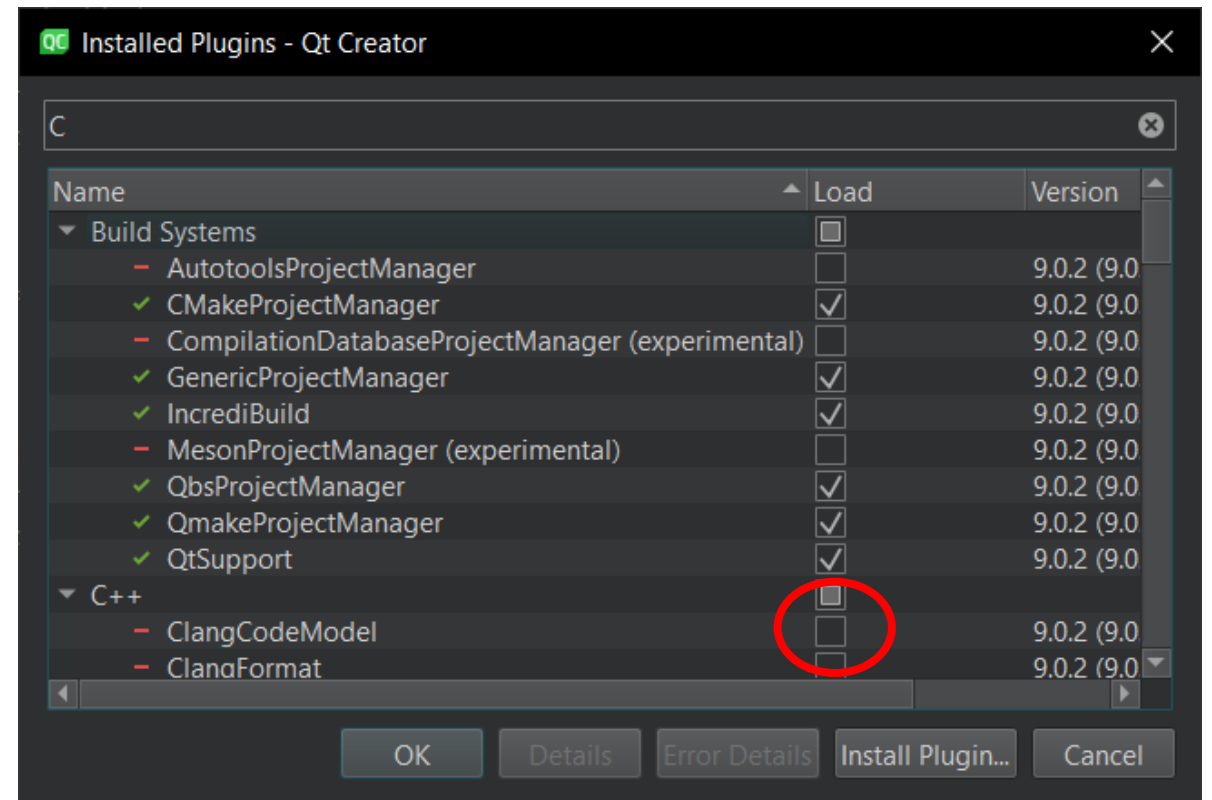
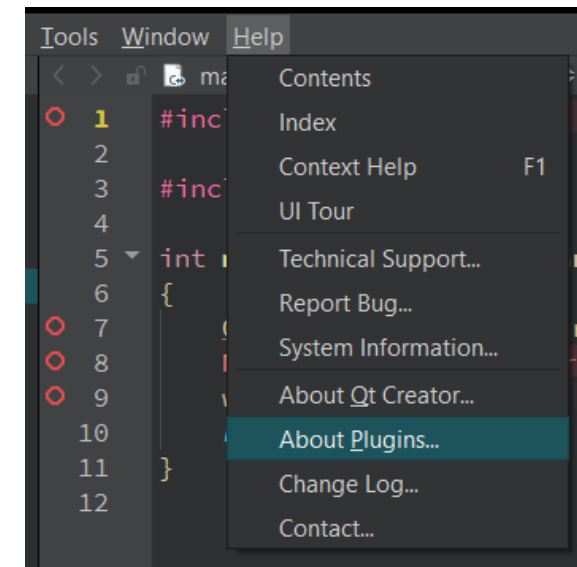
Troubleshoot (1)

- Problem:
'QMainWindow' file not found



Troubleshoot (1)

- Solution: Uncheck ClangCodeModel



Troubleshoot (2)

- Problem: Can't run
- Solution: Please **DO NOT** include Chinese in the path

```
15:41:49: The process "C:\Qt\5.15.2\msvc2019_64\bin\qmake.exe" exited with code 2.  
Error while building/deploying project Example (kit: Desktop Qt 5.15.2 MSVC2019 64bit)  
When executing step "qmake"  
15:41:49: Elapsed time: 00:00.
```


Troubleshoot (3)

- Problem: .exe can't find .dll
- Solution: Check Qt5Core.dll, Qt5Gui.dll, Qt5Widgets.dll, and qwindows.dll are in the .exe directory



Visual Studio

- Visual Studio 2019

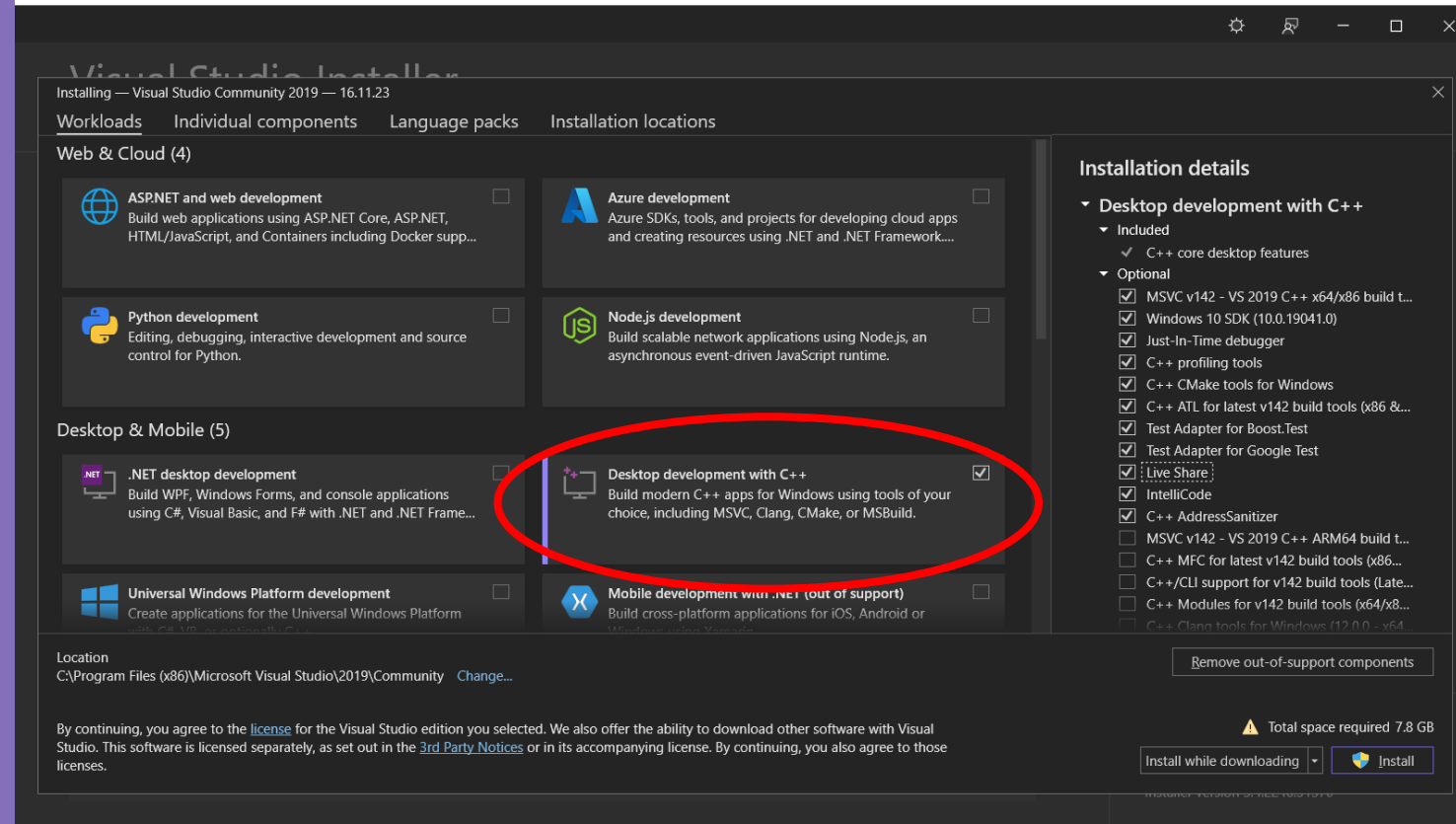
The image shows the Visual Studio 2019 splash screen. It features a dark background with a pattern of small, glowing blue dots that form a series of concentric, curved lines, creating a sense of depth and movement. The dots are more densely packed in some areas, forming a bright, glowing shape that resembles a stylized 'V' or a mountain peak. The text 'Visual Studio' is centered in the middle of the screen in a white, sans-serif font.

Visual Studio

2019

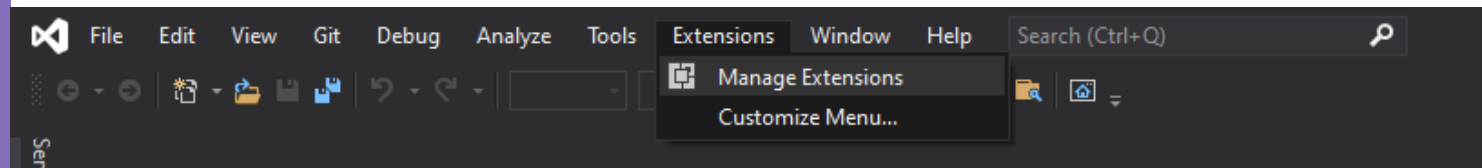
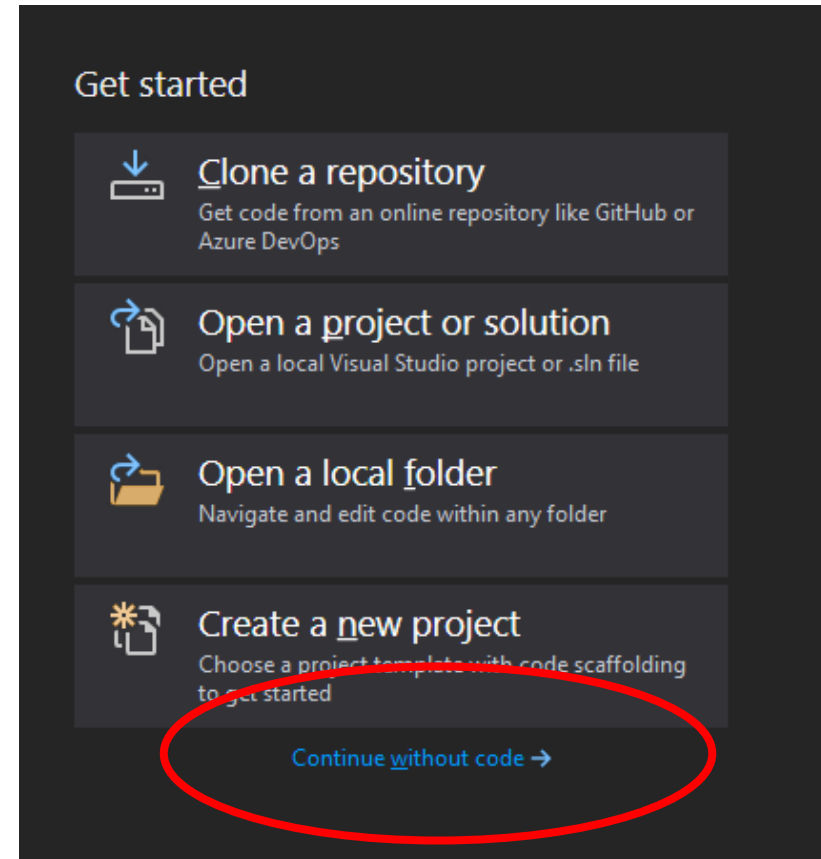
Visual Studio

