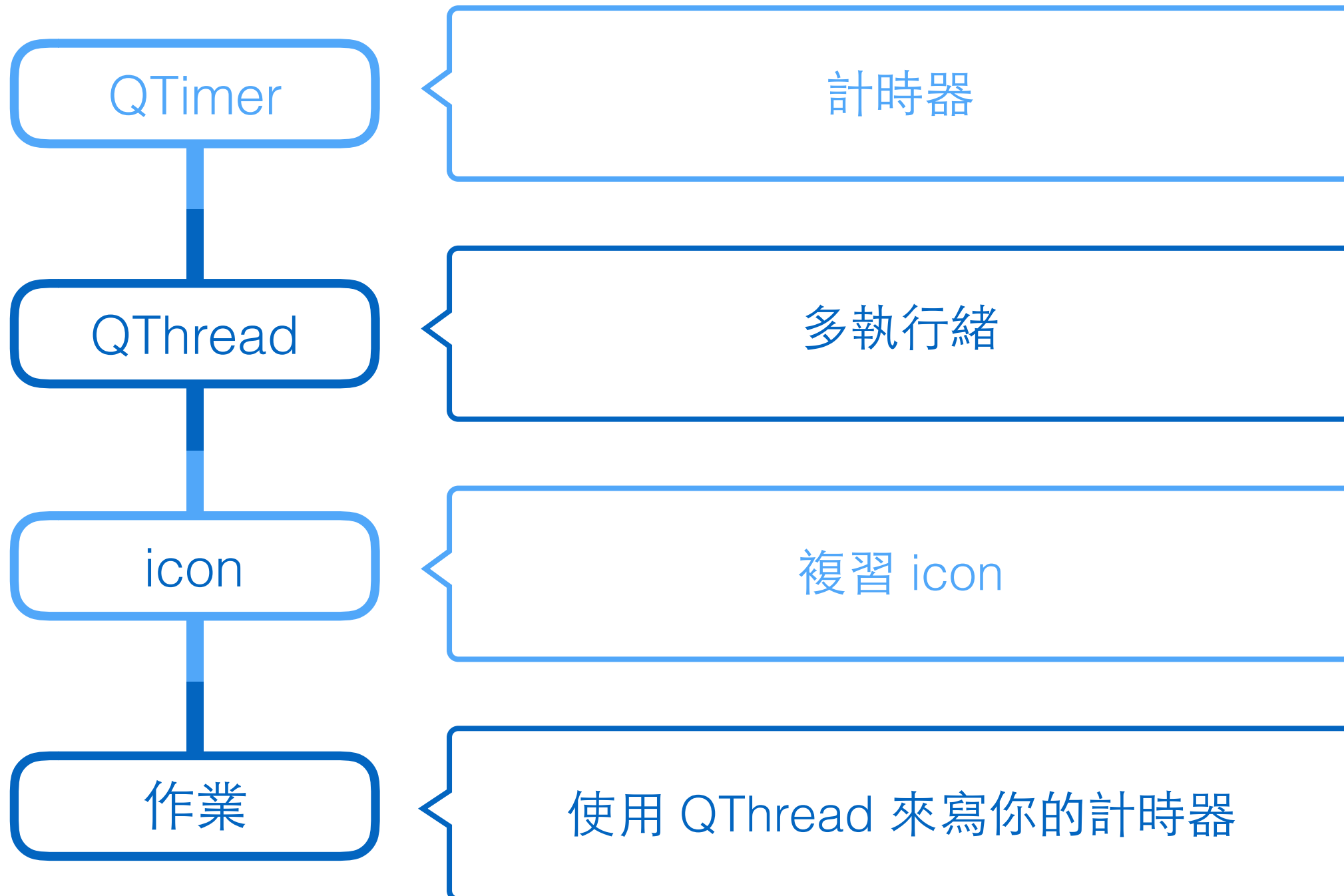


Qt Lesson 5

Lisa Tsai @ 2017 Summer

Schedule



Qt Schedule

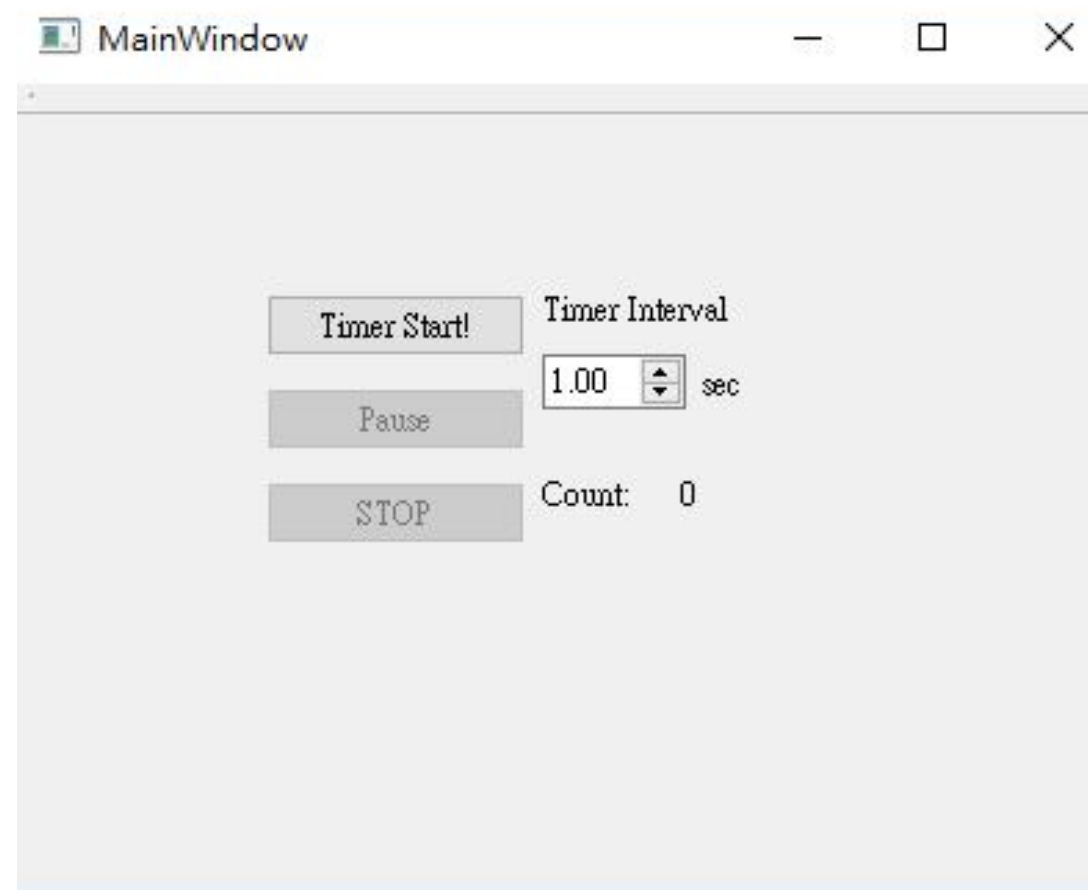
	日期	課程	作業
week 1	7/4	Qt 環境架設 & GUI 簡單介紹	點餐系統
week 2	7/25	Qt 基礎練習 / widget 介紹	會員集點
week 3	8/1	Github / SourceTree	上傳前兩次作業
week 4	8/8	Gitbook 專案管理 / 共同編輯 / 同步 / 版本控制	fork 及修改別人的專題
week 5	8/15	指標、初階物件導向應用	打磚塊
week 6	8/22	進階物件導向程式設計	
week 7	8/29	Qt IO system 功能介紹	
week 8	9/5	QTimer / QThread	計時器
week 9	9/15	使用常見函式庫 qcustomplot / OpenCV 基礎介紹	滑鼠控制事件 or 資料處理及統計
week 10	9/22	專題輔導	專題
week 11	9/29	專題輔導	
week 12	10/6	專題輔導	
week 13	10/13	專題報告	

QTimer

計時器



Timer (.ui)



Timer (.h)

```
11 class MainWindow : public QMainWindow
12 {
13     Q_OBJECT
14
15 public:
16     explicit MainWindow(QWidget *parent = 0);
17     ~MainWindow();
18
19 private slots:
20     void receive_timer_count();
21
22     void on_pushButton_start_clicked();
23
24     void on_pushButton_pause_clicked();
25
26     void on_pushButton_stop_clicked();
27
28     void on_doubleSpinBox_timer_interval_setup_valueChanged(double arg1);
29
30 private:
31     Ui::MainWindow *ui;
32
33     QTimer *Interval;
34
35     double timerInterval;
36
37     long int timerCount;
38 };
```

