

# Compte Rendu

*Projet Train*

**Stagiaire : BOROWEC Enzo**

**Enseignant: Abdel FAQIR**

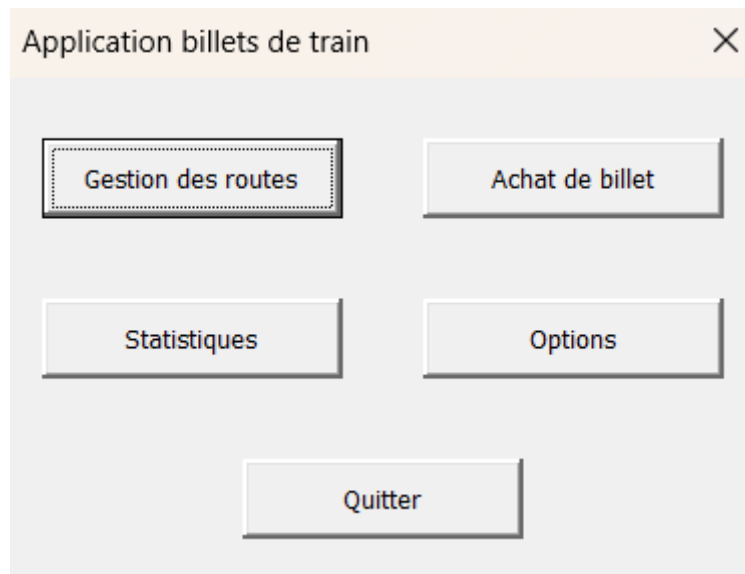
16/01/2025

**BUT GEII**

**Année 2024-2025**

Ce compte rendu de projet permet de comprendre le fonctionnement de l'application VBA en rapport avec le projet de billet de Train.

## I. Le menu



Ci-dessus voici le menu principal regroupant les différentes options du projet.

En premier, le bouton Gestion des routes va nous donner accès à la fenêtre gestion des routes comportant différentes options

Ensuite nous avons le bouton Achat de billet, nous donnant accès à la fenêtre qui nous permettra comme son nom l'indique, de procéder à l'achat de billet

Après nous avons le bouton statistique qui va nous permettre de tracer des diagrammes nous donnant un aperçu du nombre de billet acheté selon l'âge, fumeur ou non-fumeur et la classe du passager.

Nous poursuivons avec un bouton nous donnant accès à différents paramètres question prix du billet

Et un bouton pour quitter l'application

## II. Achat d'un billet

Achat d'un billet de train

Départ  Arrivée

Classe

☐ 1ère classe

☐ 2ème classe

Sexe

☐ Homme

☐ Femme

Classe

☐ Fumeur

☐ Non fumeur

Age :

Acheter

Retour au menu

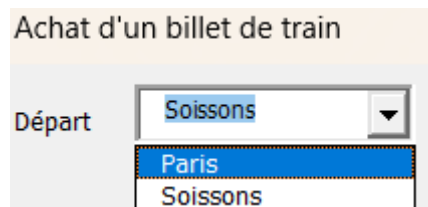
Voici la fenêtre d'achat des billets. Dans cette fenêtre, on va pouvoir y faire plusieurs choses.

En premier, on choisit l'un des départs disponible et ensuite on nous propose les différentes arrivées correspondant au départ. Voyons un exemple

Ici on a nos différentes routes :

Départ	Arrivée	1ère classe	2ème classe	Fumeur	Non Fumeur
Paris	Marseille	Oui	Oui	Non	Oui
Soissons	Paris	Oui	Oui	Oui	Oui
Soissons	Bordeaux	Non	Oui	Non	Oui

Ensuite je vais venir sélectionner mon départ :



Achat d'un billet de train

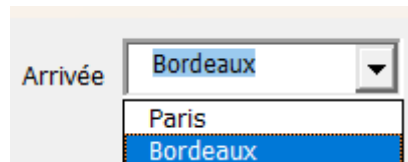
Départ

Soissons

Paris

Soissons

Et il va m'afficher uniquement les arrivés correspondant à mon départ :



Arrivée

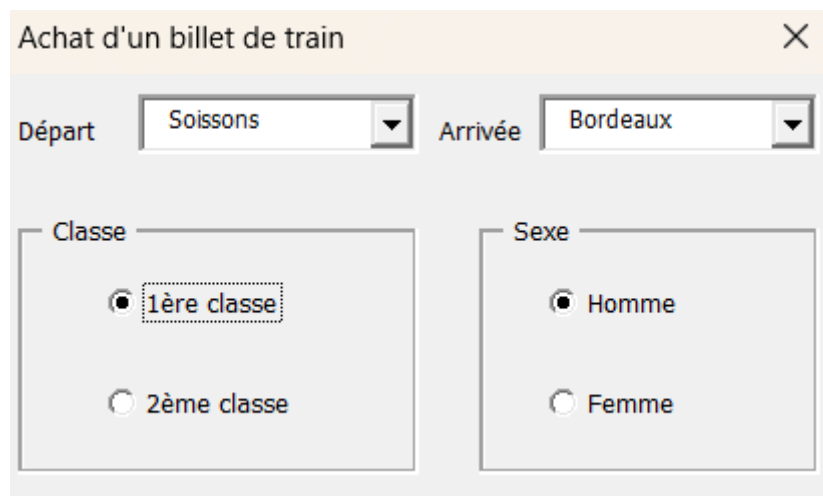
Bordeaux

Paris

Bordeaux

Ensuite on peut venir choisir notre classe et si on veut être dans un wagon fumeur ou non-fumeur. Par ailleurs un message d'erreur s'affiche si on décide d'être dans un wagon ne possédant pas les caractéristique choisies

Comme par exemple pour la route Soissons Bordeaux, on a pas de wagon 1<sup>ère</sup> classe,. Imaginons donc que je coche la 1<sup>ère</sup> classe :



Achat d'un billet de train

Départ Soissons Arrivée Bordeaux

Classe

☒ 1ère classe

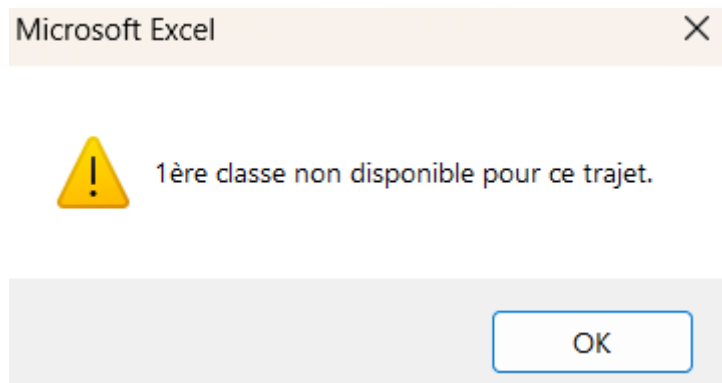
☐ 2ème classe

Sexe

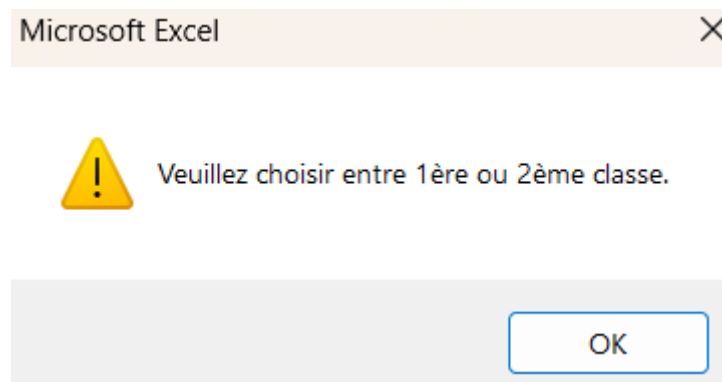
☒ Homme

☐ Femme

Je rentre mes autres informations et je confirme :



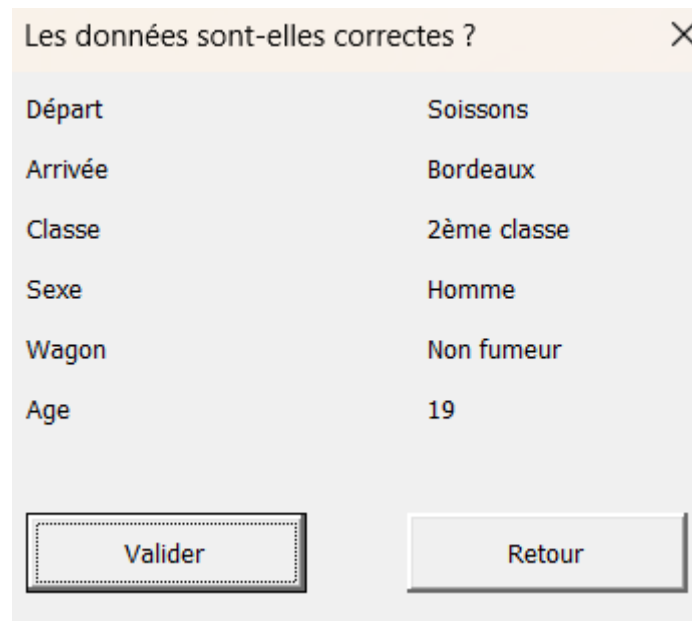
On a une première fenêtre nous indiquant que la première classe est non disponible puis on a une deuxième fenêtre nous demandant de recocher :



Si toutes nos informations sont valides on passe donc à la fenêtre suivante

### III. Données correctes

Cette fenêtre ci-dessous, sert principalement à vérifier nos informations pour nous permettre de revenir en arrière si une des informations est fausse :

