

INF1010
Programmation Orientée-Objet

Travail pratique #6
Interfaces graphiques et la gestion des exceptions

Objectifs :	Permettre à l'étudiant de se familiariser avec les interfaces graphiques (Qt) et la gestion des exceptions.
Remise du travail :	Lundi 22 Avril 2018, 8h
Références :	Notes de cours sur Moodle
Documents à remettre :	Les fichiers .cpp et .h complétés, les questions en pdf le tout sous la forme d'une archive au format .zip.
Directives :	Directives de remise des Travaux pratiques sur Moodle Les travaux dirigés s'effectuent obligatoirement en équipe de deux personnes faisant partie du même groupe. Veuillez suivre le guide de codage

Informations préalables

Veillez lire attentivement tout l'énoncé avant de commencer à écrire du code, afin de vous assurer de bien comprendre l'interaction entre les classes. Les tâches à effectuer vous sont présenter dans un ordre suggéré.

Le travail effectué dans ce TP continue celui amorcé par les TP 1, 2, 3, 4 et 5. Dans ce TP, nous créerons une interface graphique qui simule la commande en ligne de plats.

ATTENTION : Vous serez pénalisé pour l'utilisation de strings littéraires.

ATTENTION : Vous serez pénalisé pour les utilisations inutiles du mot-clé *this*. Utilisez-le seulement où nécessaire.

ATTENTION : Vous ne pouvez utiliser les fichiers du TP5 pour faire ce TP, utilisez ceux qui vous sont remis.

Remarque : Pour plus de précision sur le travail à faire et les changements à effectuer, veuillez-vous référer aux fichiers .h.

Travail à réaliser

Note : Lisez la description des classes ci-dessous et suivez aussi les indications des TODO insérées dans le code pour apporter vos modifications au code fourni.

Si une méthode d'affichage ne respecte le format de la capture donnée en annexe, vous devez modifier les méthodes en question.

Vous devez créer une interface qui simule la commande en ligne de plats dans un restaurant. Nous reprendrons les concepts que vous avez utilisé dans les derniers laboratoires.

Tous les plats que le restaurant offre sont lus depuis le fichier « polyFood.txt » et contenu dans la classe *Menu*.

L'interface de commande en ligne contiendra une première partie permettant d'appliquer un filtre sur l'ensemble des plats contenus dans la classe *Menu* afin d'obtenir des plats qui correspondent à certaines conditions. On peut vouloir certains plats en fonction du moment de la journée ou avoir certaines restrictions alimentaires telles que désirer des plats végétariens ou biologiques. La classe responsable de mémoriser les options de filtrage et de fournir les plats correspondants est la classe *Filtre*. Les plats filtrés seront affichés dans une liste, à la droite dans l'interface.

Finalement, les plats choisis par le client seront contenus dans la classe *Commande* et affichés dans une liste dans le bas de l'interface, avec le prix. Il pourra commander lorsqu'il sera satisfait de ses choix.

Design de l'interface

Fiez-vous à l'apparence voulue de l'interface dans la section « Affichage attendu ». Tout le travail doit être fait dans la classe *MainWindow*.

Création de « widgets »

Pour tous les « widgets », assurez-vous d'utiliser l'attribut correspondant de la classe.

1. Créer le bouton « Commander » de l'interface dans la fonction *creerWidgetCommander*. Vous pouvez vous inspirer de la fonction *creerWidgetBoutonsAjouterRetirer*.
2. Créer la liste contenant les plats de la commande dans la fonction *creerWidgetCommande*. Les éléments de la liste doivent être ordonnés par ordre alphabétique. Vous pouvez vous inspirer de la fonction *creerWidgetPlatsFiltres*.

Mise en page

Remplissez la fonction *designLayout* de la classe *MainWindow* afin d'obtenir une mise en page similaire à celle voulue. Insérer tous les widgets et les layouts créés dans le layout principal « *boitePrincipale* ».

Programmation événementielle

Connection du modèle à l'interface

Dans le main, notifier les fonctions de mise à jour de l'interface de *MainWindow* lorsqu'il y a un changement dans le modèle. Le modèle se compose des classes *Menu*, *Commande* et *Filtre*. Seules les classes *Commande* et *Filtre* devraient changer en fonction des actions de l'utilisateur. Ces fonctions de mise à jour sont les slots publics de la classe *MainWindow*.

