

IN00BQ93 Laite- ja tuotesuunnittelun syventävät opinnot: ohjelmoinnin jatkokurssi

Olio-ohjelmointi: virtuaalifunktiot, prosessi ja säie

Tehtävä 1:

- Tee uusi projekti nimeltä **MyTimeThreadApp**, joka on tyyppiä **Application->Qt Console Application**.
- Lisää ohjelmaan kaksi luokkaa **MyTimeToUIThreadClass** ja **MyTimeToFileThreadClass**
- Laita yllä lisätyt luokat perimään luokka **QThread** ja lisää luokkiin virtuaalinen jäsenfunktion **run()** opiskelumateriaalin luvussa 8 opitun mukaisesti.
- Muodostin- ja tuhoajafunktiota ei luokissa tarvita, koska niille ei ole tässä vaiheessa nähtävissä mitään tehtävää.
- Jäsenfunktiossa **run()** ei luoda olioita tai paikallisia muuttujia.
- Käytä **qDebug()** funktiota näytölle tulostamiseen.

IN00BQ93 Laite- ja tuotesuunnittelun syventävät opinnot: ohjelmoinnin jatkokurssi

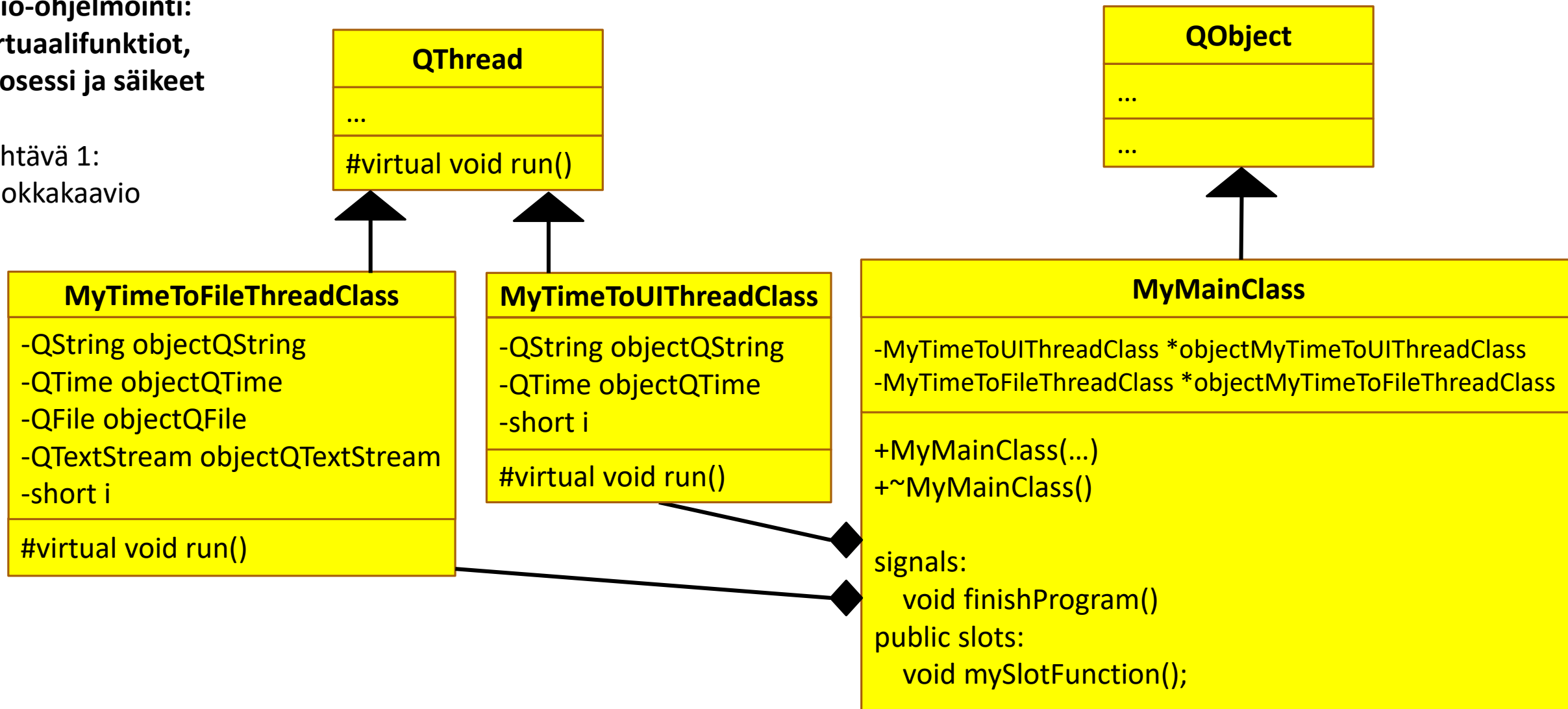
Olio-ohjelmointi: virtuaalifunktiot, prosessi ja säie