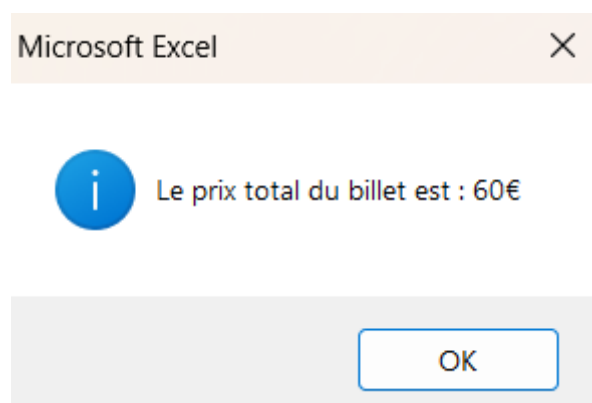


Départ	Soissons
Arrivée	Bordeaux
Classe	2ème classe
Sexe	Homme
Wagon	Non fumeur
Age	19

Valider Retour

On possède donc deux boutons :

Un Valider, qui permet de valider les informations, nous ouvrir un message concernant le prix du billet et de passer à la fenêtre suivante



Microsoft Excel

*i* Le prix total du billet est : 60€

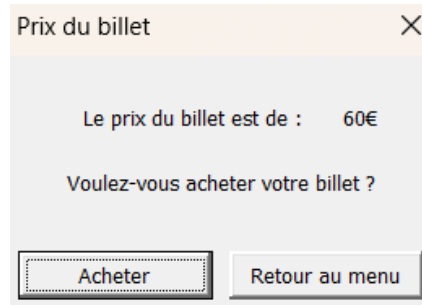
OK

Un Retour, qui permet de revenir à la fenêtre précédente

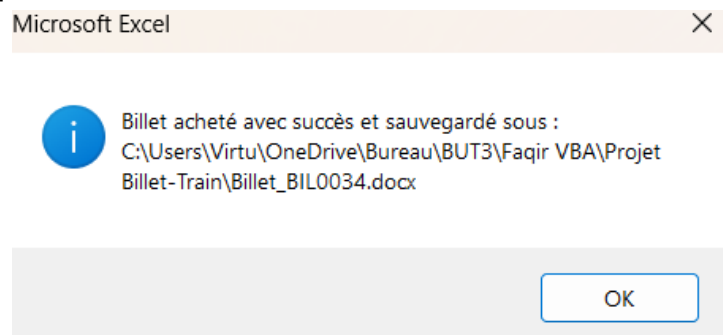
Si nos informations sont correctes passons donc à la fenêtre suivante

#### IV. Prix du billet

La prochaine fenêtre sert concrètement à valider ou non notre achat :



Si on appui sur le bouton acheter, une fenêtre indiquant que l'achat est effectué va apparaître :



Ensuite il va ouvrir un Word contenant un billet vierge avec des signets. Après le programme va venir grâce au signet remplir les informations indiquées précédemment. Pour finir, le fichier avec les informations sera enregistré en copie dans le répertoire désigné dans le programme

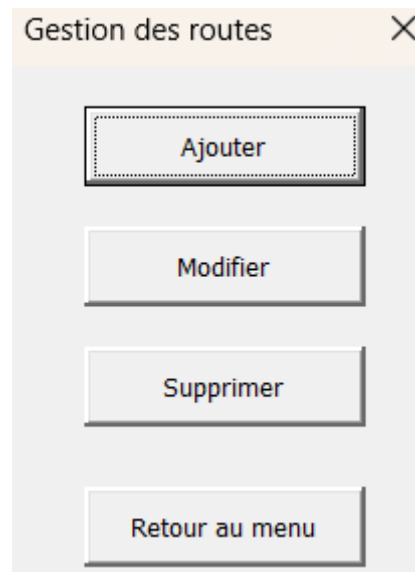
<b>BILLET</b> A composer avant l'accès au train		<b>N° billet : BIL0034</b>	
<b>DEPART EN BLEU</b>			
Départ <b>Soissons</b>		Classe <b>2è</b>	
DECOUVERTE 12-25 AGE A JUSTIFIER -ECH/REMB SO/CONDITIONS			
Arrivée <b>Bordeaux</b>			
		Prix EUR <b>60 €</b>	
LJ25 PG 25 KM0016		:DT 063938464	
:		:CA	
B		:6AD1BC Dossier QCDGCI	
08703289604334		080206 Page 1/1	

Le bouton retour lui, va annuler l'achat et retourner à la toute première fenêtre



## V. Gestion des routes

Ci-dessous voici la gestion des routes



Ici on a une interface plutôt simple par rapport aux autres. On a quatres boutons.

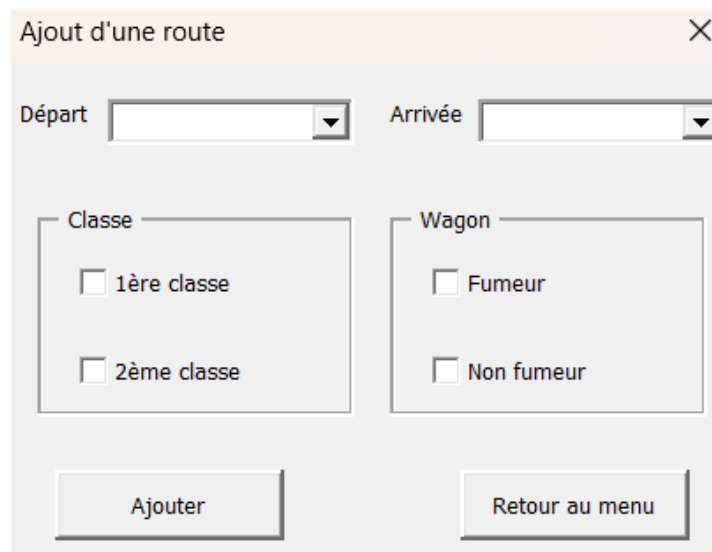
Un bouton Ajouter permettant d'ajouter une nouvelle route

Un bouton Modifier pour modifier les routes existantes

Un bouton Supprimer pour supprimer les routes existantes

Un bouton Retour menu qui retourne à la première fenêtre

## VI. Ajout d'une route



The screenshot shows a dialog box titled "Ajout d'une route" with a close button (X) in the top right corner. It contains two dropdown menus for "Départ" and "Arrivée". Below these are two sections: "Classe" with checkboxes for "1ère classe" and "2ème classe", and "Wagon" with checkboxes for "Fumeur" and "Non fumeur". At the bottom are two buttons: "Ajouter" and "Retour au menu".

Voici la fenêtre de l'ajout d'une route.

Dans les deux combobox Départ et Arrivée, on écrit les villes que l'on souhaite ajouter (Attention à l'écriture).

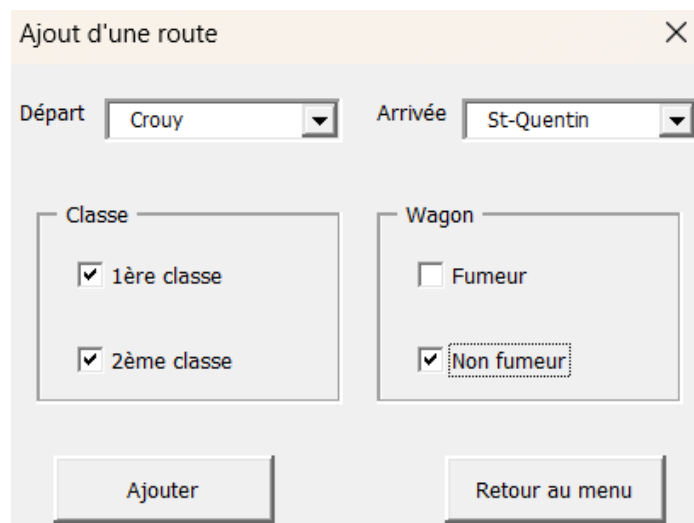
Ensuite on coche les différents wagon disponible pour notre route

Ensuite on à le bouton Ajouter pour donc ajouter notre route dans la feuille route

Et on a un bouton retour au menu pour revenir à la fenêtre précédente

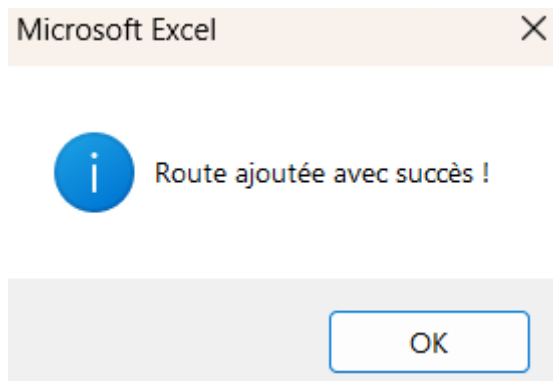
Prenons un exemple

Voici les informations entrées



This screenshot shows the same dialog box as before, but with example data entered. The "Départ" dropdown is set to "Crouy" and the "Arrivée" dropdown is set to "St-Quentin". In the "Classe" section, both "1ère classe" and "2ème classe" are checked. In the "Wagon" section, "Non fumeur" is checked, while "Fumeur" is not. The "Ajouter" and "Retour au menu" buttons are still at the bottom.