Timer (.cpp) - Start

```
4   MainWindow::MainWindow(QWidget *parent) :  
5       QMainWindow(parent),  
6   ▼   ui(new Ui::MainWindow)  
7       {  
8       ui->setupUi(this);  
9       Interval = 0;  
10      ui->doubleSpinBox_timer_interval_setup->setValue(1.00);  
11      timerCount = 0;  
12  }
```

```
21 ▼ void MainWindow::on_pushButton_start_clicked()  
22  {  
23  ▼   if(ui->pushButton_start->isEnabled())  
24      {  
25  ▼       if(!Interval)  
26          {  
27              Interval = new QTimer(0);  
28              Interval->setInterval(timerInterval*1000);  
29              Interval->start();  
30              connect(Interval, SIGNAL(timeout()), this, SLOT(receive_timer_count()));  
31          }  
32  ▼       else  
33          {  
34              Interval->start();  
35          }  
36  
37          ui->pushButton_pause->setEnabled(true);  
38          ui->pushButton_start->setEnabled(false);  
39          ui->pushButton_stop->setEnabled(true);  
40          ui->doubleSpinBox_timer_interval_setup->setEnabled(false);  
41      }  
42  }
```

Timer (.cpp) - Pause & Stop

```
46 void MainWindow::on_pushButton_pause_clicked()
47 {
48     Interval->stop();
49     ui->pushButton_pause->setEnabled(false);
50     ui->pushButton_start->setEnabled(true);
51 }
52
53 void MainWindow::on_pushButton_stop_clicked()
54 {
55     Interval->stop();
56     delete Interval;
57     Interval = 0;
58     timerCount = 0;
59     ui->label_show_count_num->setText(QString::number(timerCount));
60
61     ui->pushButton_pause->setEnabled(false);
62     ui->pushButton_start->setEnabled(true);
63     ui->pushButton_stop->setEnabled(false);
64     ui->doubleSpinBox_timer_interval_setup->setEnabled(true);
65
66 }
```


Timer (.cpp) - Count

```
68  void MainWindow::on_doubleSpinBox_timer_interval_setup_valueChanged(double arg1)
69  {
70      timerInterval = arg1;
71  }
72
73  void MainWindow::receive_timer_count()
74  {
75      timerCount++;
76      ui->label_show_count_num->setText(QString::number(timerCount));
77  }
78
```

