

R2.02 : développement d'application avec IHM Feuille TD n° 6

Application avec fenêtre principale et fenêtre de Dialogue

Objectifs :

- 1.- Créer une application avec au moins 2 fenêtre : 1 principale et 1 fenêtre de Dialogue

1. Étape 1 : Préparer votre répertoire de travail

Dans le dossier consacré à la ressource R2.02, créer un dossier td6.

Activité 1 : Connexion

Sujet :

Créer une application simple avec 2 fenêtres.

Fenêtre 1 : C'est une fenêtre de connexion. La zone de texte destinée à recevoir le mot de passe n'affiche pas son contenu.

Lorsque la connexion est réussie (utilisateur et mot de passe corrects), un message (QMessageBox ::Information) s'affiche, puis une fenêtre de Dialogue apparaît.

Lorsque la connexion échoue, un message (QMessageBox ::Information) s'affiche, on revient à la fenêtre initiale.

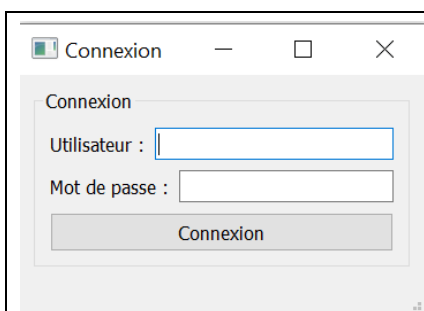


Figure 1 : Etat initial

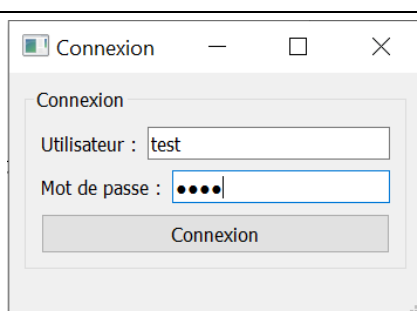


Figure 2 : Eléments saisis incorrects

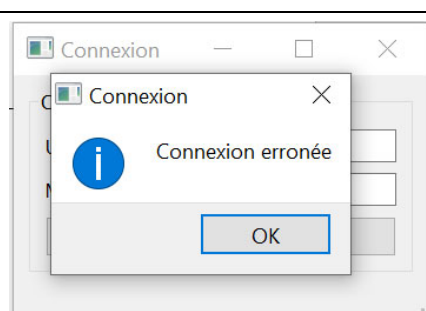


Figure 3 : Message approprié

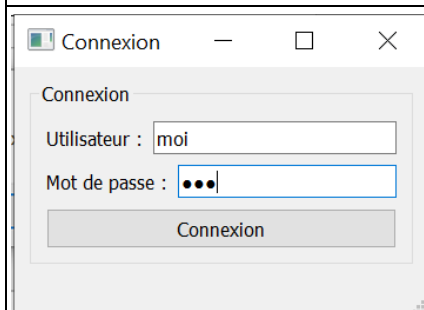


Figure 4 : Eléments saisis corrects

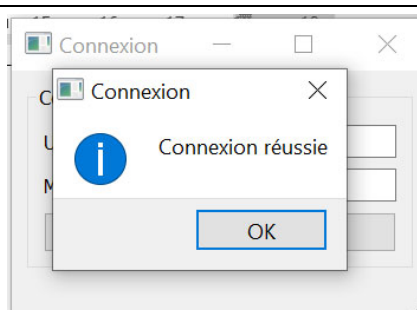


Figure 5 : Affichage approprié

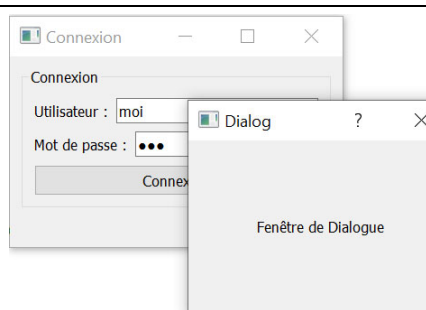


Figure 6 : Une nouvelle fenêtre de Dialogue s'ouvre

2. Étape 2 : Créer un projet nommé connexion

La fenêtre de Dialogue sera créée en utilisant les options montrées ci-dessous :