- Community 2019
- Desktop development with C++



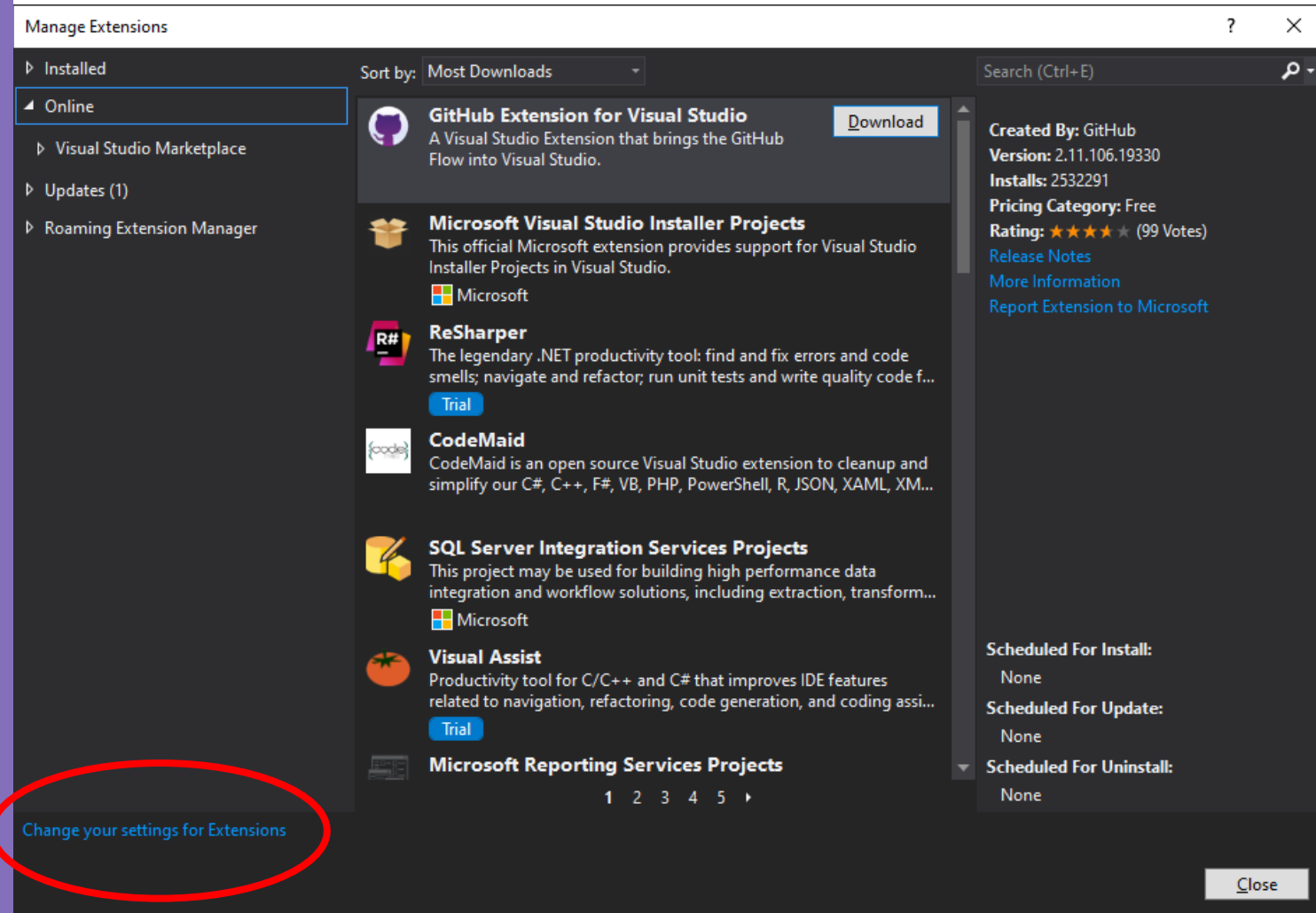
Visual Studio

- Launch Visual Studio
- Continue without code
- Click Manage Extensions



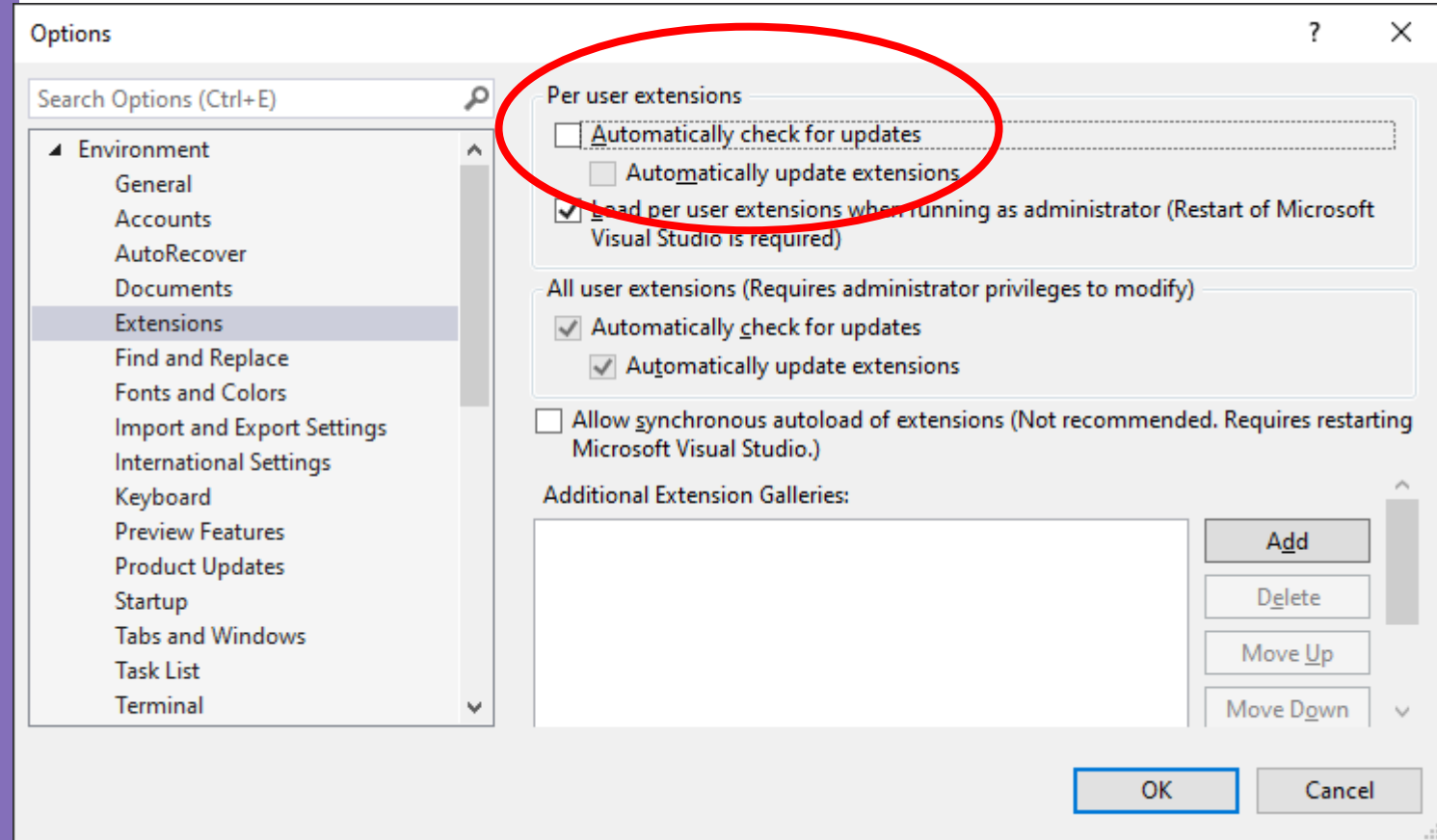
Visual Studio

- Change your settings for Extensions



Visual Studio

- Unchecked
Automatically updates







VS Qt VSIX

- MSVC 2019-2.4.3
- [Link](#)

Qt Downloads

[Qt Home](#) [Bug Tracker](#) [Code Review](#) [Planet Qt](#) [Get Qt Extensions](#)

Name	Last modified	Size	Metadata
↑ Parent Directory		-	
 qt-vsaddin-msvc2019-2.4.3.vsix	07-Jan-2020 13:54	12M	Details
 qt-vsaddin-msvc2017-2.4.3.vsix	07-Jan-2020 13:54	12M	Details
 qt-vsaddin-msvc2015-2.4.3.vsix	07-Jan-2020 13:54	12M	Details
 md5sums.txt	07-Jan-2020 13:55	195	Details

For Qt Downloads, please visit qt.io/download

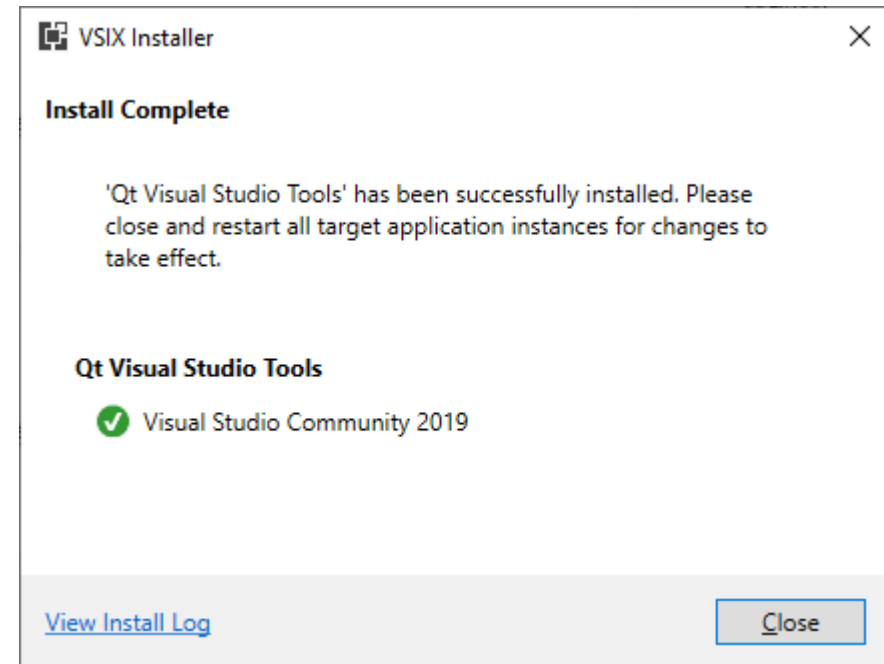
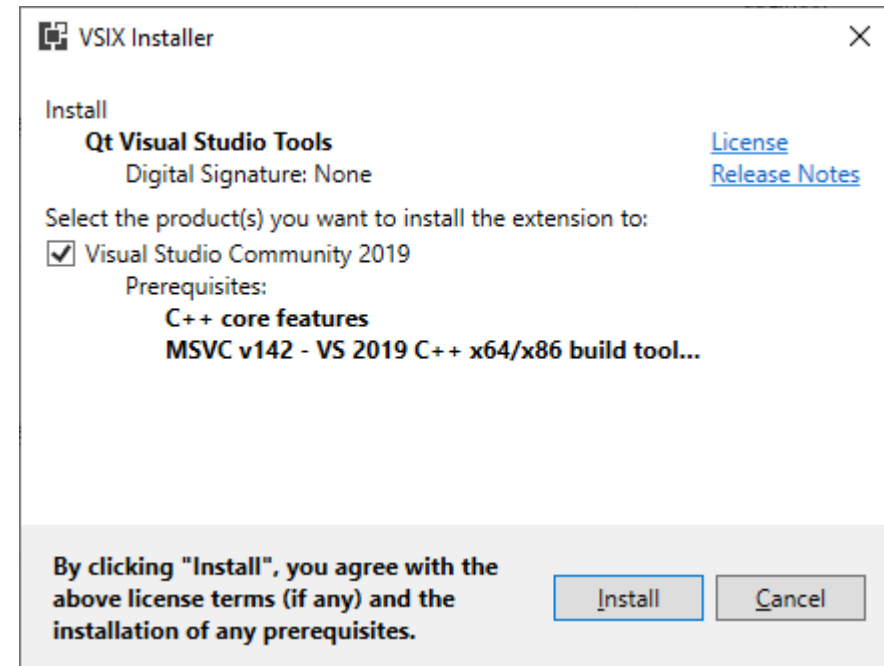
QtÅ® and the Qt logo is a registered trade mark of The Qt Company Ltd and is used pursuant to a license from The Qt Company Ltd.
All other trademarks are property of their respective owners.