QThread

多執行緒

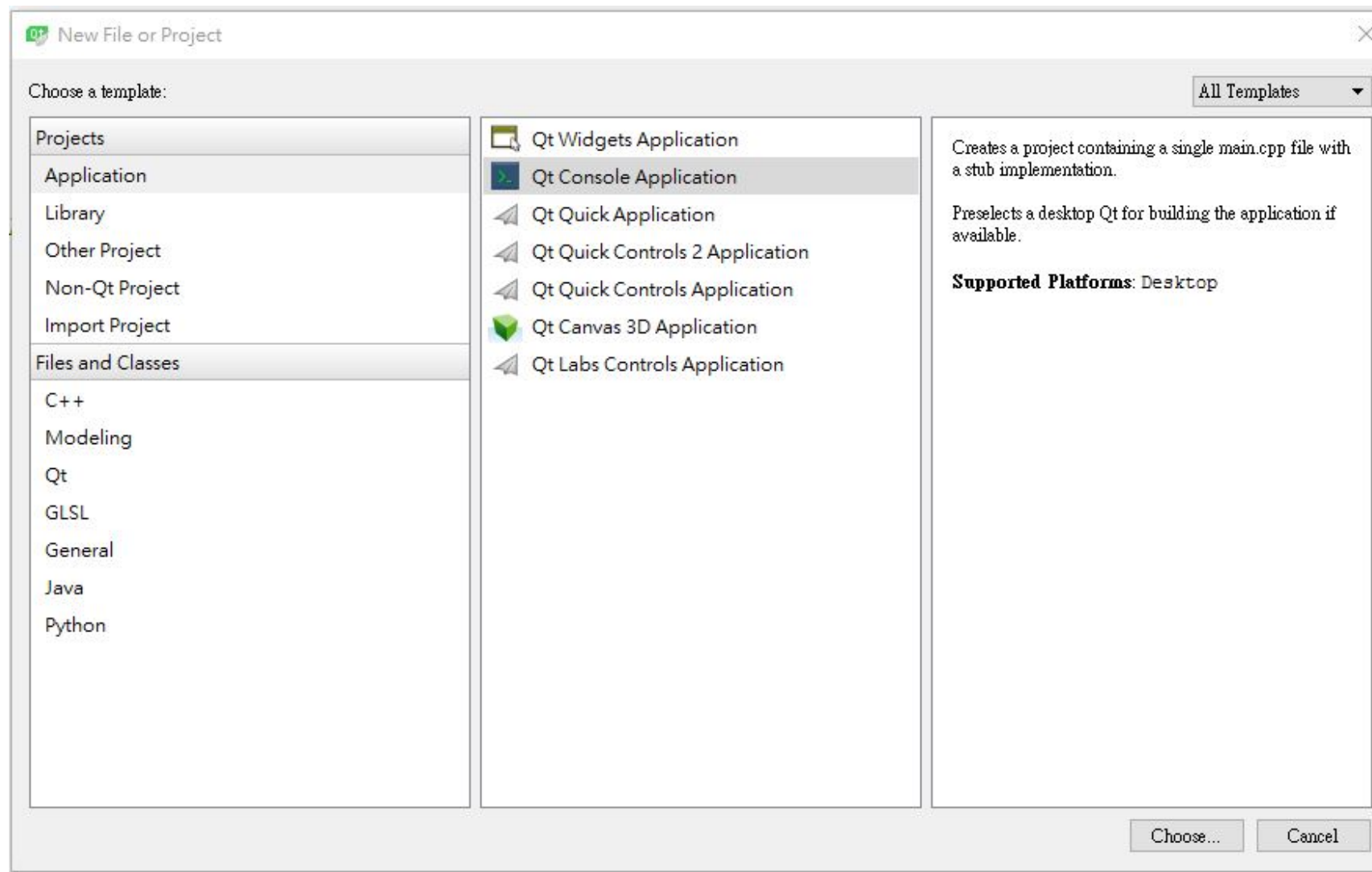


QThread - QT的多執行緒

- 執行序是程序中的執行流程，一個程序可以包含多個執行序，例如一方面接受網路的訊息，一方面接受使用者輸入訊息。
- 有時會需要對程序做觸發的動作，但又不影響到主程序的計算過程，例如計算時間。

建立Console 專案

- Console 是沒有介面設計的專案



MyThread (.h)

```
1  #ifndef MYTHREAD_H
2  #define MYTHREAD_H
3
4  #include <QThread>
5
6  class MyThread: public QThread
7  {
8  public:
9      MyThread();
10     void run();
11     QString name;
12
13 };
14
15 #endif // MYTHREAD_H
16
```

MyThread (.cpp) - Test&Run

```
1  #include "mythread.h"
2  #include <QThread>
3  #include <QDebug>
4
5
6  MyThread::MyThread()
7  {
8
9  }
10