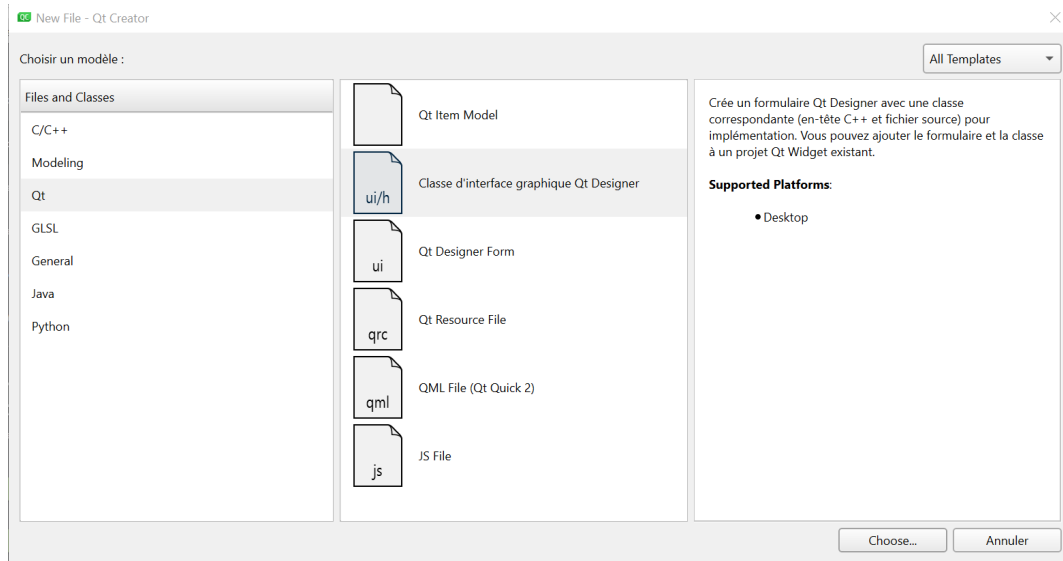


Figure 7 : Fenêtre de Dialogue : création de la classe et de sa fenêtre

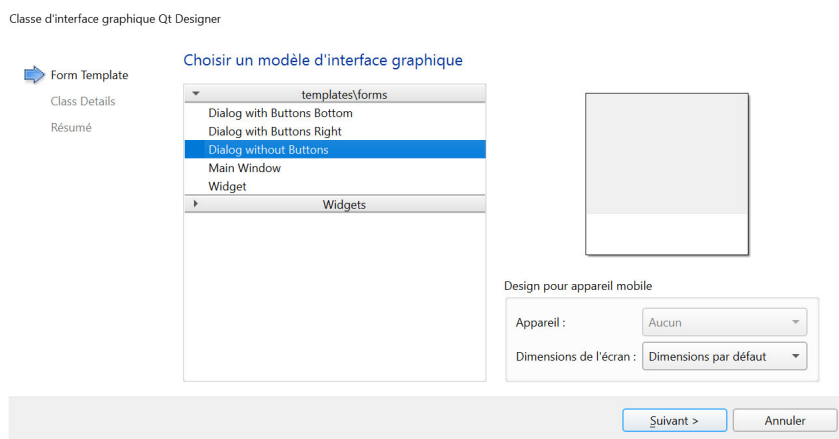


Figure 8 : Choix du style de la fenêtre de Dialogue

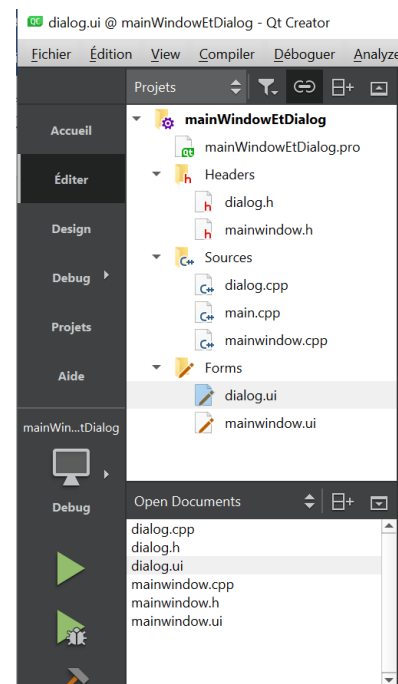


Figure 9 : Classe et fenêtre créées

Liens utiles pour les 2 étapes suivantes :

<https://doc.qt.io/qt-5/qdialog.html>

<https://doc.qt.io/qt-5/qmessagebox.html>

3. Étape 3 : Coder, tester **et commenter/décrire** chaque version fournie.

a) Version a

```
void MainWindow::demandeConnexion()
{
    QString utilisateur = ui->lineEditUtilisateur->text();
    QString mdP = ui->lineEditMdP->text();

    if ((utilisateur == "moi") && (mdP == "moi"))
    {
        QMessageBox::information(this, "Connexion", "Connexion réussie");
        Dialog maDlg(nullptr);
        maDlg.show(); // ou maDlg.exec();
    }
    else
    {
        QMessageBox::information(this, "Connexion", "Connexion erronée");
    }
}
```

b) Version b

```
void MainWindow::demandeConnexion()
{
    QString utilisateur = ui->lineEditUtilisateur->text();
    QString mdP = ui->lineEditMdP->text();

    if ((utilisateur == "moi") && (mdP == "moi"))
    {
        QMessageBox::information(this, "Connexion", "Connexion réussie");
        Dialog * maDlg= new Dialog(nullptr);
        maDlg->show();
    }
    else
    {
        QMessageBox::information(this, "Connexion", "Connexion erronée");
    }
}
```

c) Version c

```
void MainWindow::demandeConnexion()
{
    QString utilisateur = ui->lineEditUtilisateur->text();
    QString mdP = ui->lineEditMdP->text();

    if ((utilisateur == "moi") && (mdP == "moi"))
    {