The Qt Company Ltd, Bertel Jungin aukio D3A, 02600 Espoo, Finland. Org. Nr. 2637805-2

[List of official Qt-project mirrors](#)

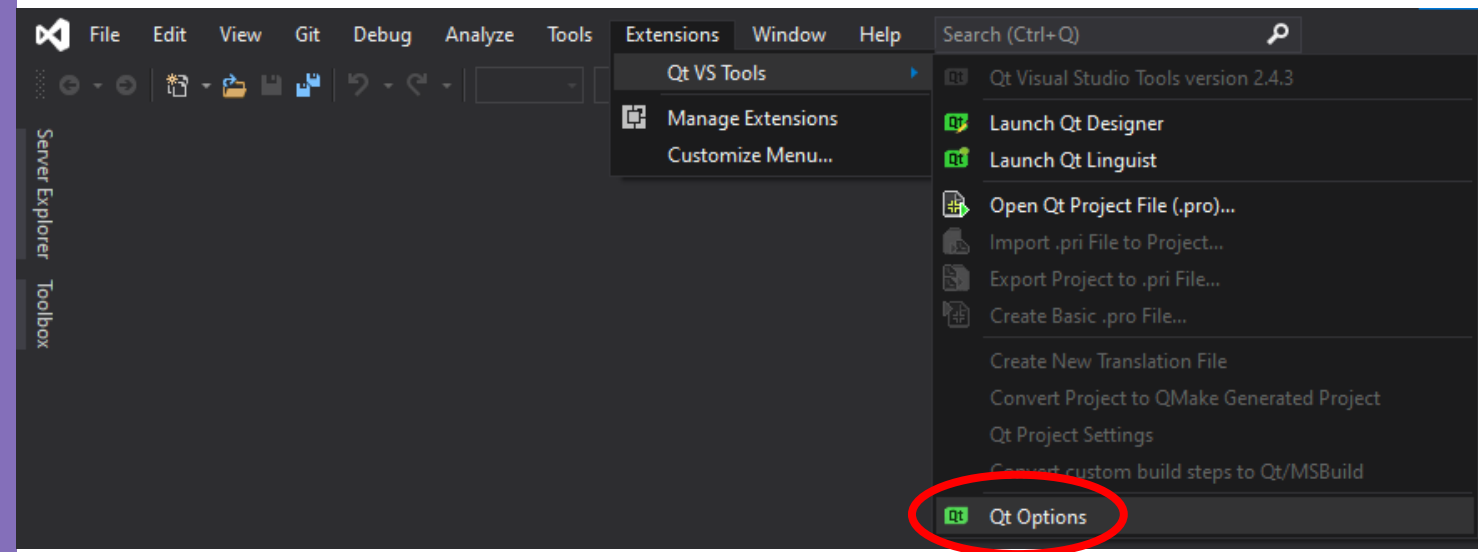
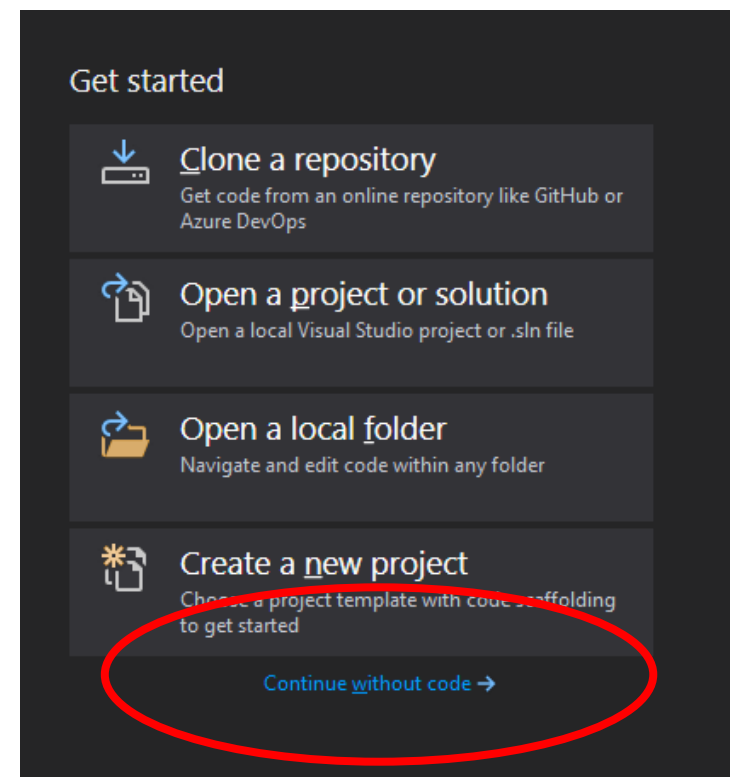
VS Qt VSIX

- Close all VS windows
- Install Qt VSIX



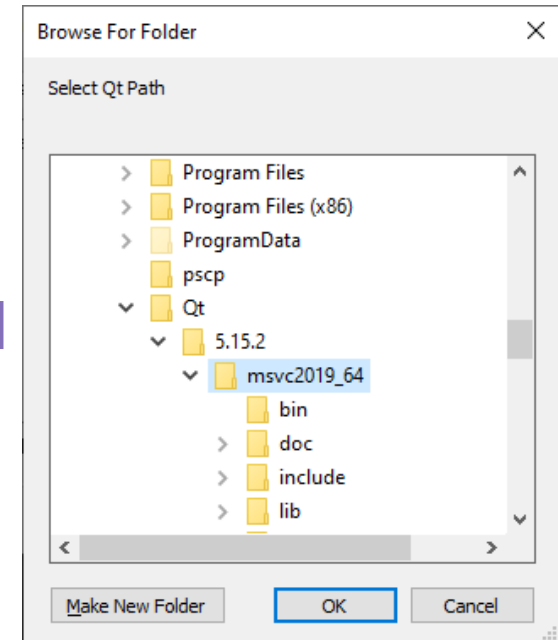
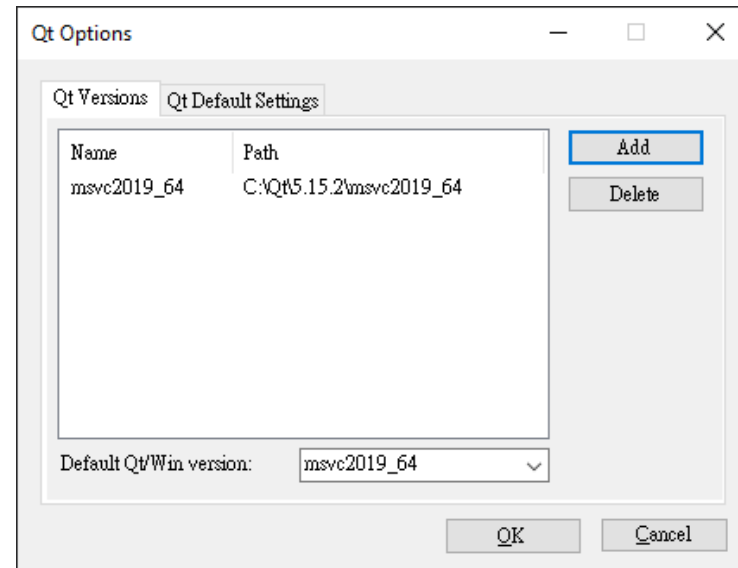
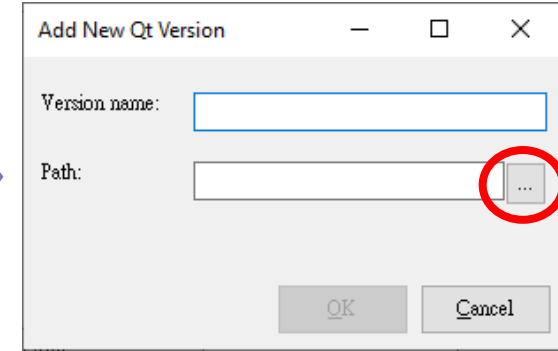
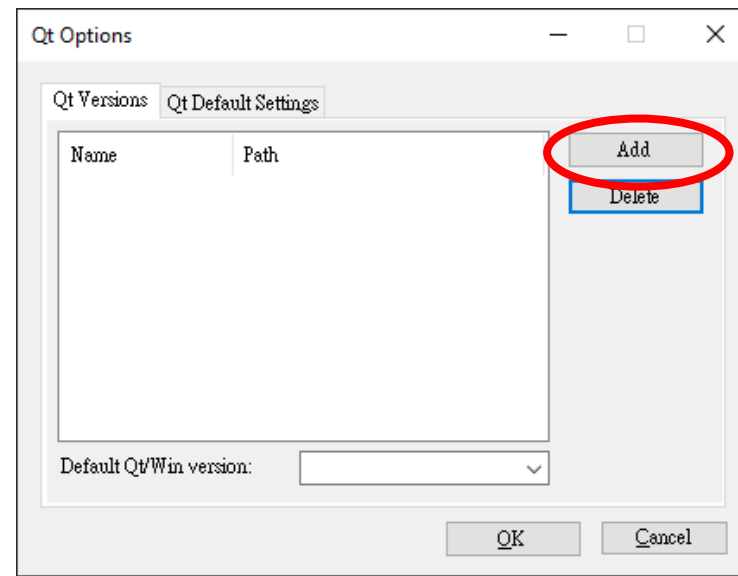
VS Qt VSIX

- Launch Visual Studio
- Continue without code
- Click Qt Options



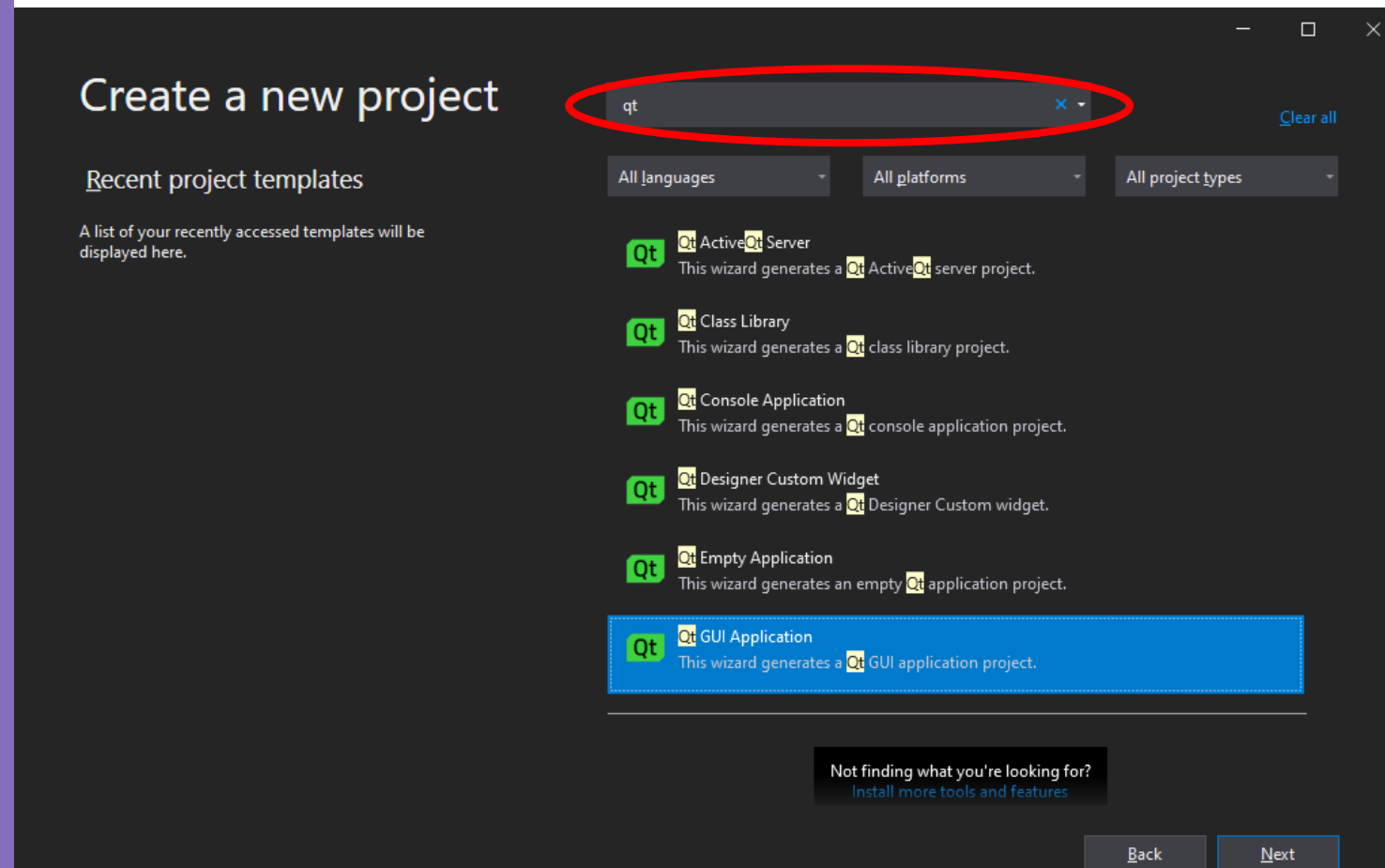
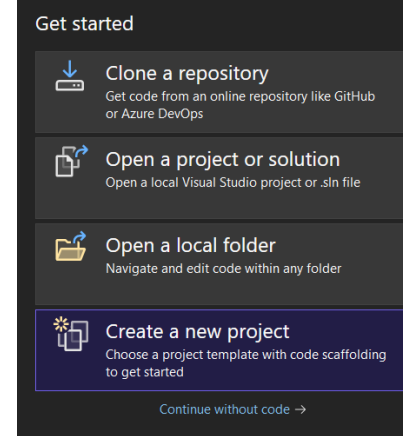
VS Qt VSIX