11  void MyThread::run()
12  {
13      qDebug() << this->name << "Running";
14  }
15
16
```

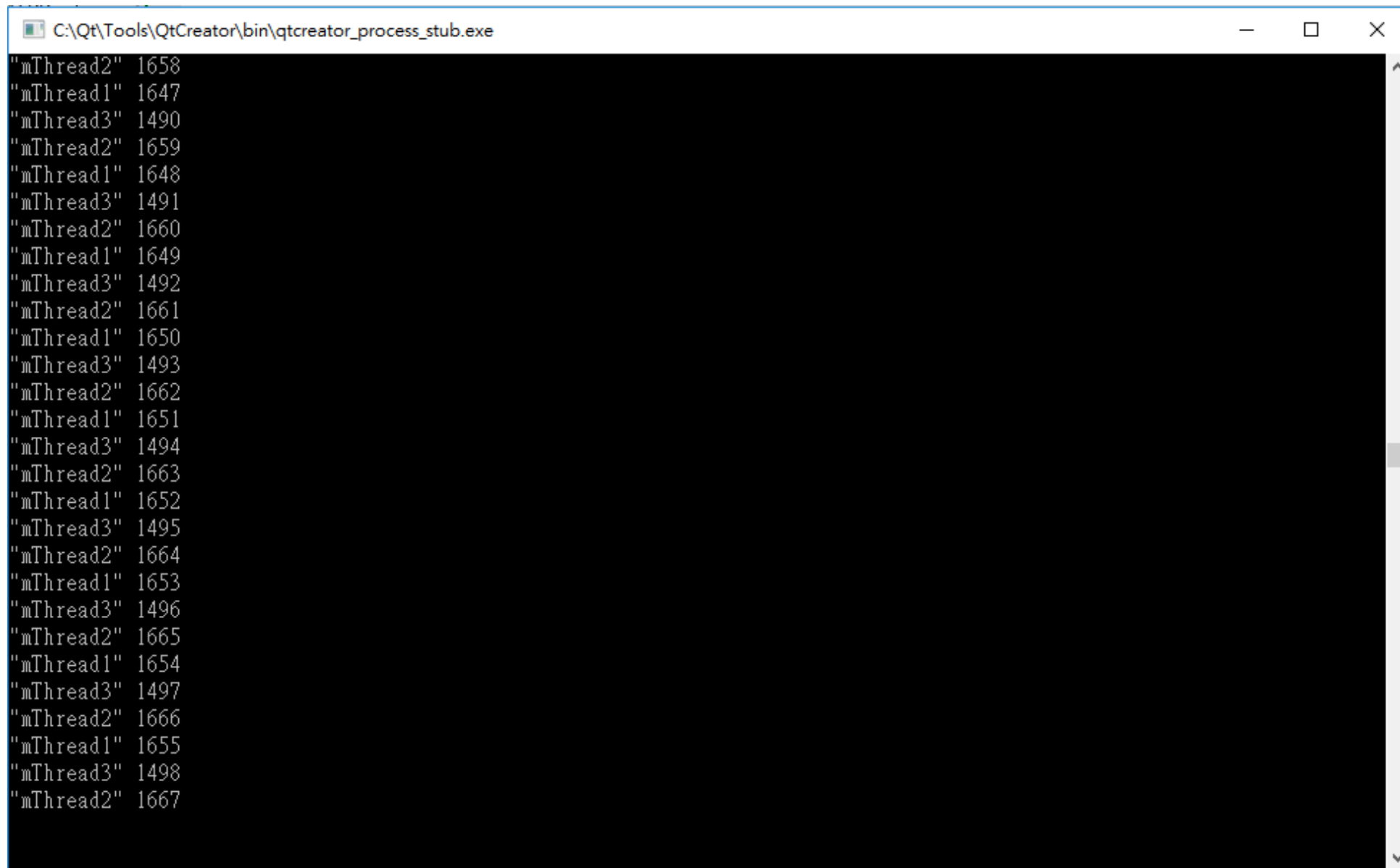
使用QThread (main.cpp)

```
1  #include <QCoreApplication>
2  #include "mythread.h"
3
4  int main(int argc, char *argv[])
5  {
6      QCoreApplication a(argc, argv);
7
8      MyThread mThread1;
9      mThread1.name="mThread1";
10
11     MyThread mThread2;
12     mThread2.name="mThread2";
13
14     MyThread mThread3;
15     mThread3.name="mThread3";
16
17
18     mThread1.start();
19     mThread2.start();
20     mThread3.start();
21
22     return a.exec();
23 }
24
```