Je clique ensuite sur ajouter, un message va apparaître nous confirmant l'ajout de notre route :

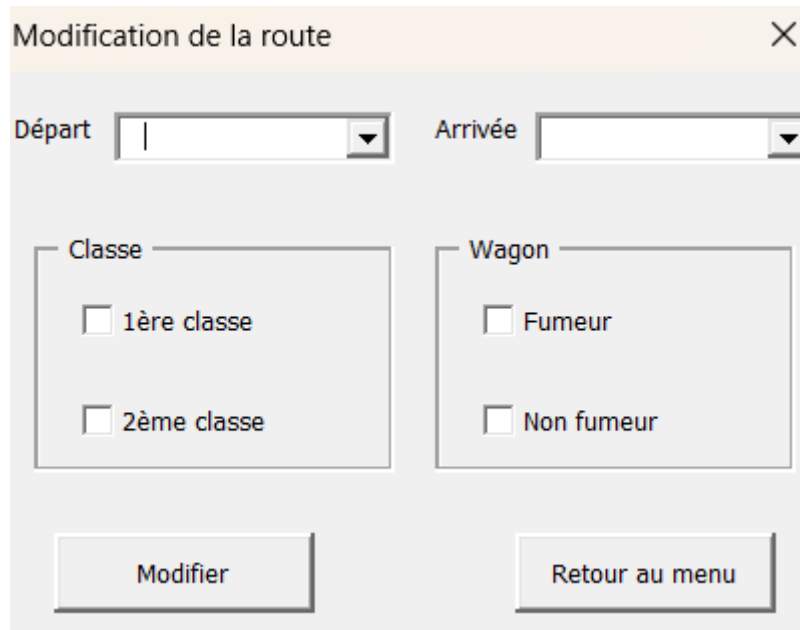


Puis on peut apercevoir notre route dans la feuille excel

Départ	Arrivée	1ère classe	2ère classe	Fumeur	Non Fumeur
Paris	Marseille	Oui	Oui	Non	Oui
Soissons	Paris	Oui	Oui	Oui	Oui
Soissons	Bordeaux	Non	Oui	Non	Oui
Crouy	St-Quentin	Oui	Oui	Non	Oui

## VII. Modification d'une route

Voici la fenêtre de modification d'une route :

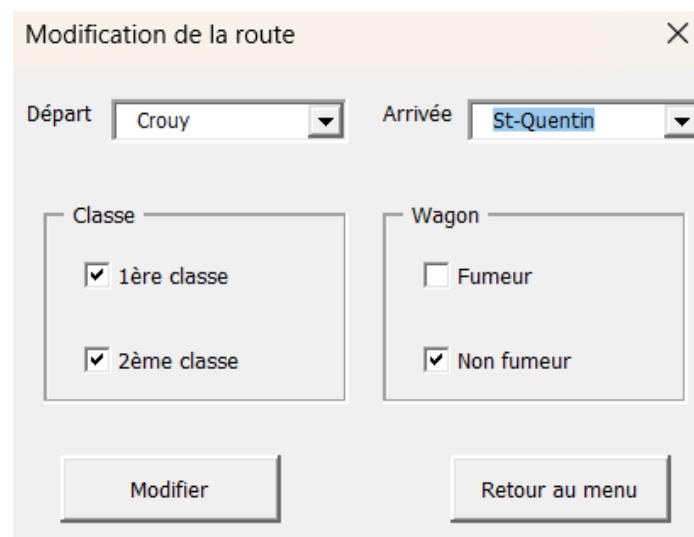


The screenshot shows a window titled "Modification de la route" with a close button (X) in the top right corner. It contains two dropdown menus for "Départ" and "Arrivée", both of which are empty. Below these are two groups of checkboxes. The "Classe" group has two options: "1ère classe" and "2ème classe", both of which are unchecked. The "Wagon" group has two options: "Fumeur" and "Non fumeur", both of which are unchecked. At the bottom of the window are two buttons: "Modifier" and "Retour au menu".

Elle se présente comme la fenêtre d'ajout d'une route à quelques différences prêtes que voici :

- Notre choix de départ et d'arrivée fonctionne comme pour l'achat de billet
- Les cases se cochent automatiquement dès que le départ et l'arrivée sont choisis, il ne reste donc plus qu'à modifier ce que l'on souhaite

Prenons un exemple où je modifie la route que nous avons créé :



The screenshot shows the same "Modification de la route" window, but now with data. The "Départ" dropdown is set to "Crouy" and the "Arrivée" dropdown is set to "St-Quentin". In the "Classe" group, both "1ère classe" and "2ème classe" are checked. In the "Wagon" group, "Non fumeur" is checked, while "Fumeur" remains unchecked. The "Modifier" and "Retour au menu" buttons are still present at the bottom.

Comme on aperçoit, les cases sont automatiquement remplit sur la fenêtre imaginons que changement de dernière minute, il y'a des wagons fumeurs, on vient donc cocher :

Modification de la route

Départ
Crouy

Arrivée
St-Quentin

Classe

☒ 1ère classe
☒ 2ème classe

Wagon

☒ Fumeur
☒ Non fumeur

Modifier

Retour au menu

On appui sur le bouton modifié, on a un message pour nous confirmer la modification et sur notre feuille, les infos s'actualisent :

Microsoft Excel

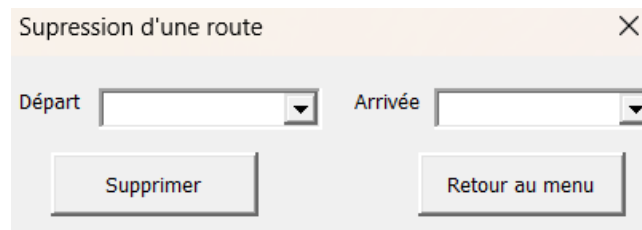
i

Route modifiée avec succès!

OK

Départ	Arrivée	1ère classe	2ère classe	Fumeur	Non Fumeur
Paris	Marseille	Oui	Oui	Non	Oui
Soissons	Paris	Oui	Oui	Oui	Oui
Soissons	Bordeaux	Non	Oui	Non	Oui
Crouy	St-Quentin	Oui	Oui	Oui	Oui

## VIII. Suppression route

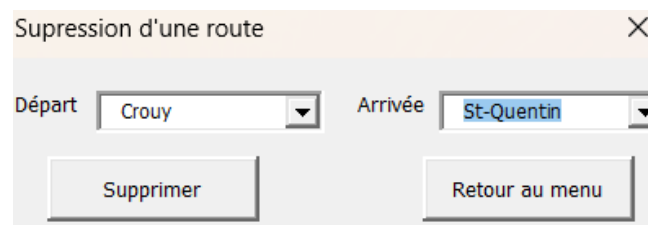


The dialog box titled 'Suppression d'une route' has a close button (X) in the top right corner. It contains two dropdown menus: 'Départ' and 'Arrivée'. Below the 'Départ' dropdown is a button labeled 'Supprimer'. Below the 'Arrivée' dropdown is a button labeled 'Retour au menu'.

Voici la fenêtre de suppression de route.

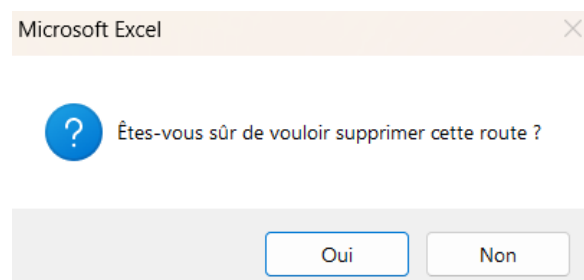
Comme pour la modification, on nous affiche les différents départs et les arrivées en fonction du départ. Un bouton pour supprimer et un bouton retour menu.

Imaginons que nous souhaitons supprimer la route de Crouy :



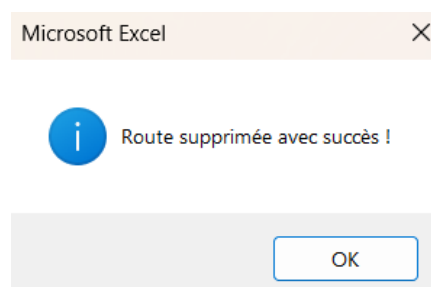
The dialog box titled 'Suppression d'une route' shows 'Crouy' selected in the 'Départ' dropdown menu. The 'Arrivée' dropdown menu shows 'St-Quentin' selected. The 'Supprimer' and 'Retour au menu' buttons are visible at the bottom.

On appui sur supprimer, un message nous demande si on souhaite vraiment supprimer :



A dialog box from Microsoft Excel with a question mark icon. The text reads: 'Êtes-vous sûr de vouloir supprimer cette route ?'. At the bottom, there are two buttons: 'Oui' and 'Non'.