- Add Qt path



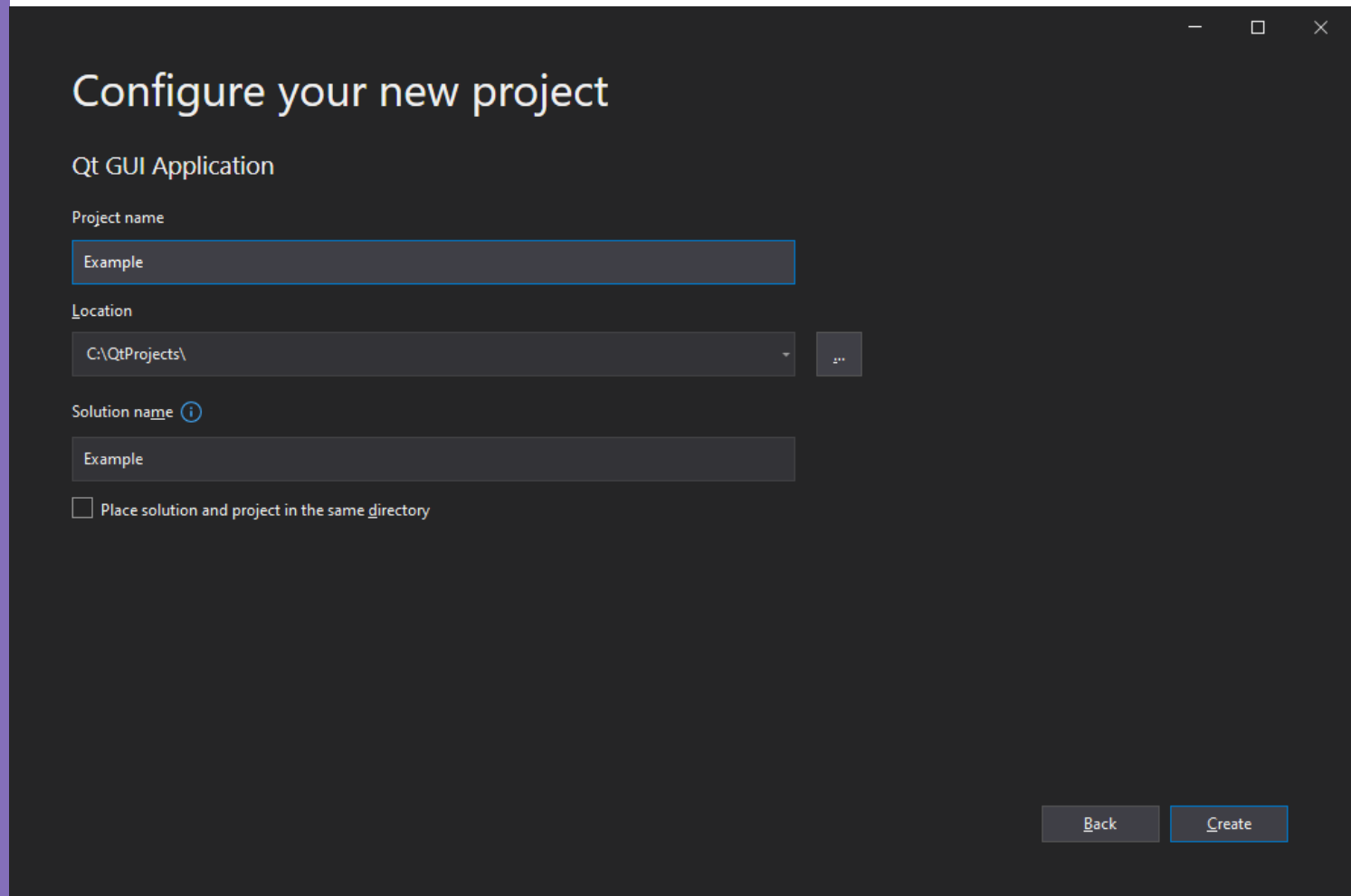
Initialization

- Close Visual Studio
- Launch Visual Studio
- Create a new project
- Type Qt in search bar
- Choose Qt GUI Application
- Next



Initialization

- Type your project name
- Create



The screenshot shows a 'Configure your new project' dialog box with a dark theme. At the top, it says 'Qt GUI Application'. Below this, there are three main sections: 'Project name' with a text field containing 'Example'; 'Location' with a text field containing 'C:\QtProjects\' and a browse button; and 'Solution name' with a text field containing 'Example' and an information icon. At the bottom, there is a checkbox labeled 'Place solution and project in the same directory' which is currently unchecked. In the bottom right corner, there are two buttons: 'Back' and 'Create'.