Tehtävä 1:

- **MyTimeToUiThreadClass** luokan säikeen täytyy tulostaa **run()** funktiossa 10 kertaa näytölle allekkain sekunnin välein kellonaika muodossa **tt:mm:ss**
 - Käytä toteutuksessa luokkia **QString** ja **QTime**. Jos **QTime** luokka ei löydy Qt-luokkakirjastosta **Search** –hakutoiminnolla, niin klikkaa linkkiä **All Qt C++ Classes** ja etsi luokka sitä kautta.
 - Etsi luokasta **QTime** funktio, jonka avulla voidaan selvittää tämänhetkinen/nykyinen kellonaika.
 - Kun kellonaika funktiolla luettu, niin se täytyy muuttaa **QString** tyyppiseksi, jotta se voidaan tulostaa näytölle **QDebug()** –funktiolla.
 - Kun säie on työnsä tehnyt täytyy näytölle tulostaa **MyTimeToUiThreadClass run() -function finished!**
- **MyTimeToFileThreadClass** luokan säikeen täytyy kirjoittaa hakemistossa **c:/temp** olevaan tiedostoon **myTime.txt**, 10 kertaa joka toisen sekunnin välein kellonaika muodossa **tt:mm:ss**
 - Käytä toteutuksessa luokkia **QString**, **QFile**, **QTime** ja **QTextStream**
 - Joka kerta kun kellonaika on kirjoitettu tiedostoon, tulostetaan näytölle **Write operation X done!** (X tarkoittaa järjestysnumeroa 0-9)
 - Kun säie on työnsä tehnyt täytyy näytölle tulostaa **MyTimeToFileThreadClass run() -function finished!**
- Lisää projektiin luokka **MyMainClass** ja rakenna se seuraavalla sivulla olevan luokkakaavion mukaisesti. *HUOM! Muista luokkaa lisättäessä laittaa ikkunassa **Define Class** luokan kantaluokaksi (**Base class**) luokka **QObject**.* Näin saadaan siis perintärakenne suoraan kuntoon, ja tarvittavat rakenteet SIGNAL/SLOT toiminnallisuutta varten.

Olio-ohjelmointi:
virtuaalifunktiot,
prosessi ja säikeet

Tehtävä 1:
Luokkakaavio



IN00BQ93 Laite- ja tuotesuunnittelun syventävät opinnot: ohjelmoinnin jatkokurssi

Tehtävä 1: Olio-ohjelmointi: virtuaalifunktiot, prosessi ja säie

- **main()** funktiossa luodaan luokasta **MyMainClass** olio nimeltä **objectMyMainClass** käyttäen dynaamista muistinhallintaa.
- Olio **objectMyMainClass** liitetään **connect** käskyllä aikaisemman opitun mukaisesti signaaliin **finishProgram()**, jonka kutsuminen aiheuttaa luokan **QCoreApplication** funktion **quit()** lähettämisen sanomajonoon, ja sovelluksen päättymisen oikein.
- Luokan **MyMainClass** muodostinfunktiossa luodaan oliot **objectMyTimeToUIThreadClass** ja **objectMyTimeToFileThreadClass** ja oliot tuhoetaan luokan **MyMainClass** tuhoajafunktiossa. Luokkien välillä on kooste-yhteys luokkakaavion mukaisesti.
- Olioiden luonnin jälkeen muodostinfunktiossa liitetään **connect** käskyllä olio **objectMyTimeToFileThreadClass** signaaliin **finished()**. Signaalin tapahtuminen aiheuttaa luokkaan **MyMainClass** kuuluvan **public slot:** osasassa olevan funktion **mySlotFunction()** kutsumisen. Esimerkkikoodi alla:

```
connect(objectMyTimeToFileThreadClass, SIGNAL(finished()), this, SLOT(mySlotFunction()));
```

- Signaali **finished()** löytyy luokasta **QThread** ja sitä voidaan käyttää, koska luokka **MyTimeToFileThreadClass** perii luokan **QThread**. Käy Qt-luokkakirjastossa etsimässä luokasta **QThread** signaali **finished()** ja tutustu signaalin toimintaan.
- Muuta rivi **return a.exec();** aikaisemmin opitun mukaisesti.
- Testaa ohjelmaa.