Connection de l'interface au modèle

1. Implémenter la fonction *insérerPlatsChoisisDansCommande* permettant d'insérer dans la commande de l'utilisateur le plat sélectionné dans la liste des plats filtrés.
2. Implémenter la fonction *retirerPlatsChoisisDeCommande* permettant de retirer de la commande de l'utilisateur le plat sélectionné dans la liste des plats filtrés.
3. Dans la fonction *connecterSignauxAuxSlots* de la classe *MainWindow*, connecter ces deux dernières fonctions au clic de l'utilisateur sur les boutons correspondants.

Gestion des erreurs

Afficher une fenêtre d'erreur si l'utilisateur essaie de retirer un plat de sa commande qui n'est pas présent. Dans la fonction *retirerPlat* de la classe *Commande*, lancer une erreur de type *ErreurPlatIntrouvable* si le plat n'est pas dans la commande. L'exception doit être attrapée dans la fonction pour retirer le plat choisi de la commande dans la classe *MainWindow* que vous avez écrit précédemment. Vous pouvez vous inspirer de la fonction *commander* de la classe *MainWindow*.

Questions

1. Pourquoi, en cas d'erreur dans un constructeur, faut-il désallouer la mémoire allouée alors qu'elle est déjà désallouée dans le destructeur?

2. Expliquer sommairement le modèle MVC. En quoi les logiciels construits avec Qt sont différents du modèle MVC?

Corrections

La correction du TP se fera sur 20 points.

Voici les détails de la correction :

- (4 points) Compilation du programme
- (3 points) Exécution du programme
- (2 points) Affichage de l'interface
- (2 points) Gestion des signaux/slots
- (3 points) Gestion des erreurs
- (2 points) Qualité du code
- (2 points) Réponse aux questions

Affichage attendu

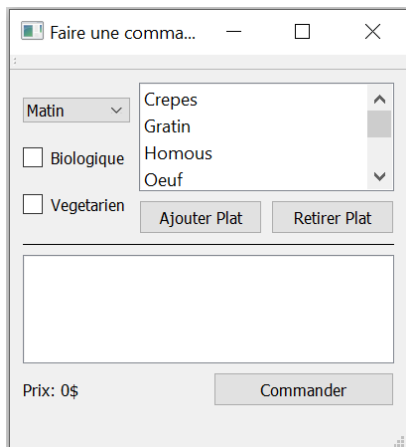


Figure 1. Affichage attendu

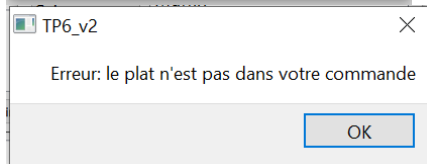


Figure 2. Message plat absent de la commande

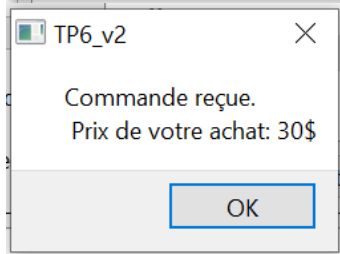


Figure 3. Message commande confirmée

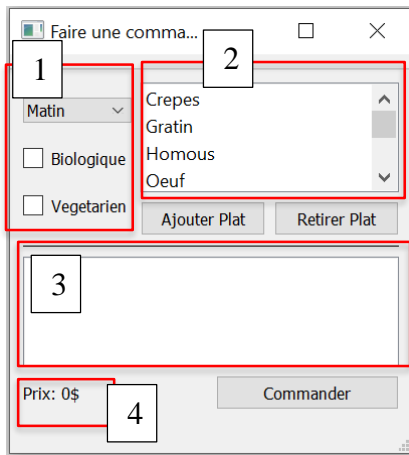


Figure 4 : Sous-sections de l'interface

1 : section contenant les options de filtrage des plats

2 : liste des plats filtrés

3 : liste des plats contenus dans la commande

4 : prix actualisé de la commande