On appui sur oui et un message nous confirme la suppression :

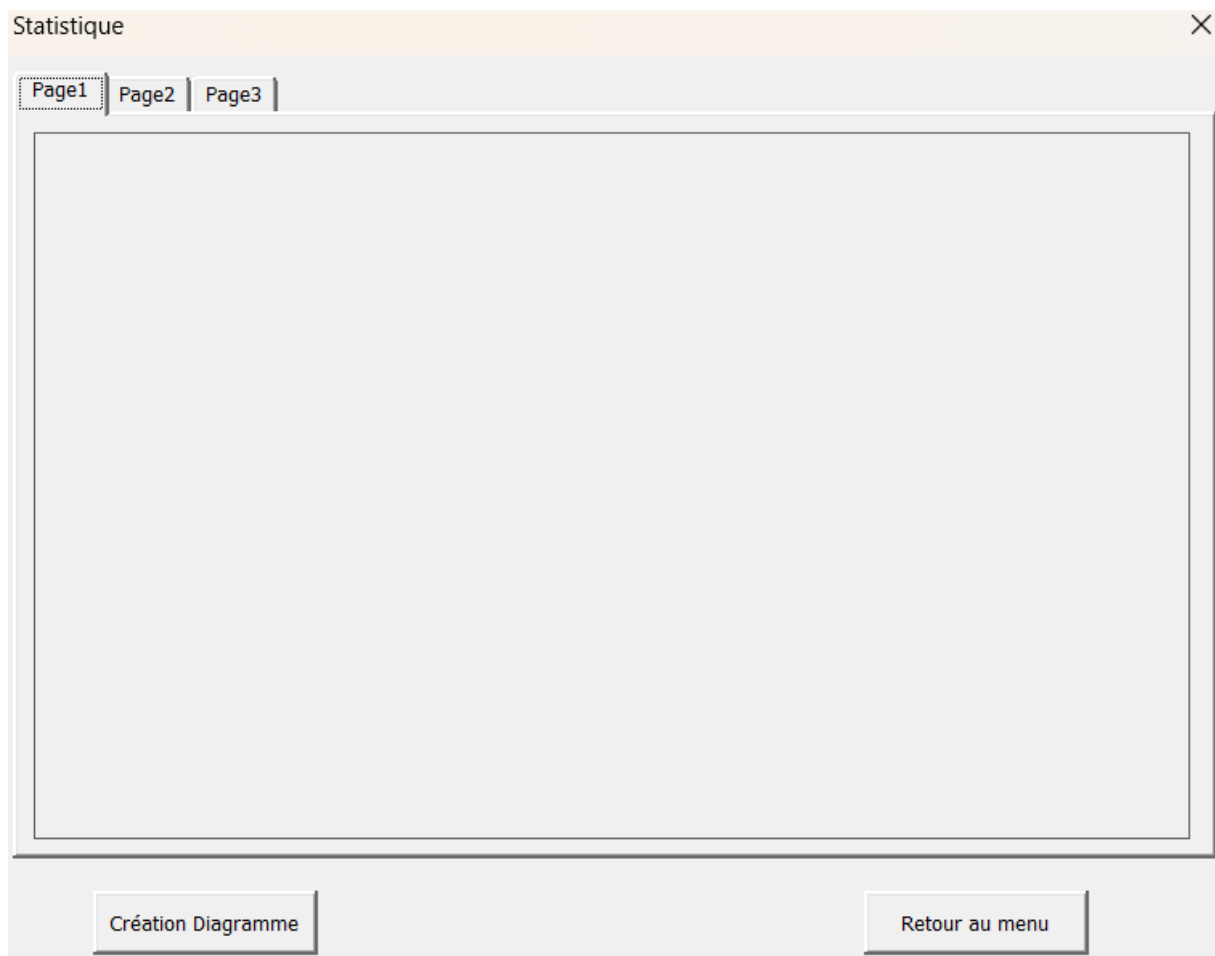


A dialog box from Microsoft Excel with an information icon (i). The text reads: 'Route supprimée avec succès !'. At the bottom, there is a single button labeled 'OK'.

Départ	Arrivée	1ère classe	2ère classe	Fumeur	Non Fumeur
Paris	Marseille	Oui	Oui	Non	Oui
Soissons	Paris	Oui	Oui	Oui	Oui
Soissons	Bordeaux	Non	Oui	Non	Oui

On remarque donc que la route de Crouy est supprimée

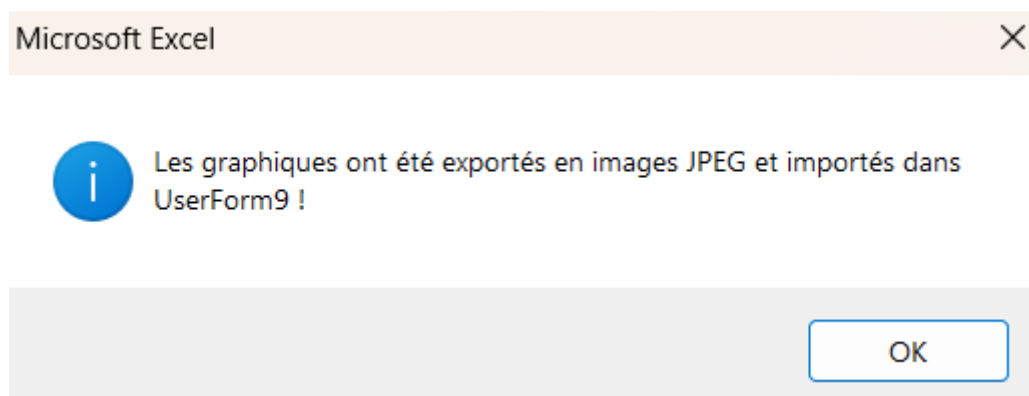
## IX. Statistique



Voici la fenêtre, on a deux boutons




Une création de diagramme qui va venir regarder dans la feuille statistique qui enregistre tous les billets et mettre sous forme de diagramme différents d'infos. Ensuite, il va enregistrer les diagrammes en image Jpeg et nous les insérer sur les différentes pages.

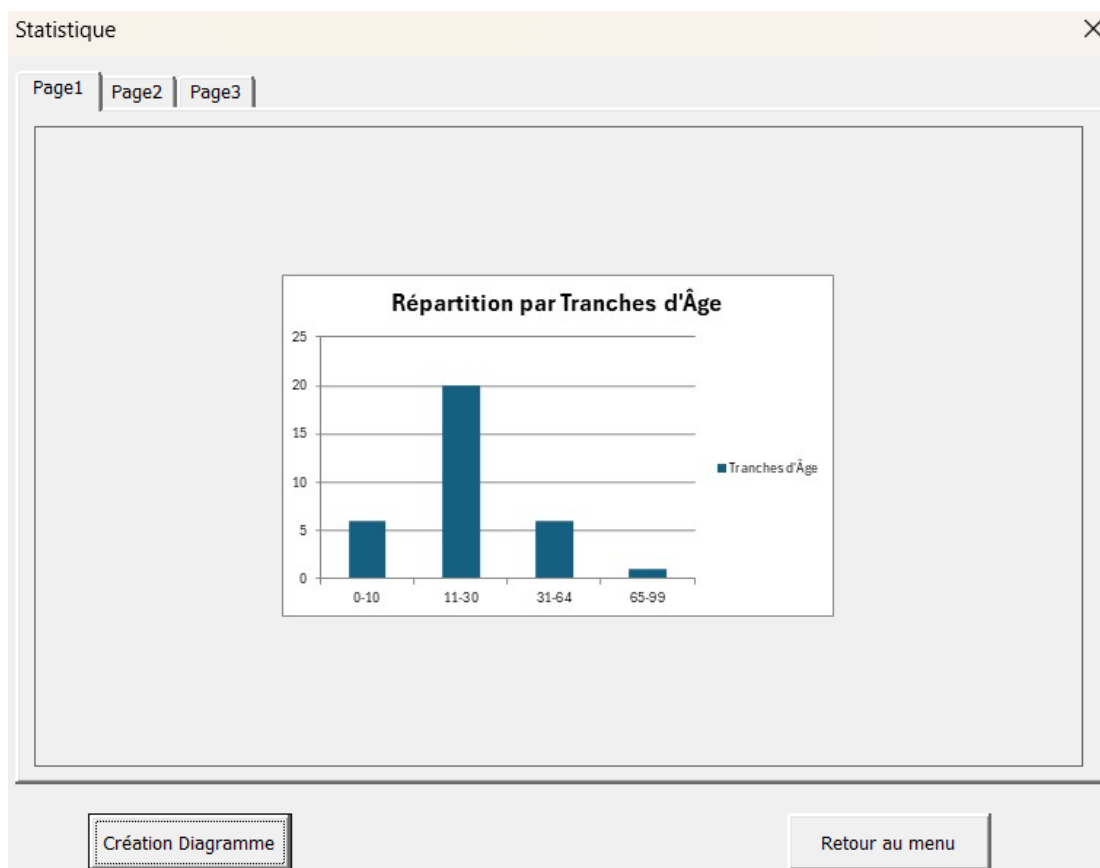
Voici un exemple :

