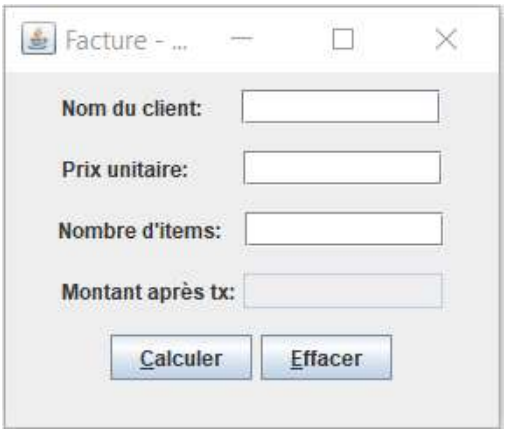
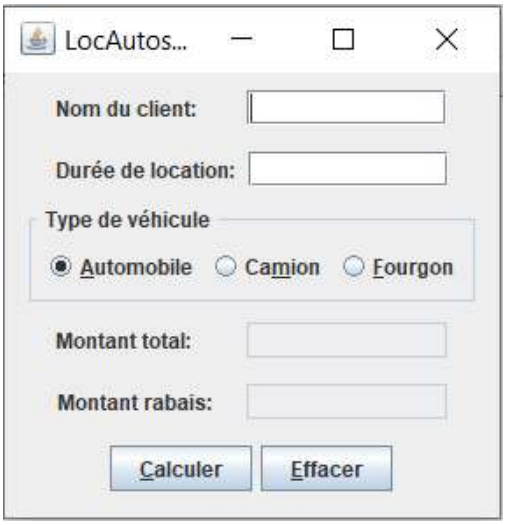
Atelier 05.2 – Interfaces graphiques

Exercice 1 : On vous demande de réaliser une application permettant de calculer le montant d'une facture. L'application doit contenir le formulaire (FrmFacture) ci-dessous.



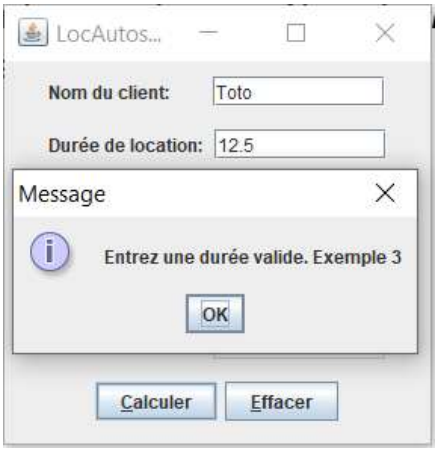
1. Écrire le code nécessaire pour créer le formulaire avec tous ses composants. La zone de texte *Montant après taxes* doit être non éditable (en lecture seule).
2. Écrire le code approprié pour le bouton *Calculer*. Lorsque l'utilisateur clique sur ce bouton, l'application doit calculer et afficher le montant avec taxes. Les taux de taxes TPS et TVQ sont de 5% et 9,975% respectivement. Utilisez la fonction *String.Format* pour afficher les montants avec deux chiffres après la virgule et le symbole monétaire.
3. Écrire le code pour le bouton *Effacer*. Le clic sur ce bouton doit effacer le contenu des zones de texte et placer ensuite le focus sur la zone de texte *Nom du client*.

Exercice 2 : ***LocAutos*** est une entreprise spécialisée dans la location de trois types de véhicules : Camion, Fourgon et Auto. Le coût de location par jour d'un camion est 100 $, celui d'un fourgon est 60 $ et celui d'une auto est 30 $. Les taxes sont incluses dans les prix. Le propriétaire désire un logiciel permettant de préparer facilement la facture. Il suffit d'entrer le nom du client, le type de véhicule et la durée (nombre de journées) de location. Un rabais s'applique sur le total de la facture dépendant du type de véhicule et du nombre de jours de location. Pour un camion et un fourgon, le rabais est de 15% si la durée de location est entre 3 et 6 jours et 25% pour 7 jours ou plus. Pour une automobile, le rabais est de 10% si la durée de location est entre 3 et 6 jours et 20% pour 7 jours ou plus.
Votre application doit contenir le formulaire (FrmFacture) ci-dessous :



1. Écrire le code nécessaire pour créer le formulaire avec tous ses composants.
    a. Les zones de texte *Montant total* et *Montant rabais* doivent être non éditables.
    b. Les boutons radio doivent être groupés et leur conteneur doit avoir le titre «*Type de véhicule*».
    c. Le bouton radio Automobile doit être sélectionné par défaut.

