

Interfejsy aplikacji w środowisku Windows

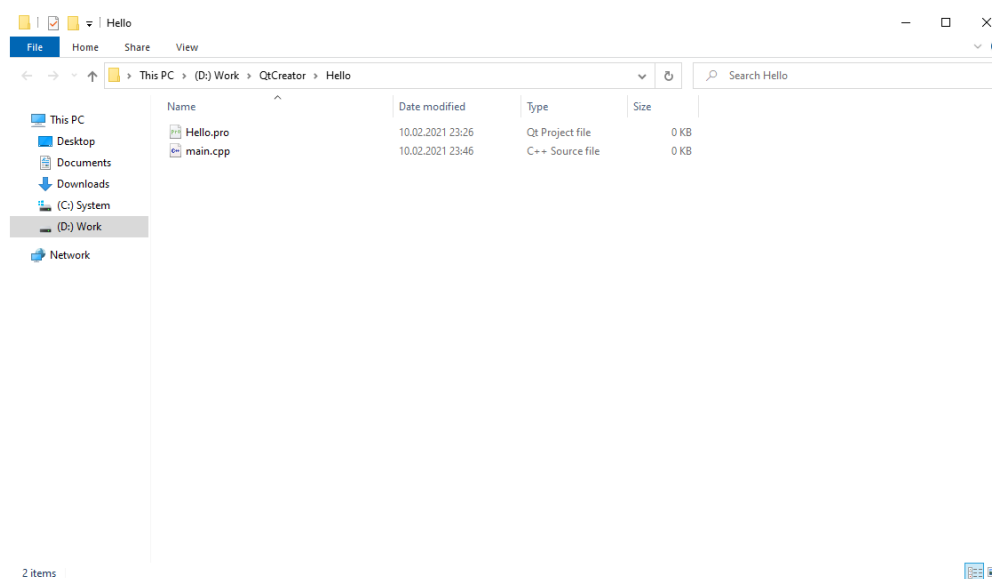
Struktura aplikacji Qt

Pierwsza aplikacja „od zera”

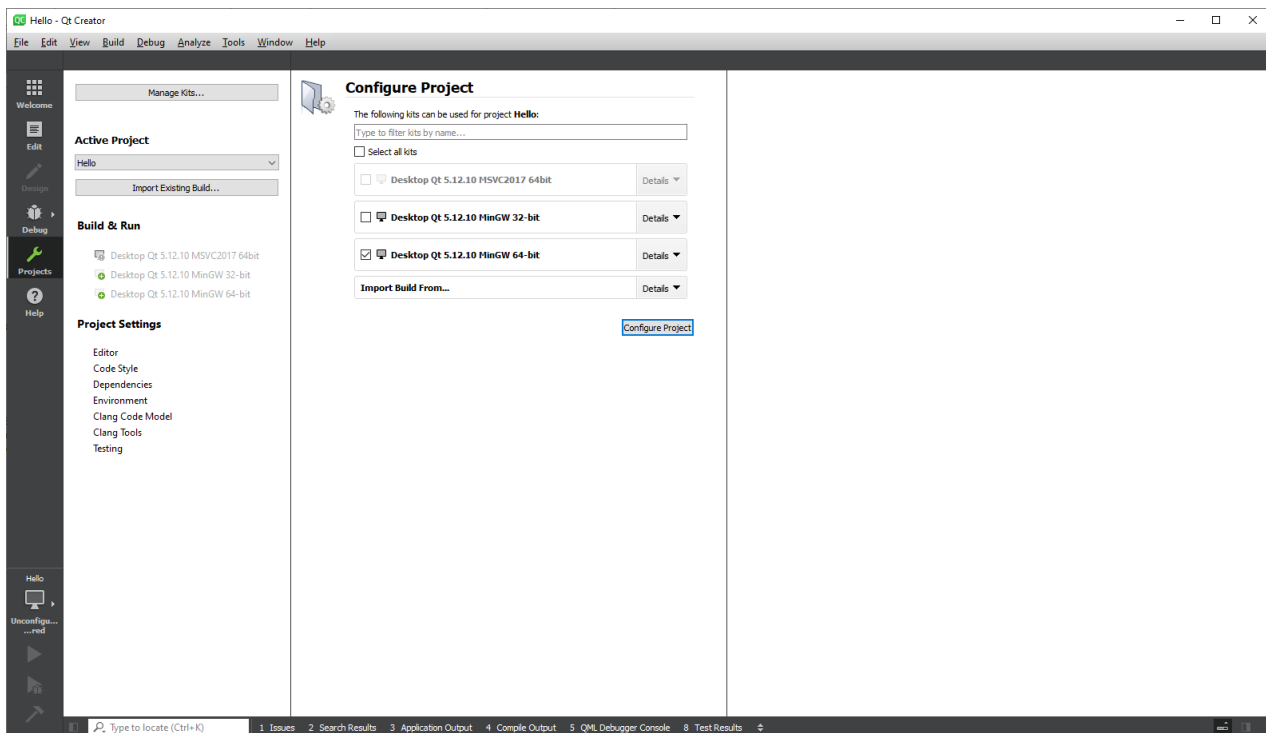
1. Przykład

Przykład pokazuje sposób tworzenia aplikacji „od zera” z wykorzystaniem biblioteki Qt.