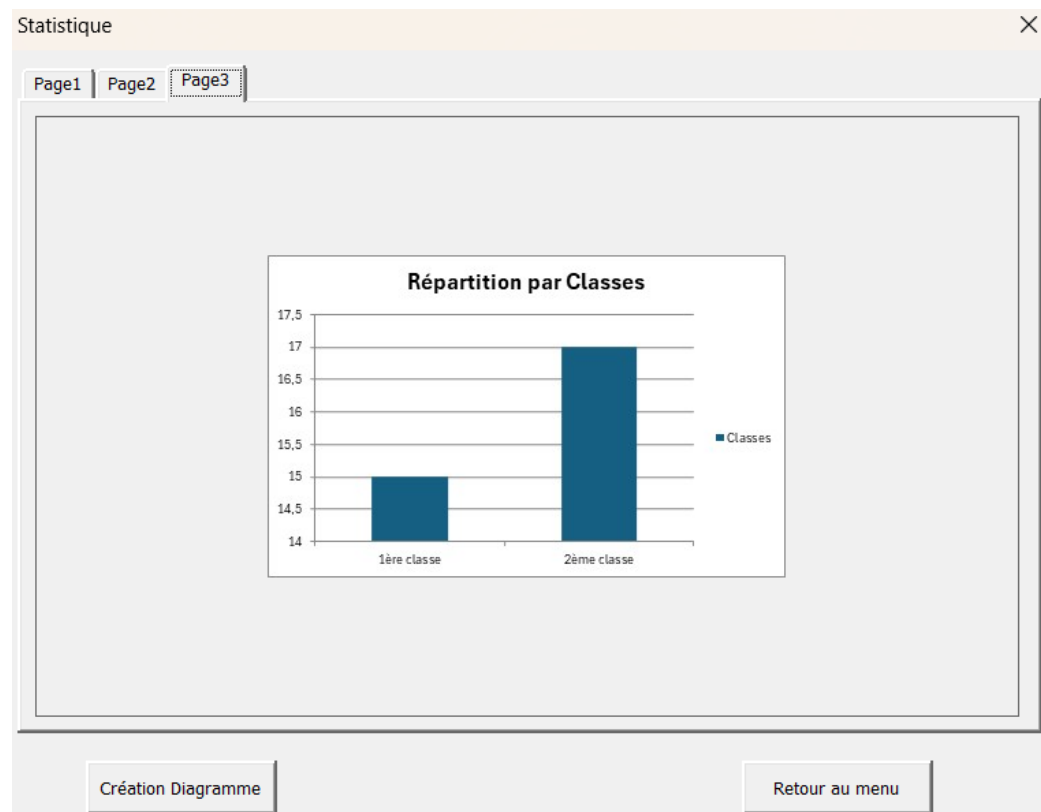
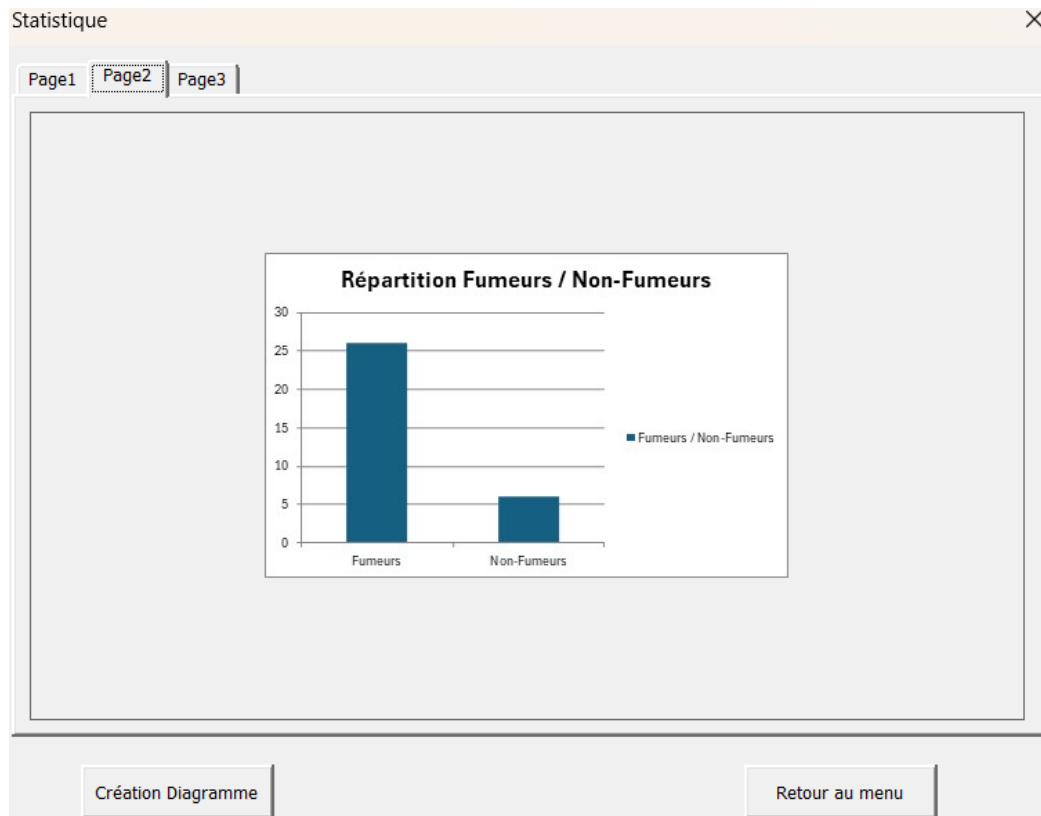




Un message nous confirme l'enregistrement des images et qu'elles sont importés :

 Chart_Age	✓	16/01/2025 19:27	Fichier JPG	16 Ko
 Chart_Class	✓	16/01/2025 19:27	Fichier JPG	16 Ko
 Chart_Fumeur	✓	16/01/2025 19:27	Fichier JPG	16 Ko





## X. Options

Options

Taux de réduction

Pour les plus de 60 ans

Pour les moins de 10 ans

Prix du km par classe

1ère classe

2ème classe

Taux supplémentaire pour les fumeurs

Modifier

Retour au menu

Voici la fenêtre options qui permet de modifier les taux de réductions et majoration selon les tranches d'âges, la classe ou si on est fumeur.

Les paramètres sont enregistrés dans une feuille paramètre.

Réduction Enfant	Réduction Seniors	Majoration 1ère Classe	Majoration 2ème classe	Majoration Fumeur
0,6	0,4	1,35	1,2	1,1

Allons modifier les paramètres :

Options

Taux de réduction

Pour les plus de 60 ans

Pour les moins de 10 ans

Prix du km par classe

1ère classe

2ème classe

Taux supplémentaire pour les fumeurs

Modifier

Retour au menu

(Il est important de mettre une virgule et non un point pour éviter des problèmes de compatibilité)

Après avoir appuyer sur modifier, un message apparaît :

Microsoft Excel X



Options modifiées avec succès!

OK

Réduction Enfant	Réduction Seniors	Majoration 1ère Classe	Majoration 2ème classe	Majoration Fumeur
0,2	0,1	2,35	1,1	0,1

Code en entier :

```
' ***** UserForm1: Menu Principal *****
```

```
Private Sub btnGestionDesRoutes_Click()
```

```
    Unload Me
```

```
    UserForm5.Show
```

```
End Sub
```

```
Private Sub btnAchatBillet_Click()
```

```
    Unload Me
```

```
    UserForm2.Show
```

```
End Sub
```

```
Private Sub btnStatistiques_Click()
```

```
    Unload Me
```

```
    UserForm9.Show
```

```
End Sub
```

```
Private Sub btnOptions_Click()
```

```
    Unload Me
```

```
    UserForm10.Show
```

```
End Sub
```

```
Private Sub btnQuitter_Click()
```

```
    Unload Me
```

```
End Sub
```

```
' ***** UserForm2: Achat billet *****
```

```
Private Sub UserForm_Initialize()
```

```
    ' Remplir le ComboBox de départ avec les valeurs uniques de la colonne  
    "Départ"
```

```
    Dim i As Long
```

```
    Dim departUnique As Object
```

```
    Set departUnique = CreateObject("Scripting.Dictionary")
```

```
    With Feuil1
```

```
        ' Effacer les éléments précédents dans le ComboBox
```

```
        cboDepart.Clear
```

```
        ' Ajouter les départs uniques dans le Dictionary
```

```
        For i = 2 To .Cells(.Rows.Count, 1).End(xlUp).Row
```

```
            departUnique(.Cells(i, 1).Value) = 1 ' Stocke la valeur unique, l'indice  
est ignoré
```

```
        Next i
```

```
        ' Ajouter les départs uniques au ComboBox
```

```
        Dim key As Variant
```

```
        For Each key In departUnique.Keys
```

```
            cboDepart.AddItem key
```

```
        Next key
```

```
    End With
```

```
End Sub
```

```

Private Sub cboDepart_Change()
    ' Remplir le ComboBox d'arrivée en fonction du départ sélectionné

    Dim i As Long

    Dim alreadyExists As Boolean

    cboArrivee.Clear

    If cboDepart.Value <> "" Then
        With Feuil1
            ' Ajouter les arrivées correspondant au départ sélectionné

            For i = 2 To .Cells(.Rows.Count, 1).End(xlUp).Row

                If .Cells(i, 1).Value = cboDepart.Value Then
                    ' Vérifier si l'arrivée est déjà dans la liste

                    alreadyExists = False

                    ' Parcours de tous les éléments de cboArrivee pour vérifier la
présence

                    For j = 0 To cboArrivee.ListCount - 1

                        If cboArrivee.List(j) = .Cells(i, 2).Value Then

                            alreadyExists = True

                            Exit For

                        End If

                    Next j

                    ' Ajouter l'arrivée si elle n'est pas déjà présente

                    If Not alreadyExists Then

                        cboArrivee.AddItem .Cells(i, 2).Value

                    End If

                End If

            End If

        End With
    End If
End Sub

```

```

        Next i
    End With
End If
End Sub

Private Sub btnAcheter_Click()
    ' Validation des données
    If cboDepart.Value = "" Or cboArrivee.Value = "" Or txtAge.Text = "" Then
        MsgBox "Veuillez remplir tous les champs.", vbExclamation
        Exit Sub
    End If

    ' Enregistrer l'état des options de classe dans les variables publiques
    opt1ereClasseSelected = opt1ereClasse.Value
    opt2emeClasseSelected = opt2emeClasse.Value

    ' Vérification des options Fumeur/Non Fumeur et 1ère/2ème classe
    Dim i As Long
    Dim depart As String, arrivee As String
    Dim fumeurPossible As Boolean, nonFumeurPossible As Boolean
    Dim premiereClassePossible As Boolean, deuxiemeClassePossible As Boolean

    depart = cboDepart.Value
    arrivee = cboArrivee.Value