2. Écrire le code approprié pour le bouton *Calculer*. Lorsque l'utilisateur clique sur ce bouton, l'application doit calculer et afficher le montant total à payer (après rabais) par le client ainsi que le montant des rabais accordés. Utilisez la fonction *String.Format* pour afficher les montants avec deux chiffres après la virgule et le symbole monétaire. Vous devez vérifier que les données saisies sont valides (la durée est un nombre entier et le nom du client comporte au moins 2 caractères), sinon un message approprié doit être affiché dans une nouvelle boite de dialogue.



3. Écrire le code pour le bouton *Effacer*. Le clic sur ce bouton doit effacer le contenu des zones de texte et placer ensuite le focus sur la zone de texte *Nom du client*. Il doit également remettre la sélection sur le bouton radio *Automobile*.

Exercice 3 : On vous demande de reprendre l'exercice 1 et de réécrire le code de votre application en ajoutant une nouvelle classe **Facture** qui contiendra les constantes et attributs privés suivants :

- TPS = 0.05 (constante de type double)
- TVQ = 9.9975 (constante de type double)
- ***nomClient*** (nom du clients) de type String
- ***prixUnitaire*** (prix unitaire) de type double
- ***nbItems*** (nombre d'items) de type int

Cette classe contient également les constructeurs et méthodes suivants que vous devez implémenter:

- Les accesseurs (getters) et mutateurs (setters) pour tous les attributs
- Un constructeur qui permettra de construire une facture à partir du nom du client, du prix unitaire et du nombre d'items.
- Une méthode `public double calculerMontantApresTaxes()` qui permet de calculer et retourner le montant total après taxes de la facture.

Vous devez modifier le code du bouton *Calculer*. Lorsque l'utilisateur clique sur ce bouton, l'application doit :

- Déclarer un objet ***facture*** de la classe **Facture** et initialiser cet objet en utilisant les informations (nom du client, prix unitaire et nombre d'items) entrées dans la fenêtre.
- Appeler la méthode `calculerMontantApresTaxes()` pour calculer le montant après taxes et l'afficher ensuite dans le champ de texte correspondant.

Exercice 4 : On vous demande de reprendre l'exercice 2 et de réécrire le code de votre application en ajoutant une nouvelle classe **Location** qui contiendra les constantes et attributs privés suivants :

- PRIX_AUTO = 30 (constante de type double)
- PRIX_CAMION = 100 (constante de type double)
- PRIX_FOURGON = 60 (constante de type double)
- ***nomClient*** (nom du clients) de type String
- ***duree*** (durée de location) de type int
- ***typeVehicule*** (type de véhicule) de type String

Cette classe contient également les constructeurs et méthodes suivants que vous devez implémenter:

- Les accesseurs (getters) et mutateurs (setters) pour tous les attributs
- Un constructeur qui permettra de construire une location à partir du nom du client, de la durée de location et du type de véhicule.
- Une méthode `public double calculerMontantAvantRabais()` qui permet de calculer et retourner le montant avant rabais de la facture.

- Une méthode `public double calculerMontantTotal()` qui permet de calculer et retourner le montant total de la facture.
- Une méthode `public double calculerMontantRabais()` qui permet de calculer et retourner le montant du rabais.

Vous devez modifier le code du bouton *Calculer*. Lorsque l'utilisateur clique sur ce bouton, l'application doit :

- Déclarer un objet ***location*** de la classe **Location** et initialiser cet objet en utilisant les informations (nom du client, durée de location et type de véhicule) entrées dans la fenêtre si elles sont valides.
- Appeler la méthode `calculerMontantRabais()` pour calculer le montant du rabais et l'afficher ensuite dans le champ de texte correspondant.
- Appeler la méthode `calculerMontantTotal()` pour calculer le montant total et l'afficher ensuite dans le champ de texte correspondant.