Configure your new project

Qt GUI Application

Project name

Example

Location

C:\QtProjects\

Solution name ⓘ

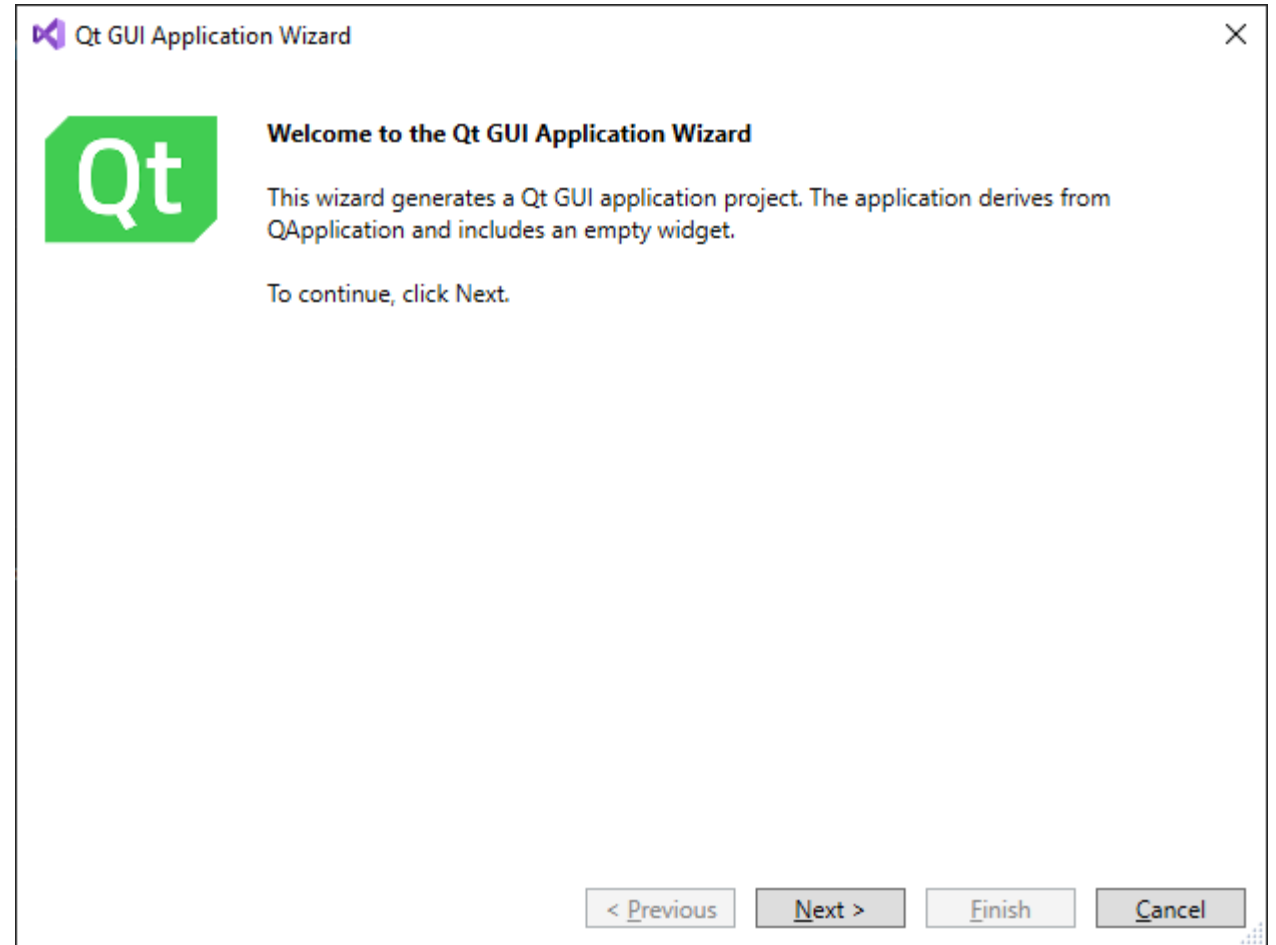
Example

☐ Place solution and project in the same directory

Back Create

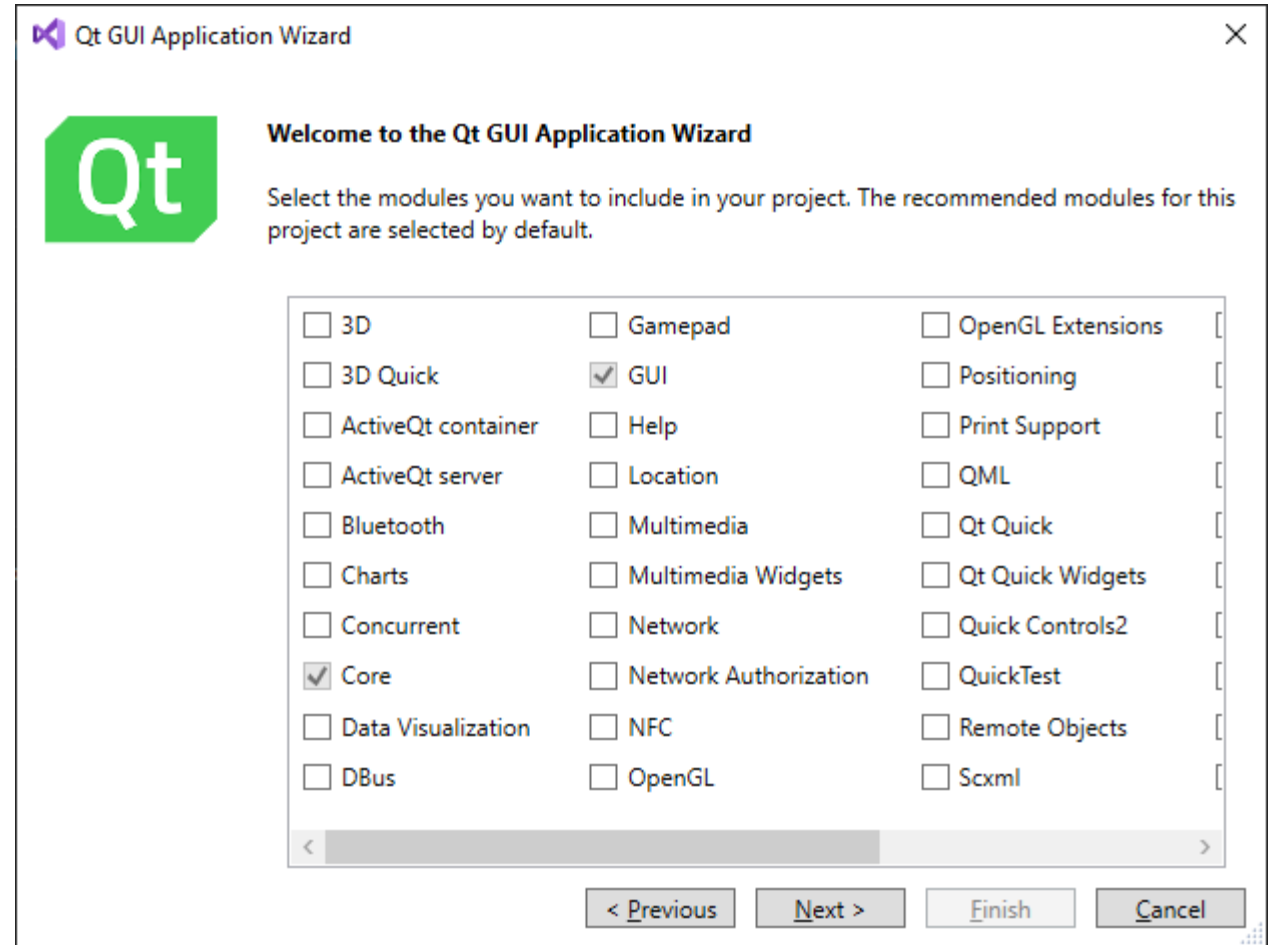
Initialization

- Next



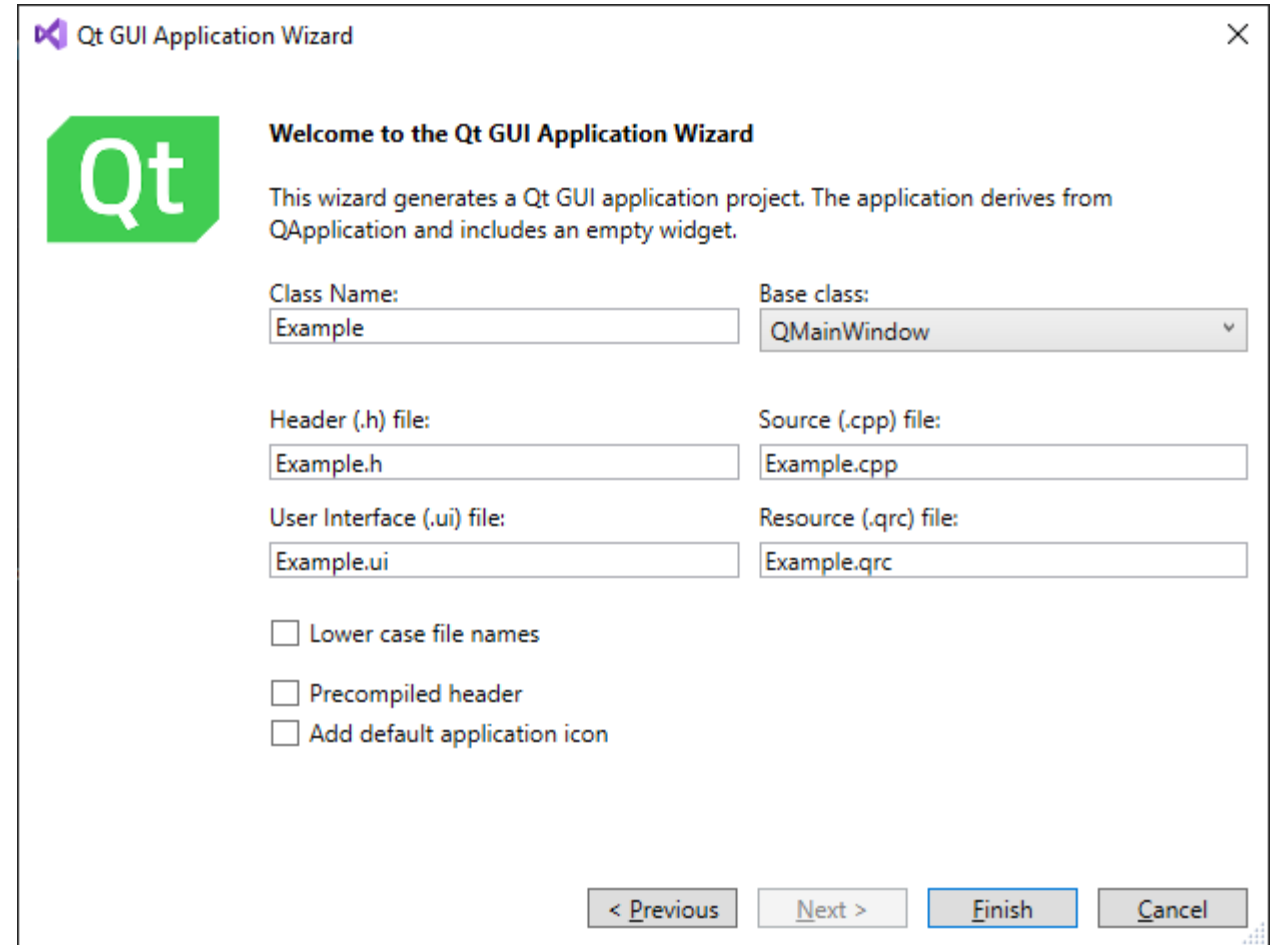
Initialization

- Use default



Initialization

- Finish



The image shows the 'Qt GUI Application Wizard' dialog box. It has a title bar with the Qt logo and the text 'Qt GUI Application Wizard'. The main area contains a green 'Qt' logo on the left and text on the right that says 'Welcome to the Qt GUI Application Wizard' and 'This wizard generates a Qt GUI application project. The application derives from QApplication and includes an empty widget.' Below this, there are several input fields and checkboxes. The 'Class Name' field contains 'Example'. The 'Base class' dropdown menu is set to 'QMainWindow'. The 'Header (.h) file' field contains 'Example.h'. The 'Source (.cpp) file' field contains 'Example.cpp'. The 'User Interface (.ui) file' field contains 'Example.ui'. The 'Resource (.qrc) file' field contains 'Example.qrc'. At the bottom, there are three checkboxes: 'Lower case file names', 'Precompiled header', and 'Add default application icon', all of which are unchecked. At the very bottom, there are four buttons: '< Previous', 'Next >', 'Finish', and 'Cancel'. The 'Finish' button is highlighted with a blue border.

Qt GUI Application Wizard

Welcome to the Qt GUI Application Wizard

This wizard generates a Qt GUI application project. The application derives from QApplication and includes an empty widget.

Class Name:

Base class:

Header (.h) file:

Source (.cpp) file:

User Interface (.ui) file:

Resource (.qrc) file:

☐ Lower case file names

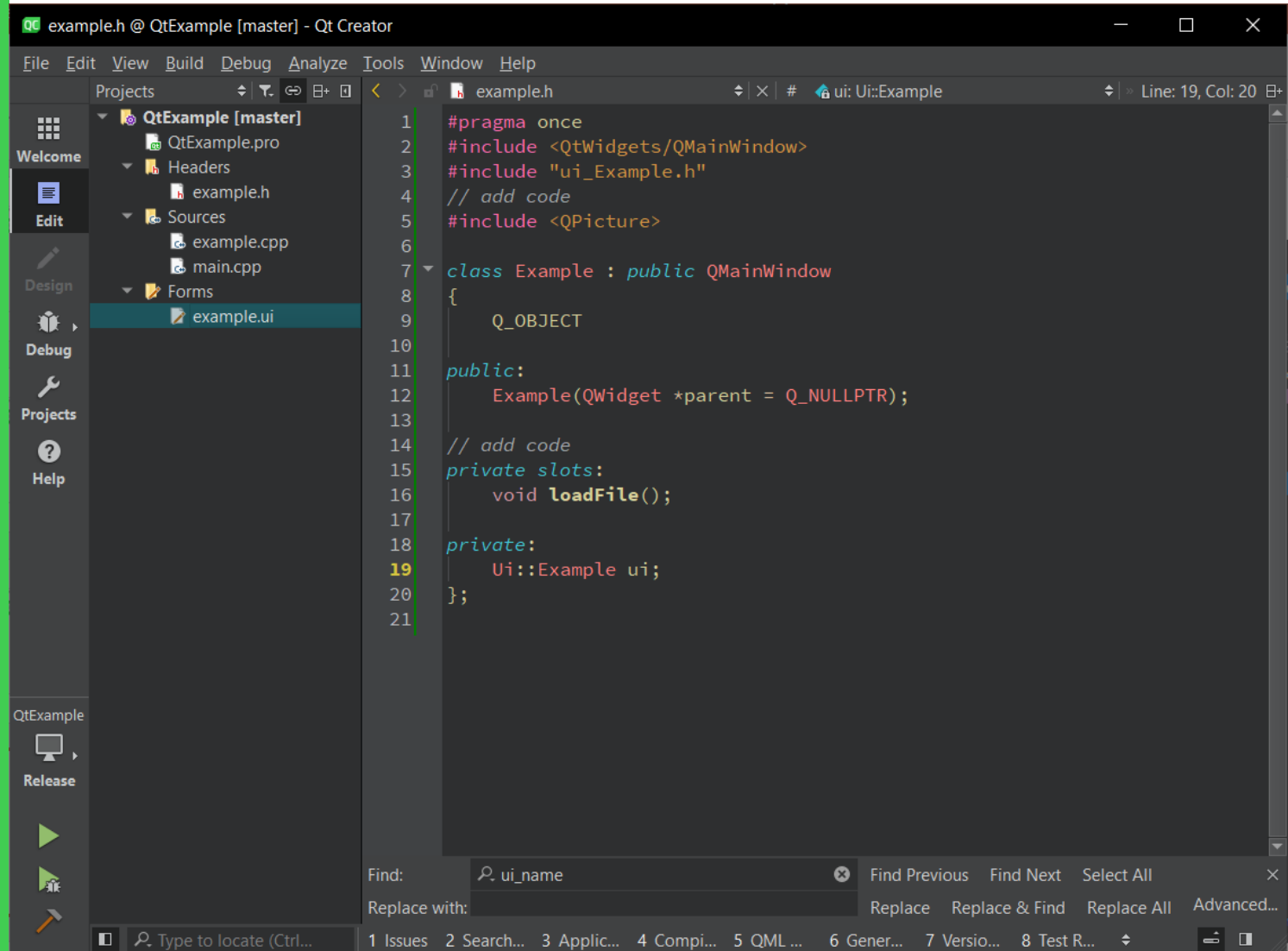
☐ Precompiled header

☐ Add default application icon

< Previous Next > **Finish** Cancel

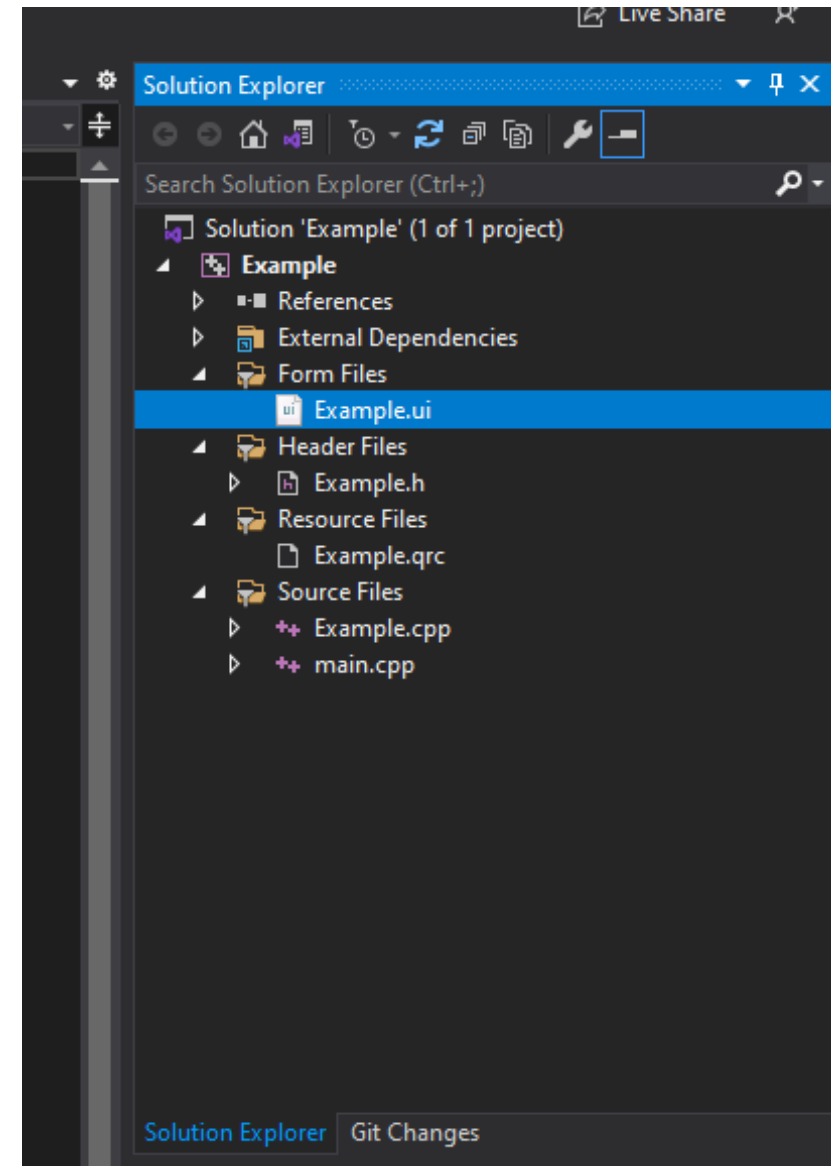
Introduction

- Double click *.ui to open Qt Designer (Qt Creator Ver.)