```



' Initialiser les variables à False

fumeurPossible = False

nonFumeurPossible = False

premiereClassePossible = False

deuxiemeClassePossible = False

' Vérifier les options disponibles dans le tableau

With Feuil1

For i = 2 To .Cells(.Rows.Count, 1).End(xlUp).Row

If .Cells(i, 1).Value = depart And .Cells(i, 2).Value = arrivee Then

' Vérification des options Fumeur/Non fumeur

If .Cells(i, 5).Value = "Oui" Then fumeurPossible = True

If .Cells(i, 6).Value = "Oui" Then nonFumeurPossible = True

' Vérification des classes disponibles

If .Cells(i, 3).Value = "Oui" Then premiereClassePossible = True

If .Cells(i, 4).Value = "Oui" Then deuxiemeClassePossible = True

End If

Next i

End With

' Vérification de la compatibilité avec les options sélectionnées

If optFumeur.Value And Not fumeurPossible Then

MsgBox "Option Fumeur non disponible pour ce trajet.", vbExclamation

optFumeur.Value = False

End If

If optNonFumeur.Value And Not nonFumeurPossible Then

MsgBox "Option Non Fumeur non disponible pour ce trajet.",  
vbExclamation

optNonFumeur.Value = False

End If

If opt1ereClasse.Value And Not premiereClassePossible Then

MsgBox "1ère classe non disponible pour ce trajet.", vbExclamation

opt1ereClasse.Value = False

End If

If opt2emeClasse.Value And Not deuxiemeClassePossible Then

MsgBox "2ème classe non disponible pour ce trajet.", vbExclamation

opt2emeClasse.Value = False

End If

' Vérification si aucune option n'est sélectionnée

If Not (optFumeur.Value Or optNonFumeur.Value) Then

MsgBox "Veuillez choisir entre Fumeur ou Non Fumeur.", vbExclamation

Exit Sub

End If

```
If Not (opt1ereClasse.Value Or opt2emeClasse.Value) Then  
    MsgBox "Veuillez choisir entre 1ère ou 2ème classe.", vbExclamation  
    Exit Sub
```

```
End If
```

```
' Transférer les données vers UserForm3
```

```
With UserForm3
```

```
    .lblDepart.Caption = cboDepart.Value
```

```
    .lblArrivee.Caption = cboArrivee.Value
```

```
' Gestion des classes
```

```
If opt1ereClasse.Value Then
```

```
    .lblClasse.Caption = "1ère classe"
```

```
Elseif opt2emeClasse.Value Then
```

```
    .lblClasse.Caption = "2ème classe"
```

```
End If
```

```
' Gestion des options Fumeur / Non Fumeur
```

```
If optFumeur.Value Then
```

```
    .lblWagon.Caption = "Fumeur"
```

```
Elseif optNonFumeur.Value Then
```

```
    .lblWagon.Caption = "Non fumeur"
```

```
End If
```

```
' Gestion du sexe
```

```
If optHomme.Value Then  
    .lblSexe.Caption = "Homme"  
Elseif optFemme.Value Then  
    .lblSexe.Caption = "Femme"  
End If
```

```
.lblAge.Caption = txtAge.Text
```

```
End With
```

```
Me.Hide
```

```
UserForm3.Show
```

```
End Sub
```

```
Private Sub btnRetourMenu_Click()
```

```
    Unload Me
```

```
    UserForm1.Show
```

```
End Sub
```

' \*\*\*\*\* UserForm3 : Données correcte ? \*\*\*\*\*

Private Sub btnValider\_Click()

' Vérification des champs obligatoires

If lblAge.Caption = "" Then

MsgBox "Veuillez remplir tous les champs.", vbExclamation

Exit Sub

End If

' Variables pour récupérer les réductions et majorations

Dim reductionEnfant As Double, reductionSeniors As Double

Dim majorationClasse1 As Double, majorationClasse2 As Double,  
majorationFumeur As Double

' Lecture des valeurs depuis Excel (Feuil1)

With Worksheets("Parametres")

reductionEnfant = .Cells(2, 1).Value ' Réduction Enfant (Colonne 1)

reductionSeniors = .Cells(2, 2).Value ' Réduction Seniors (Colonne 2)

majorationClasse1 = .Cells(2, 3).Value ' Majoration 1ère Classe  
(Colonne 3)

majorationClasse2 = .Cells(2, 4).Value ' Majoration 2ème Classe  
(Colonne 4)

majorationFumeur = .Cells(2, 5).Value ' Majoration Fumeur (Colonne 5)

End With

' Variables pour appliquer les réductions et majorations en fonction de l'option choisie

Dim totalPrix As Double

totalPrix = 50 ' Exemple de prix de base (peut être ajusté)

' Appliquer la réduction en fonction de l'âge (exemple simplifié)

If Val(lblAge.Caption) < 18 Then

totalPrix = totalPrix \* (1 - reductionEnfant) ' Réduction enfant

Elseif Val(lblAge.Caption) >= 65 Then

totalPrix = totalPrix \* (1 - reductionSeniors) ' Réduction seniors

End If

' Appliquer les majorations en fonction de la classe choisie

If lblClasse.Caption = "1ère classe" Then

totalPrix = totalPrix \* majorationClasse1 ' Majoration 1ère classe

Elseif lblClasse.Caption = "2ème classe" Then

totalPrix = totalPrix \* majorationClasse2 ' Majoration 2ème classe

End If

' Appliquer la majoration pour les fumeurs

If lblWagon.Caption = "Fumeur" Then

totalPrix = totalPrix \* majorationFumeur ' Majoration fumeur

End If

' Afficher le prix total après application des réductions et majorations

```
MsgBox "Le prix total du billet est : " & totalPrix & "€", vbInformation
```

```
' Transfert des données vers UserForm4 (si nécessaire)
```

```
With UserForm4
```

```
    .lblPrix.Caption = totalPrix & "€"
```

```
End With
```

```
' Masquer le formulaire actuel et afficher UserForm4
```

```
Me.Hide
```

```
UserForm4.Show
```

```
End Sub
```

```
' Fonction pour récupérer la réduction à partir de la feuille "Réductions"
```

```
Private Function GetReduction(nomReduction As String) As Double
```

```
    Dim ws As Worksheet
```

```
    Set ws = ThisWorkbook.Sheets("Parametres")
```

```
    Dim i As Long
```

```
    For i = 2 To ws.Cells(Rows.Count, 1).End(xlUp).Row ' Parcourt les lignes  
de la feuille
```

```
        If ws.Cells(i, 1).Value = nomReduction Then
```

```
            GetReduction = ws.Cells(i, 2).Value
```

```
            Exit Function
```

```
        End If
```

```
    Next i
```

' Retourne une valeur par défaut (0) si la catégorie n'est pas trouvée

GetReduction = 0

End Function

' Fonction pour récupérer la majoration à partir de la feuille "Réductions"

Private Function GetMajoration(nomMajoration As String) As Double

Dim ws As Worksheet

Set ws = ThisWorkbook.Sheets("Parametres")

Dim i As Long

For i = 2 To ws.Cells(Rows.Count, 1).End(xlUp).Row ' Parcourt les lignes  
de la feuille

If ws.Cells(i, 1).Value = nomMajoration Then

GetMajoration = ws.Cells(i, 2).Value

Exit Function

End If

Next i

' Retourne une valeur par défaut (0) si la catégorie n'est pas trouvée

GetMajoration = 0

End Function

Private Sub btnRetour\_Click()

Me.Hide



UserForm2.Show

End Sub

' \*\*\*\*\* UserForm4: Prix final \*\*\*\*\*

Private Sub btnAcheter\_Click()

Dim ws As Worksheet

Dim nouvelleLigne As Long

Dim numeroBillet As String

Dim wordApp As Object

Dim wordDoc As Object

Dim cheminModele As String

Dim cheminEnregistrement As String

On Error GoTo Erreur

' Référencer la feuille "Statistique"

Set ws = ThisWorkbook.Sheets("Statistique")

' Déterminer la prochaine ligne vide

nouvelleLigne = ws.Cells(ws.Rows.Count, 1).End(xlUp).Row + 1

' Générer un numéro de billet unique basé sur le numéro de ligne

numeroBillet = "BIL" & Format(nouvelleLigne, "0000") ' Exemple : BIL0001,  
BIL0002...