        QMessageBox::information(this, "Connexion", "Connexion réussie");
        Dialog * maDlg= new Dialog(this);
        maDlg->exec();
    }
    else
    {
        QMessageBox::information(this, "Connexion", "Connexion erronée");
    }
}
```

d) Version

d :

La variable `maDlg` a été déclarée comme membre privé de la classe associée à la fenêtre principale.

```
void MainWindow::demandeConnexion()
{
    QString utilisateur = ui->lineEditUtilisateur->text();
    QString mdP = ui->lineEditMdP->text();

    if ((utilisateur == "moi") && (mdP == "moi"))
    {
        QMessageBox::information(this, "Connexion", "Connexion réussie");
        maDlg= new Dialog(this);
        maDlg->show();
    }
    else
    {
        QMessageBox::information(this, "Connexion", "Connexion erronée");
    }
}
```

e) Version e

La variable `maDlg` a été déclarée comme membre privé de la classe associée à la fenêtre principale.

```
void MainWindow::demandeConnexion()
{
    QString utilisateur = ui->lineEditUtilisateur->text();
    QString mdP = ui->lineEditMdP->text();

    if ((utilisateur == "moi") && (mdP == "moi"))
    {
        QMessageBox::information(this, "Connexion", "Connexion réussie");
        maDlg= new Dialog(this);
        this-> hide();
        maDlg->exec();
        this-> show();
    }
    else
    {
        QMessageBox::information(this, "Connexion", "Connexion erronée");
    }
}
```

4. Étape 4 : Comparaison QMessageBox et QDialog

Suite aux tests réalisés à l'étape précédente, à quel type de fenêtre de Dialogue particulière correspondent les fenêtres créées par QMessageBox ?

Pour vous en convaincre, consulter la documentation.