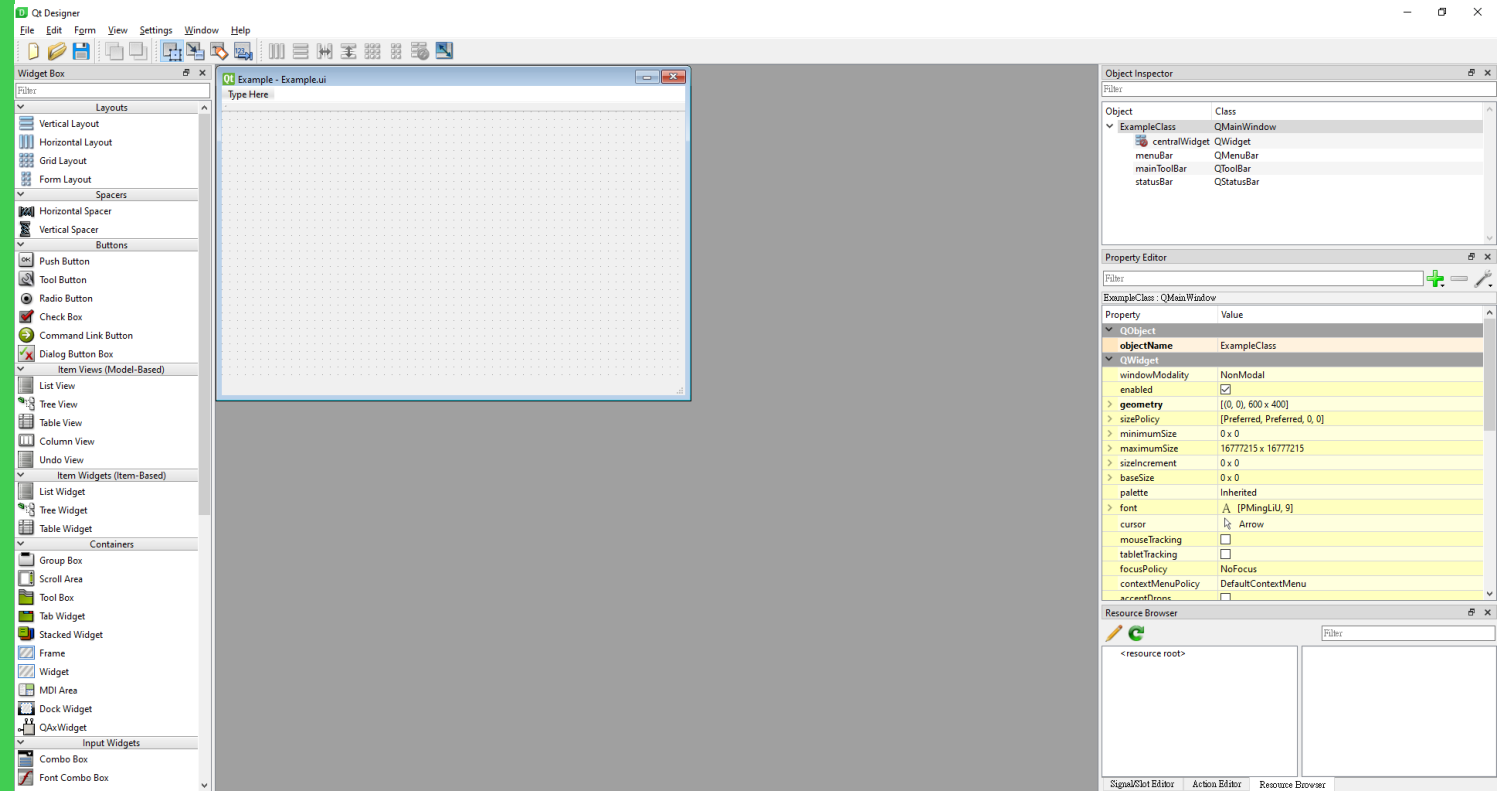
Introduction

- Double click *.ui to open Qt Designer (Visual Studio Ver.)



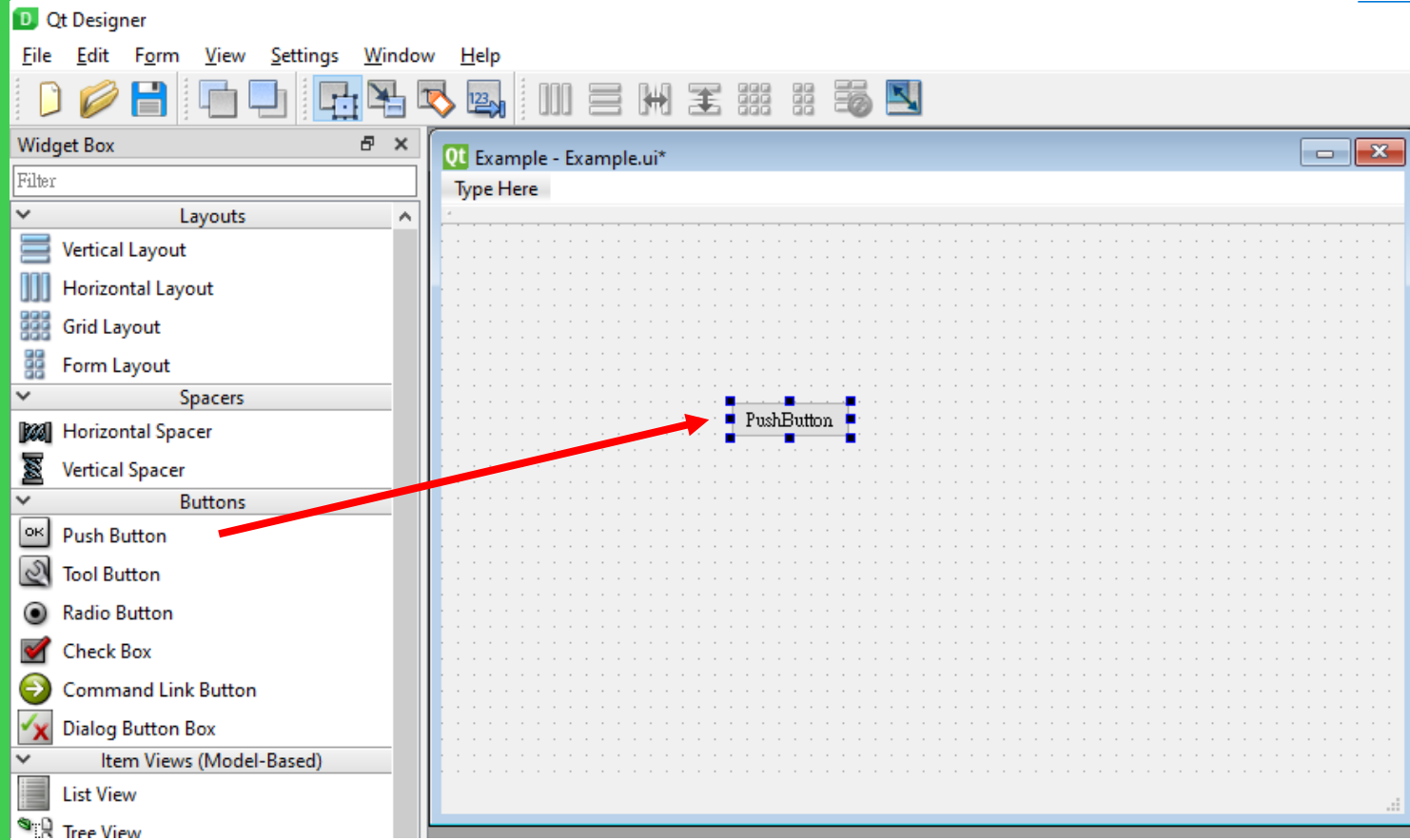
Introduction

- Qt Designer for GUI



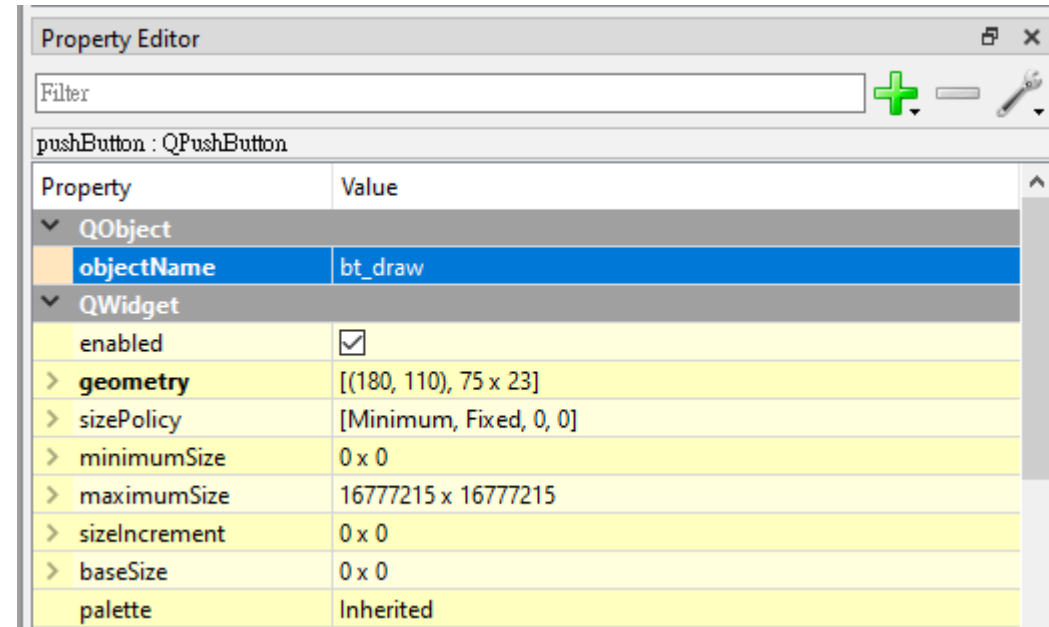
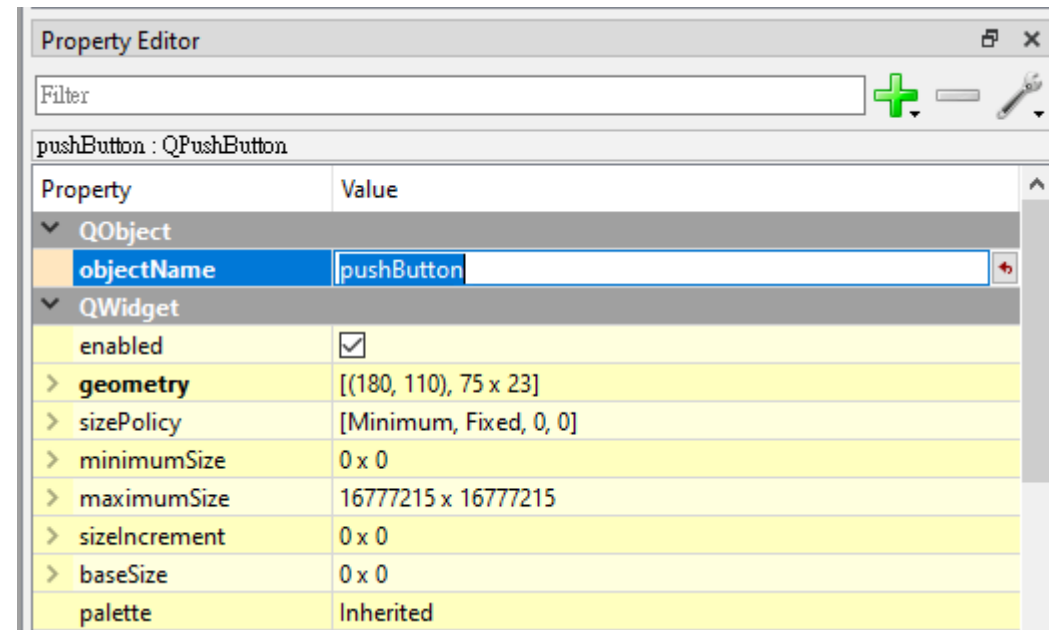
Practice