' Enregistrer les données dans la feuille "Statistique"

ws.Cells(nouvelleLigne, 1).Value = numeroBillet ' Numéro du  
billet

```

ws.Cells(nouvelleLigne, 2).Value = Now                ' Date
ws.Cells(nouvelleLigne, 3).Value = UserForm3.lblDepart.Caption '
Depart
ws.Cells(nouvelleLigne, 4).Value = UserForm3.lblArrivee.Caption '
Arrivee
ws.Cells(nouvelleLigne, 5).Value = UserForm3.lblClasse.Caption '
Classe
ws.Cells(nouvelleLigne, 6).Value = UserForm3.lblWagon.Caption '
Wagon
ws.Cells(nouvelleLigne, 7).Value = UserForm3.lblSexe.Caption    ' Sexe
ws.Cells(nouvelleLigne, 8).Value = UserForm3.lblAge.Caption     ' Age
ws.Cells(nouvelleLigne, 9).Value = Replace(UserForm4.lblPrix.Caption,
"€", "") ' Prix

' Chemin du modèle de billet
cheminModele = "C:\Users\Virtu\OneDrive\Bureau\BUT3\Faqir
VBA\Projet Billet-Train\Billet_vide.docx"

' Chemin pour enregistrer le fichier
cheminEnregistrement = "C:\Users\Virtu\OneDrive\Bureau\BUT3\Faqir
VBA\Projet Billet-Train\Billet_" & numeroBillet & ".docx"

' Vérification du fichier modèle
If Dir(cheminModele) = "" Then
    MsgBox "Fichier modèle introuvable : " & cheminModele, vbCritical
    Exit Sub
End If

```

' Vérification du nom de fichier

If Len(numeroBillet) = 0 Or InStr(numeroBillet, ":") > 0 Then

    MsgBox "Numéro de billet incorrect : " & numeroBillet, vbCritical

    Exit Sub

End If

' Ouvrir Word et charger le modèle

Set wordApp = CreateObject("Word.Application")

wordApp.Visible = False ' Rendre visible pour débogage si nécessaire

Set wordDoc = wordApp.Documents.Open(cheminModele)

' Remplir les champs dans le modèle Word

With wordDoc

    ' Vérifier si le signet existe

    If BookmarkExists(wordDoc, "numbillet") Then

        ' Effacer tout le texte existant dans le signet

        .Bookmarks("numbillet").Range.Text = "" ' Effacer le texte existant

        ' Ajouter uniquement le numéro du billet

        .Bookmarks("numbillet").Range.Text = numeroBillet

    Else

        MsgBox "Le signet 'numbillet' est manquant dans le modèle Word.",  
vbCritical

    End If

```
' Remplir les autres signets comme avant

If BookmarkExists(wordDoc, "depart") Then
.Bookmarks("depart").Range.Text = UserForm3.lblDepart.Caption

If BookmarkExists(wordDoc, "arrivé") Then
.Bookmarks("arrivé").Range.Text = UserForm3.lblArrivee.Caption

If BookmarkExists(wordDoc, "classe") Then
.Bookmarks("classe").Range.Text = UserForm3.lblClasse.Caption

If BookmarkExists(wordDoc, "prix") Then .Bookmarks("prix").Range.Text =
Replace(UserForm4.lblPrix.Caption, "€", "") & " €"

End With
```

```
' Enregistrer le fichier rempli

wordDoc.SaveAs2 cheminEnregistrement

wordDoc.Close

wordApp.Quit

MsgBox "Billet acheté avec succès et sauvegardé sous : " &
cheminEnregistrement, vbInformation
```

```
' Fermer le formulaire actuel et revenir au menu principal

Unload Me

UserForm1.Show

Exit Sub
```

Erreur:

```
MsgBox "Erreur lors de l'enregistrement ou de l'ouverture du modèle Word  
:" & Err.Description, vbCritical
```

```
On Error Resume Next
```

```
MsgBox "Chemin modèle utilisé : " & cheminModele
```

```
If Not wordApp Is Nothing Then wordApp.Quit
```

```
End Sub
```

' Fonction pour vérifier l'existence d'un signet dans le document Word

```
Function BookmarkExists(doc As Object, bookmarkName As String) As  
Boolean
```

```
On Error Resume Next
```

```
BookmarkExists = Not doc.Bookmarks(bookmarkName) Is Nothing
```

```
On Error GoTo 0
```

```
End Function
```

```
Private Sub btnRetourMenu_Click()
```

```
Unload Me
```

```
UserForm1.Show
```

```
End Sub
```

' \*\*\*\*\* UserForm5: Gestion des Routes \*\*\*\*\*

Private Sub btnAjouter\_Click()

    Unload Me

    UserForm6.Show

End Sub

Private Sub btnModifier\_Click()

    Unload Me

    UserForm7.Show

End Sub

Private Sub btnSupprimer\_Click()

    Unload Me

    UserForm8.Show

End Sub

Private Sub btnRetourMenu\_Click()

    Unload Me

    UserForm1.Show

End Sub

' \*\*\*\*\* UserForm6: Ajout d'une Route \*\*\*\*\*

Private Sub btnAjouter\_Click()

Dim ligne As Long

ligne = Feuil1.Cells(Rows.Count, 1).End(xlUp).Row + 1

' Validation des champs obligatoires

If txtDepart.Text = "" Or txtArrivee.Text = "" Then

MsgBox "Veuillez remplir tous les champs obligatoires.", vbExclamation

Exit Sub

End If

' Ajouter les données à la feuille

Feuil1.Cells(ligne, 1).Value = txtDepart.Text

Feuil1.Cells(ligne, 2).Value = txtArrivee.Text

Feuil1.Cells(ligne, 3).Value = If(opt1ereClasse.Value, "Oui", "Non") ' 1ère  
classe

Feuil1.Cells(ligne, 4).Value = If(opt2emeClasse.Value, "Oui", "Non") '  
2ème classe

Feuil1.Cells(ligne, 5).Value = If(optFumeur.Value, "Oui", "Non") ' Fumeur

Feuil1.Cells(ligne, 6).Value = If(optNonFumeur.Value, "Oui", "Non") ' Non-  
fumeur

MsgBox "Route ajoutée avec succès !", vbInformation

' Réinitialisation

txtDepart.Text = ""



```
txtArrivee.Text = ""
```

```
opt1ereClasse.Value = True
```

```
optFumeur.Value = True
```

```
End Sub
```

```
Private Sub btnRetourMenu_Click()
```

```
    Unload Me
```

```
    UserForm5.Show
```

```
End Sub
```

' \*\*\*\*\* UserForm7: Modification de la route \*\*\*\*\*

Private Sub UserForm\_Initialize()

' Remplir le ComboBox de départ avec les valeurs uniques de la colonne  
"Départ"

Dim i As Long

Dim departUnique As Object

Set departUnique = CreateObject("Scripting.Dictionary")

With Feuil1

' Effacer les éléments précédents dans le ComboBox

cboDepart.Clear

' Ajouter les départs uniques dans le Dictionary

For i = 2 To .Cells(.Rows.Count, 1).End(xlUp).Row

departUnique(.Cells(i, 1).Value) = 1 ' Stocke la valeur unique, l'indice  
est ignoré

Next i

' Ajouter les départs uniques au ComboBox

Dim key As Variant

For Each key In departUnique.Keys

cboDepart.AddItem key

Next key

End With

End Sub

```

Private Sub cboDepart_Change()
    ' Remplir le ComboBox d'arrivée en fonction du départ sélectionné

    Dim i As Long

    Dim alreadyExists As Boolean

    cboArrivee.Clear

    If cboDepart.Value <> "" Then
        With Feuil1
            ' Ajouter toutes les arrivées correspondant au départ sélectionné
            For i = 2 To .Cells(.Rows.Count, 1).End(xlUp).Row
                If .Cells(i, 1).Value = cboDepart.Value Then
                    ' Vérifier si l'arrivée est déjà dans la liste
                    alreadyExists = False

                    ' Parcours de tous les éléments de cboArrivee pour vérifier la
présence
                    For j = 0 To cboArrivee.ListCount - 1
                        If cboArrivee.List(j) = .Cells(i, 2).Value Then
                            alreadyExists = True

                            Exit For
                        End If
                    Next j

                    ' Ajouter l'arrivée si elle n'est pas déjà présente
                    If Not alreadyExists Then
                        cboArrivee.AddItem .Cells(i, 2).Value
                    End If
                End If
            End If
        End With
    End If
End Sub

```

```

        Next i
    End With
End If
End Sub

Private Sub cboArrivee_Change()
    ' Mettre à jour les CheckBox en fonction de l'arrivée sélectionnée
    Dim ligne As Range
    Dim depart As String
    Dim arrivee As String
    Dim departFeuille As String
    Dim arriveeFeuille As String
    Dim trouve As Boolean

    ' Initialiser la variable qui indique si une correspondance a été trouvée
    trouve = False

    ' Récupérer les valeurs des ComboBox et les nettoyer (enlever espaces et
    mettre en majuscules)
    depart = UCase(Trim(cboDepart.Value))
    arrivee = UCase(Trim(cboArrivee.Value))

    ' Affichage des valeurs dans la fenêtre immédiate pour vérifier les valeurs
    sélectionnées
    Debug.Print "Départ sélectionné : " & depart & ""
    Debug.Print "Arrivée sélectionnée : " & arrivee & ""

```

' Rechercher la ligne qui correspond au départ et à l'arrivée

With Feuil1

' Affichage de toutes les lignes et vérification des valeurs de départ et d'arrivée

Debug.Print "Début de la recherche dans la feuille..."