- W katalogu z projektami (na dysku sieciowym **i:** we wcześniej założonym katalogu projektów) utwórz katalog nowej aplikacji (np. o nazwie **Hello**).
- Utwórz w tym katalogu dwa puste pliki tekstowe o nazwach: **Hello.pro** i **main.cpp**. Pamiętaj o usunięciu ewentualnych rozszerzeń ***.txt**.



- Otwórz plik projektu **Hello.pro** przez podwójne kliknięcie. Otworzy się IDE **Qt Creator** z oknem konfiguracji projektu. Wybierz odpowiedni **Kit**.



- Uzupełnij plik projektu o następujący kod.

```
QT += core gui
QT += widgets

CONFIG += c++11

SOURCES += main.cpp
```

- W katalogu projektu zostanie automatycznie utworzony dodatkowy plik **Hello.pro.user** w formacie **XML** z konfiguracją projektu.
- Po zapisaniu zmian plik **main.cpp** zostanie włączony do projektu. Uzupełnij go o kod.

```
#include <QApplication>
#include <QPushButton>

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

    QPushButton mojPrzcisk("Cześć!");
    mojPrzcisk.resize(100, 30);
    mojPrzcisk.show();

    return mojaAplikacja.exec();
}
```

- Uruchom program (skrót [CTRL + R] lub zielona strzałka w dolnej lewej części okna). Przetestuj i prześledź działanie programu.

2. Przykład

Poniższy przykład pokazuje aplikację wykorzystującą obiekt **QMainWindow**. Dzięki systemowi sygnałów i slotów, przycisk typu **QPushButton** został wyposażony w funkcjonalność zamknięcia okna aplikacji.

```
#include <QApplication>
#include <QPushButton>
#include <QMainWindow>

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

    QMainWindow mojeOkno;
    mojeOkno.setGeometry(400, 300, 300, 200);
    mojeOkno.setWindowTitle("Cześć!");

    QPushButton mojPrzycisk("Wyjście", &mojeOkno);
    mojPrzycisk.setGeometry(100, 120, 100, 40);

    QObject::connect(&mojPrzycisk, SIGNAL(clicked()), &mojaAplikacja, SLOT(quit()));

    mojeOkno.show();

    return mojaAplikacja.exec();
}
```

- Przepisz, uruchom i przetestuj program.

Pierwsza aplikacja orientowana obiektowo „od zera”

3. Zadanie

Poniższy kod pokazuje przykładową aplikację OO wykorzystującą bibliotekę Qt.

```
#include <QApplication>
#include <QMainWindow>
#include <QPushButton>

class GlowneOkno: public QMainWindow
{
public:
    GlowneOkno();
    ~GlowneOkno();

private:
    QPushButton *mojPrzycisk;
};

GlowneOkno::GlowneOkno(): QMainWindow()
{
    setGeometry(400, 300, 300, 200);
    setWindowTitle("Cześć!");

    mojPrzycisk = new QPushButton("Wyjście", this);
    mojPrzycisk->setGeometry(100, 120, 100, 40);

    connect(mojPrzycisk, SIGNAL(clicked()), QApplication::instance(), SLOT(quit()));
    // connect(&mojPrzycisk, SIGNAL(clicked()), qApp, SLOT(quit()));
}

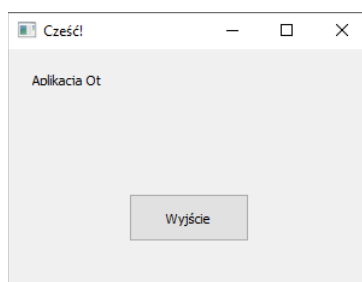
GlowneOkno::~GlowneOkno()
{
};
```

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

    GłowneOkno mojeGłowneOkno;
    mojeGłowneOkno.show();

    return mojaAPlikacja.exec();
}
```

- Utwórz ręcznie nowy projekt, jak wyżej, przepisz, uruchom i przetestuj program.
- Do okna dodaj obiekt typu **QLabel**.



- Wykreuj jego instancję jako właściciela podając wskaźnik **this** (jak w przypadku **QPushButton**).
- Zmień tekst umieszczony na etykiecie i ustal położenie.