- Drag a Push Button to the window



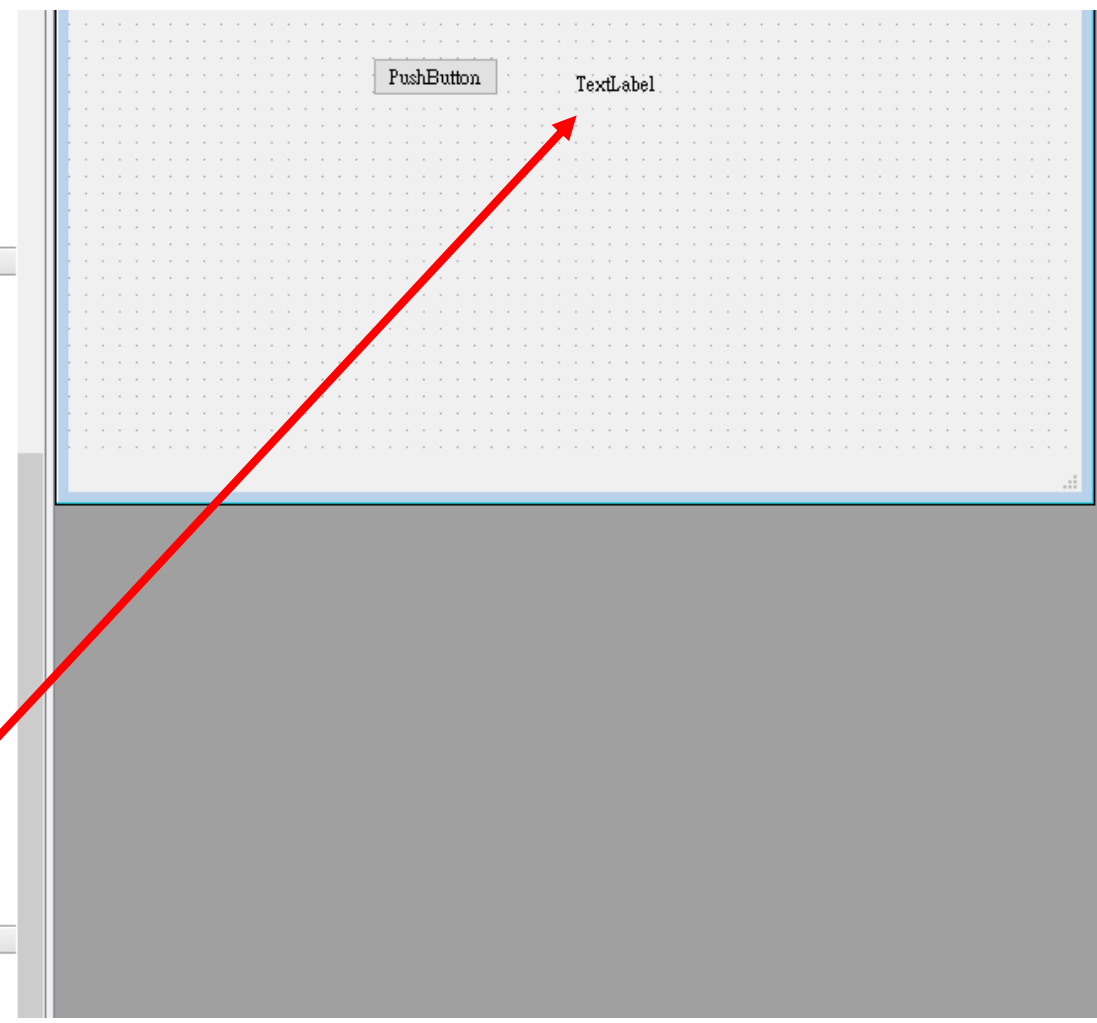
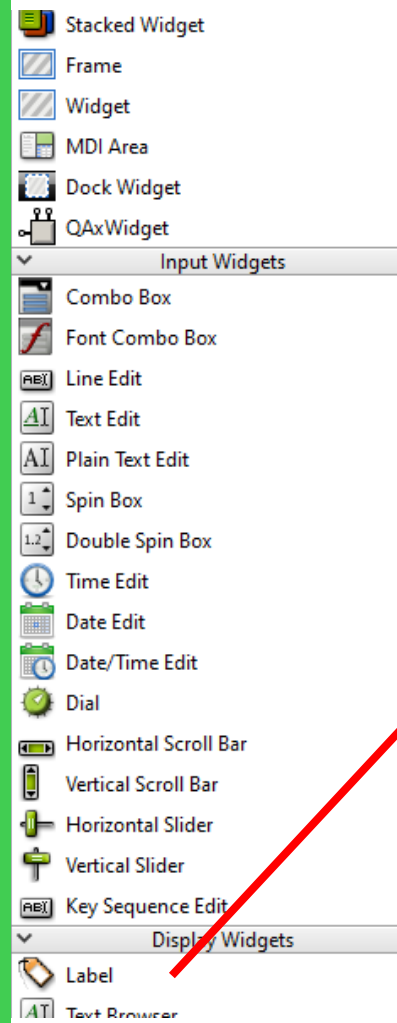
Practice

- Change the object name



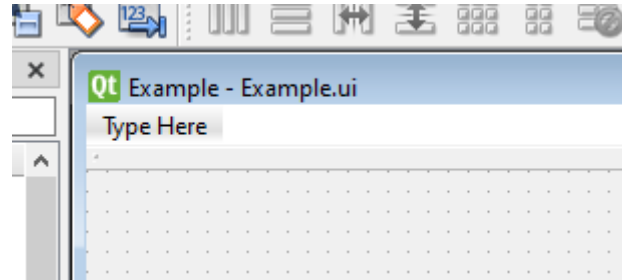
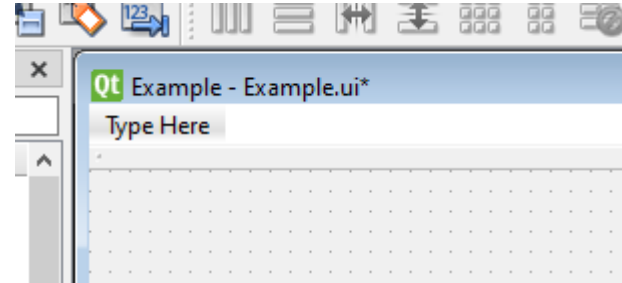
Practice

- Drag a Label to the window



Practice

- Ctrl + S to save the layout



Practice

- Example.h

```
1)  #pragma once
2)  #include <QtWidgets/QMainWindow>
3)  #include "ui_Example.h"
4)  // add code
5)  #include <QPicture>

6)  class Example : public QMainWindow
7)  {
8)      Q_OBJECT

9)  public:
10)     Example(QWidget *parent = Q_NULLPTR);

11) // add code
12) private slots:
13)     void loadFile();

14) private:
15)     Ui::Example ui;
16) };
```

Practice

- Example.cpp
- Load any picture (e.g. logo.png)
- You should put your images into release directory

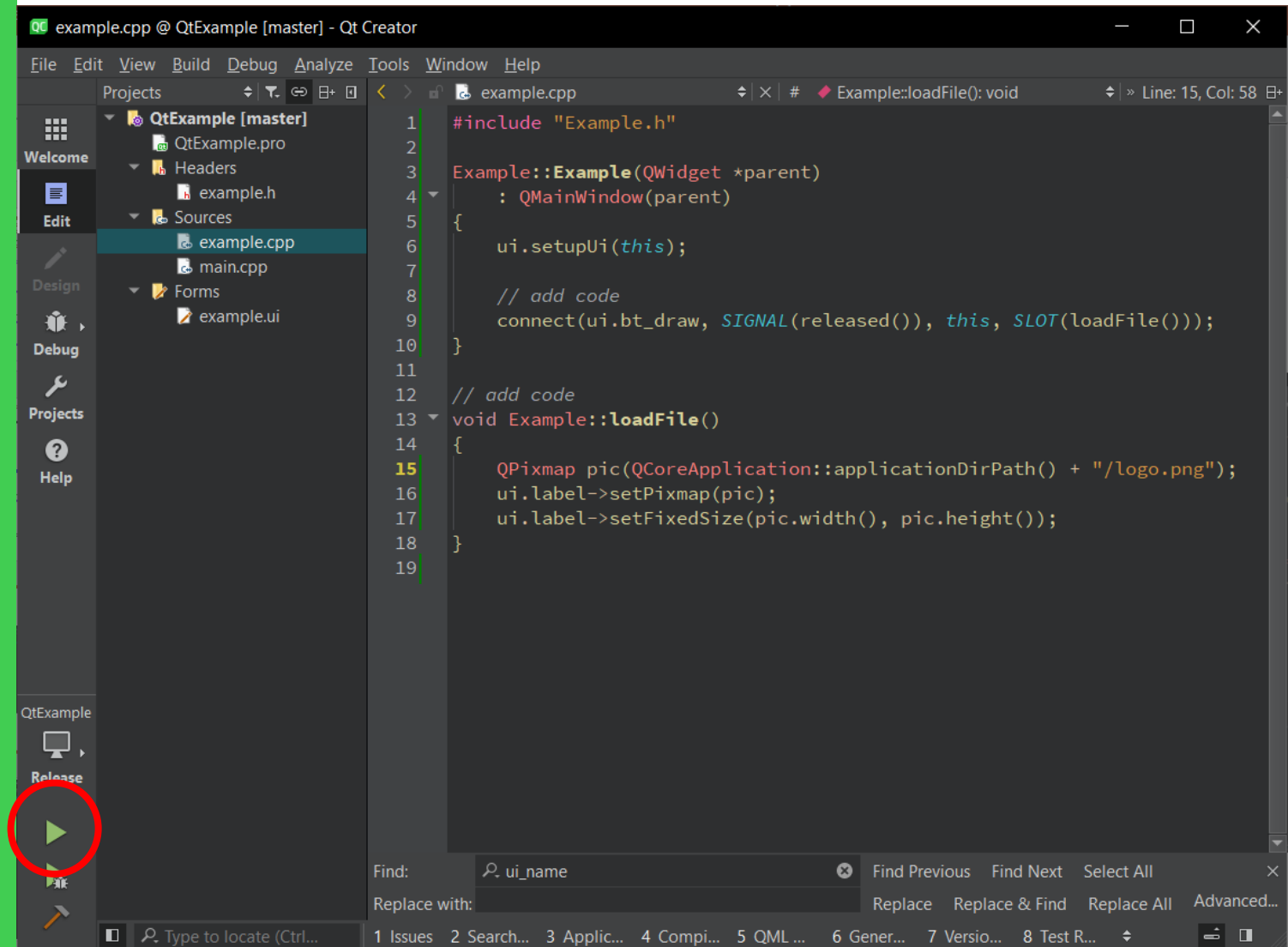
```
1)  #include "Example.h"

2)  Example::Example(QWidget *parent)
3)      : QMainWindow(parent)
4)  {
5)      ui.setupUi(this);
6)
7)      // add code
8)      connect(ui.bt_draw, SIGNAL(released()),
9)             this, SLOT(loadFile()));
10) }

10) // add code
11) void Example::loadFile()
12) {
13)     QPixmap pic("logo.png");
14)     ui.label->setPixmap(pic);
15)     ui.label->setFixedSize(pic.width(),
16)                            pic.height());
16) }
```