使用QThread (.cpp)

```
1  #include "mythread.h"
2  #include <QThread>
3  #include <QDebug>
4
5
6  MyThread::MyThread()
7  {
8
9  }
10
11 void MyThread::run()
12 {
13     //QDebug() << this->name << "Running";
14
15     for(int i=0; i<10000; i++)
16     {
17         qDebug() << this->name << i;
18     }
19
20 }
21
```


Result



A screenshot of a Qt Creator console window. The title bar shows the file path "C:\Qt\Tools\QtCreator\bin\qtcreator_process_stub.exe". The console output displays a sequence of thread identifiers and their corresponding IDs, alternating between "mThread2" and "mThread1" followed by "mThread3". The IDs range from 1490 to 1667. The output is as follows:

```
"mThread2" 1658
"mThread1" 1647
"mThread3" 1490
"mThread2" 1659
"mThread1" 1648
"mThread3" 1491
"mThread2" 1660
"mThread1" 1649
"mThread3" 1492
"mThread2" 1661
"mThread1" 1650
"mThread3" 1493
"mThread2" 1662
"mThread1" 1651
"mThread3" 1494
"mThread2" 1663
"mThread1" 1652
"mThread3" 1495
"mThread2" 1664
"mThread1" 1653
"mThread3" 1496
"mThread2" 1665
"mThread1" 1654
"mThread3" 1497
"mThread2" 1666
"mThread1" 1655
"mThread3" 1498
"mThread2" 1667
```

設定優先順序

- `mThread1.start(QThread::HighestPriority);`
- `mThread2.start();`
- `mThread3.start(QThread::LowestPriority);`

Review

icon



新增icon

- in .pro file : `RC_ICONS = favicon.ico`
- 執行qmake
- 可以到以下網址製作ico
- <http://www.favicon.cc/>
- 畫一個icon放到程式裡

作業

使用 QThread 來寫你的計時器



Homework

- 將上面的範例Qtimer改用Qthread寫
- 時間的計算 `#include <time.h>`
- Time step固定為1秒即可
- 製作icon在自己的程式裡

參考網站

Qt 官方網站

[http://doc.qt.io/qt-5/
qtexamplesandtutorials.html](http://doc.qt.io/qt-5/qtexamplesandtutorials.html)

阿洲（中文）

http://monkeycoding.com/?page_id=5

影片教學（英文）

[https://www.youtube.com/watch?
v=6KtOzh0StTc&list=PL2D1942A4688E9D
63](https://www.youtube.com/watch?v=6KtOzh0StTc&list=PL2D1942A4688E9D63)

Thank you~~