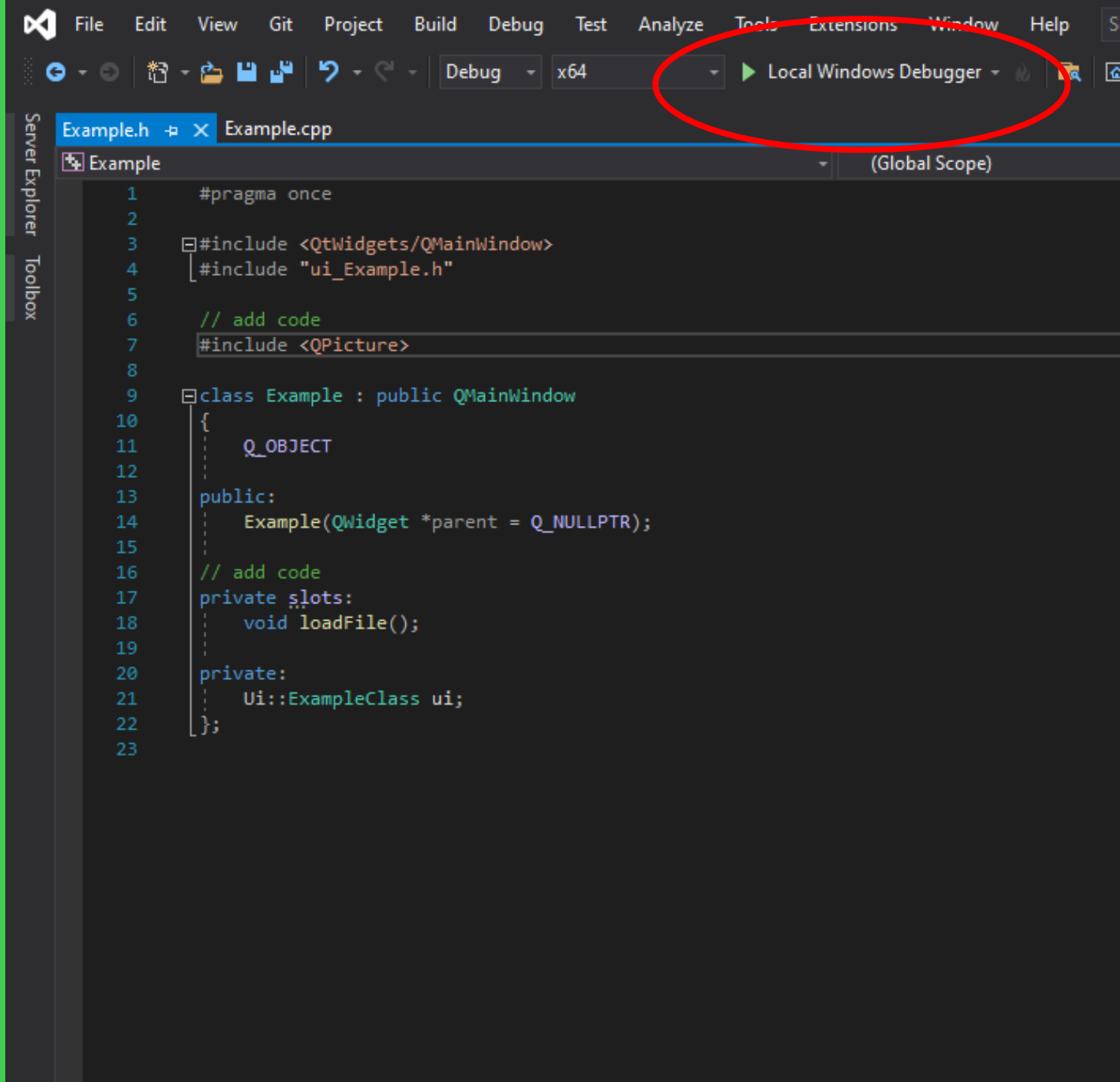
Practice

- Run (Qt Creator Ver.)
- Add
QCoreApplication::applicationDirPath()
before the file



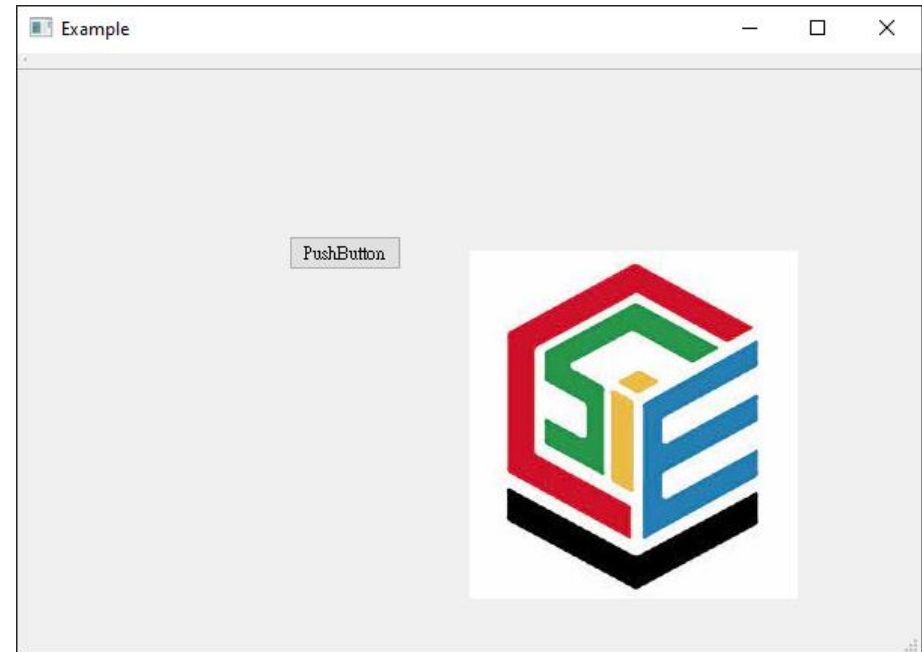
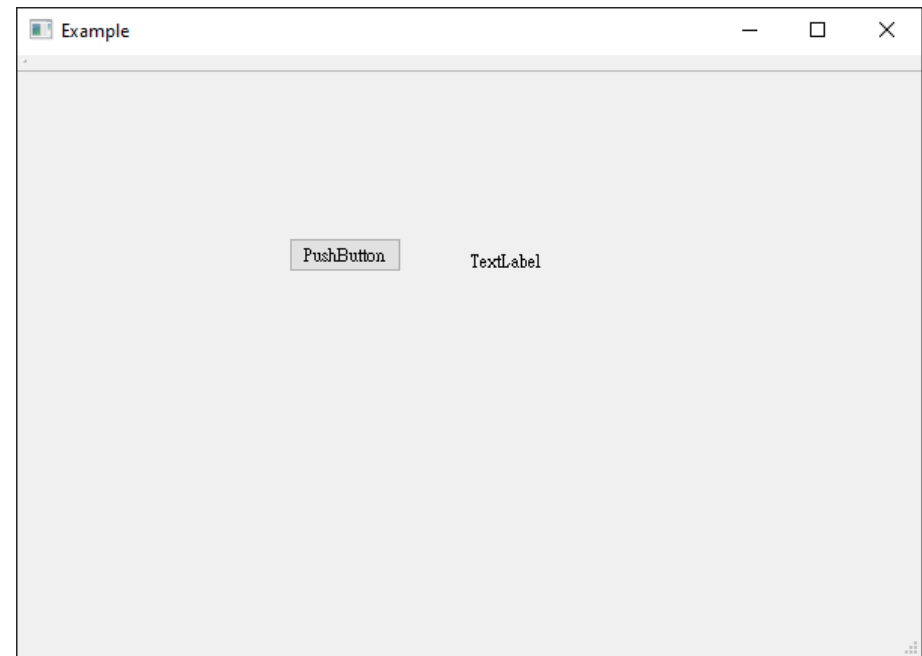
Practice

- Run (Visual Studio Ver.)



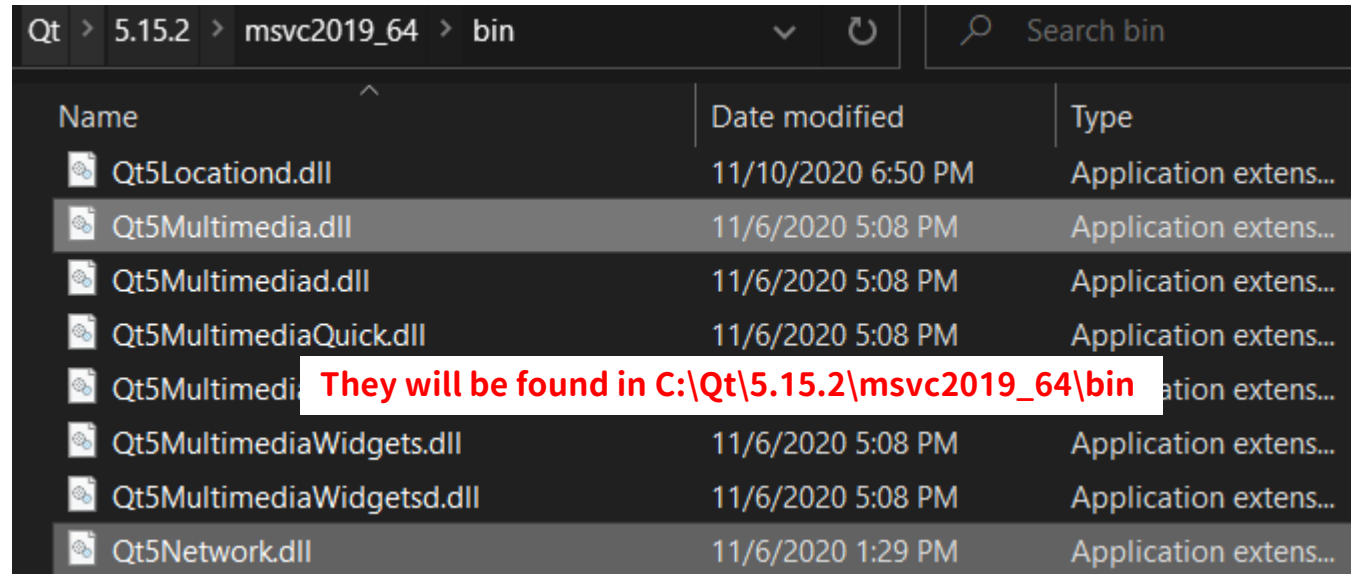
Practice

- Click PushButton
- Display logo.png



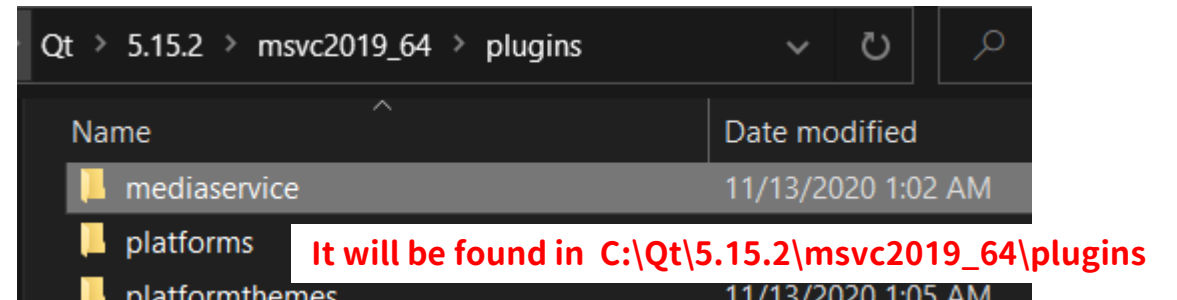
QMediaPlayer

- Copy following .dll to .exe location:
Qt5Multimedia.dll
Qt5Network.dll
- Copy directory
“mediaservice” to .exe location



Qt > 5.15.2 > msvc2019_64 > bin			Search bin
Name	Date modified	Type	
Qt5Locationd.dll	11/10/2020 6:50 PM	Application extens...	
Qt5Multimedia.dll	11/6/2020 5:08 PM	Application extens...	
Qt5Multimediad.dll	11/6/2020 5:08 PM	Application extens...	
Qt5MultimediaQuick.dll	11/6/2020 5:08 PM	Application extens...	
Qt5MultimediaWidgets.dll	11/6/2020 5:08 PM	Application extens...	
Qt5MultimediaWidgets.d.dll	11/6/2020 5:08 PM	Application extens...	
Qt5Network.dll	11/6/2020 1:29 PM	Application extens...	

They will be found in C:\Qt\5.15.2\msvc2019_64\bin

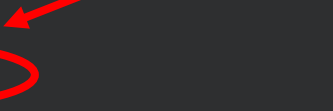


Qt > 5.15.2 > msvc2019_64 > plugins		
Name	Date modified	
mediaservice	11/13/2020 1:02 AM	
platforms		
platformthemes	11/13/2020 1:05 AM	

It will be found in C:\Qt\5.15.2\msvc2019_64\plugins

QMediaPlayer

- Add multimedia into Example.pro



```
1 QT      += core gui \
2         |   |   |   | multimedia
3
4 greaterThan(QT_MAJOR_VERSION, 4): QT += widgets
5
6 CONFIG += c++17
7
8 # You can make your code fail to compile if it uses deprecated APIs.
9 # In order to do so, uncomment the following line.
10 #DEFINES += QT_DISABLE_DEPRECATED_BEFORE=0x060000    # disable support
11
12 SOURCES += \
13     main.cpp \
14     PokemonGame.cpp
15
16 HEADERS += \
17     PokemonGame.h
18
19 FORMS += \
20     PokemonGame.ui
21
22 # Default rules for deployment.
23 qnx: target.path = /tmp/${TARGET}/bin
24 else: unix:!android: target.path = /opt/${TARGET}/bin
25 !isEmpty(target.path): INSTALLS += target
26
```

QMediaPlayer

- Example.cpp
- Load any sounds (e.g. battle.mp3)
- You should put your audios into release directory

```
1) #include <QMediaPlayer>
```

```
2) QMediaPlayer* bgMusic = new  
   QMediaPlayer;
```

```
3)     bgMusic->  
       >setMedia(QUrl::fromLocalFil  
       e("Music/battle.mp3"));
```

```
4)     bgMusic->setVolume(100);
```

```
5)     bgMusic->play();
```



Qt Tutorial

Object Oriented Programming