' Parcours de chaque ligne

For Each ligne In .Range(.Cells(2, 1), .Cells(.Rows.Count, 1)).Rows

' Retirer les espaces des valeurs de la feuille Excel et les mettre en majuscules

departFeuille = UCase(Trim(ligne.Cells(1, 1).Value))

arriveeFeuille = UCase(Trim(ligne.Cells(1, 2).Value))

' Affichage des valeurs à comparer pour chaque ligne de la feuille

Debug.Print "Ligne " & ligne.Row & ": Départ feuille = " & departFeuille & " | Arrivée feuille = " & arriveeFeuille & ""

' Vérification stricte de la correspondance

If departFeuille = depart And arriveeFeuille = arrivee Then

Debug.Print "Correspondance trouvée pour la ligne " & ligne.Row

' Mettre à jour les CheckBox en fonction des données

opt1ereClasse.Value = (ligne.Cells(1, 3).Value = "Oui")

opt2emeClasse.Value = (ligne.Cells(1, 4).Value = "Oui")

optFumeur.Value = (ligne.Cells(1, 5).Value = "Oui")

optNonFumeur.Value = (ligne.Cells(1, 6).Value = "Oui")

```

        trouve = True ' Une correspondance a été trouvée
        Exit For ' Nous avons trouvé la correspondance, on quitte la boucle
    End If
Next ligne

' Si aucune correspondance trouvée, afficher un message
If Not trouve Then
    Debug.Print "Aucune correspondance trouvée pour ce départ et cette
arrivée."
    MsgBox "L'arrivée ne correspond pas à ce départ.", vbExclamation
End If

Debug.Print "Fin de la recherche dans la feuille..."
End With
End Sub

Private Sub btnModifier_Click()
    Dim ligne As Range
    Dim depart As String
    Dim arrivee As String
    Dim trouve As Boolean

    depart = cboDepart.Value
    arrivee = cboArrivee.Value
    trouve = False

```

' Parcours de toutes les lignes pour chercher une correspondance

With Feuil1

For Each ligne In .Range(.Cells(2, 1), .Cells(.Rows.Count, 1)).Rows

' Vérifier la correspondance entre le départ et l'arrivée

If UCase(Trim(ligne.Cells(1, 1).Value)) = UCase(Trim(depart)) And \_

UCase(Trim(ligne.Cells(1, 2).Value)) = UCase(Trim(arrivee)) Then

' Si une correspondance est trouvée, mettre à jour les options

ligne.Cells(1, 3).Value = If(opt1ereClasse.Value, "Oui", "Non") ' 1ère  
classe

ligne.Cells(1, 4).Value = If(opt2emeClasse.Value, "Oui", "Non") '  
2ème classe

ligne.Cells(1, 5).Value = If(optFumeur.Value, "Oui", "Non") ' Fumeur

ligne.Cells(1, 6).Value = If(optNonFumeur.Value, "Oui", "Non") ' Non-  
fumeur

MsgBox "Route modifiée avec succès!", vbInformation

trouve = True

Exit For

End If

Next ligne

If Not trouve Then

MsgBox "Route introuvable !", vbExclamation

End If

End With

End Sub

Private Sub btnRetourMenu\_Click()

    Unload Me

    UserForm5.Show

End Sub



```
' ***** UserForm8: Suppresion*****
```

```
Private Sub UserForm_Initialize()
```

```
    ' Remplir le ComboBox de départ avec les valeurs uniques de la colonne  
    "Départ"
```

```
    Dim i As Long
```

```
    Dim departUnique As Object
```

```
    Set departUnique = CreateObject("Scripting.Dictionary")
```

```
    With Feuil1
```

```
        ' Effacer les éléments précédents dans le ComboBox
```

```
        cboDepart.Clear
```

```
        ' Ajouter les départs uniques dans le Dictionary
```

```
        For i = 2 To .Cells(.Rows.Count, 1).End(xlUp).Row
```

```
            departUnique(.Cells(i, 1).Value) = 1 ' Stocke la valeur unique, l'indice  
est ignoré
```

```
        Next i
```

```
        ' Ajouter les départs uniques au ComboBox
```

```
        Dim key As Variant
```

```
        For Each key In departUnique.Keys
```

```
            cboDepart.AddItem key
```

```
        Next key
```

```
    End With
```

```
End Sub
```

```

Private Sub cboDepart_Change()
    ' Remplir le ComboBox d'arrivée en fonction du départ sélectionné

    Dim i As Long

    Dim alreadyExists As Boolean

    cboArrivee.Clear

    If cboDepart.Value <> "" Then
        With Feuil1
            ' Ajouter les arrivées correspondant au départ sélectionné

            For i = 2 To .Cells(.Rows.Count, 1).End(xlUp).Row

                If .Cells(i, 1).Value = cboDepart.Value Then
                    ' Vérifier si l'arrivée est déjà dans la liste

                    alreadyExists = False

                    ' Parcours de tous les éléments de cboArrivee pour vérifier la
présence

                    For j = 0 To cboArrivee.ListCount - 1

                        If cboArrivee.List(j) = .Cells(i, 2).Value Then

                            alreadyExists = True

                            Exit For

                        End If

                    Next j

                    ' Ajouter l'arrivée si elle n'est pas déjà présente

                    If Not alreadyExists Then

                        cboArrivee.AddItem .Cells(i, 2).Value

                    End If

                End If

            End If
        End With
    End If
End Sub

```

```

Next i
End With
End If
End Sub

Private Sub btnSupprimer_Click()

    Dim ligne As Range
    Dim depart As String
    Dim arrivee As String
    Dim trouve As Boolean

    depart = cboDepart.Value
    arrivee = cboArrivee.Value
    trouve = False

    ' Parcours de toutes les lignes pour chercher une correspondance
    With Feuil1
        For Each ligne In .Range(.Cells(2, 1), .Cells(.Rows.Count, 1)).Rows
            ' Vérifier la correspondance entre le départ et l'arrivée
            If UCase(Trim(ligne.Cells(1, 1).Value)) = UCase(Trim(depart)) And _
                UCase(Trim(ligne.Cells(1, 2).Value)) = UCase(Trim(arrivee)) Then
                ' Si une correspondance est trouvée, demander la confirmation
                pour supprimer

                If MsgBox("Êtes-vous sûr de vouloir supprimer cette route ?",
vbYesNo + vbQuestion) = vbYes Then

                    ligne.EntireRow.Delete

```

```
        MsgBox "Route supprimée avec succès !", vbInformation
    ' Réinitialiser les ComboBox après suppression
    cboDepart.Clear
    cboArrivee.Clear
    trouve = True
    Exit For ' Une fois la ligne supprimée, on sort de la boucle
End If
End If
Next ligne

' Si aucune correspondance n'est trouvée
If Not trouve Then
    MsgBox "Route introuvable !", vbExclamation
End If
End With
End Sub

Private Sub btnRetourMenu_Click()
    Unload Me
    UserForm5.Show
End Sub
```

' \*\*\*\*\* UserForm9: Statistiques \*\*\*\*\*

Private Sub btnCreerDiagramme\_Click()

' Déclaration des objets et variables

Dim chartAge As ChartObject, chartFum As ChartObject, chartClass As ChartObject

Dim lastRow As Long

Dim statistiqueSheet As Worksheet, graphiqueSheet As Worksheet

Dim tranche0\_10 As Long, tranche11\_30 As Long, tranche31\_64 As Long, tranche65\_99 As Long

Dim fumeurs As Long, nonFumeurs As Long

Dim premiereClasse As Long, deuxiemeClasse As Long

Dim i As Long

Dim age As Long, fumeurNonfumeur As String, classe As String

Dim exportPath As String

' Référence à la feuille "Statistique" (où se trouvent les données)

Set statistiqueSheet = ThisWorkbook.Sheets("Statistique")

' Référence à la feuille "Graphique" (où les graphiques seront créés)

Set graphiqueSheet = ThisWorkbook.Sheets("Graphique")

' Trouver la dernière ligne des données dans la feuille "Statistique"

lastRow = statistiqueSheet.Cells(statistiqueSheet.Rows.Count, "H").End(xlUp).Row

' Initialiser les variables pour les comptages

tranche0\_10 = 0

tranche11\_30 = 0

tranche31\_64 = 0

tranche65\_99 = 0

fumeurs = 0

nonFumeurs = 0

premiereClasse = 0

deuxiemeClasse = 0

' Parcourir les données dans la feuille "Statistique"

For i = 2 To lastRow

  ' Tranches d'âges

  age = statistiqueSheet.Cells(i, "H").Value

  If age >= 0 And age <= 10 Then

    tranche0\_10 = tranche0\_10 + 1

  Elseif age >= 11 And age <= 30 Then

    tranche11\_30 = tranche11\_30 + 1

  Elseif age >= 31 And age <= 64 Then

    tranche31\_64 = tranche31\_64 + 1

  Elseif age >= 65 And age <= 99 Then

    tranche65\_99 = tranche65\_99 + 1

  End If

  ' Fumeurs / Non-fumeurs

  fumeurNonfumeur = statistiqueSheet.Cells(i, "F").Value

```
If fumeurNonfumeur = "Fumeur" Then
```

```
    fumeurs = fumeurs + 1
```

```
Elseif fumeurNonfumeur = "Non fumeur" Then
```

```
    nonFumeurs = nonFumeurs + 1
```

```
End If
```

```
' Classes
```

```
classe = statistiqueSheet.Cells(i, "E").Value
```

```
If classe = "1ère classe" Then
```

```
    premiereClasse = premiereClasse + 1
```

```
Elseif classe = "2ème classe" Then
```

```
    deuxiemeClasse = deuxiemeClasse + 1
```

```
End If
```

```
Next i
```

```
' Supprimer les graphiques existants dans la feuille "Graphique"
```

```
Dim chart As ChartObject
```

```
For Each chart In graphiqueSheet.ChartObjects
```

```
    chart.Delete
```

```
Next chart
```

```
' Créer un graphique des tranches d'âges dans la feuille "Graphique"
```

```
Set chartAge = graphiqueSheet.ChartObjects.Add(Left:=100, Width:=375,  
Top:=75, Height:=225)
```

```
With chartAge.chart
```

```

.ChartType = xlColumnClustered

.HasTitle = True

.ChartTitle.Text = "Répartition par Tranches d'Âge"

.SeriesCollection.NewSeries

.SeriesCollection(1).Name = "Tranches d'Âge"

.SeriesCollection(1).XValues = Array("0-10", "11-30", "31-64", "65-99")

.SeriesCollection(1).Values = Array(tranche0_10, tranche11_30,
tranche31_64, tranche65_99)

End With

```

' Créer un graphique des fumeurs / non-fumeurs dans la feuille "Graphique"

```

Set chartFum = graphiqueSheet.ChartObjects.Add(Left:=100,
Width:=375, Top:=325, Height:=225)

```

```

With chartFum.chart

```

```

.ChartType = xlColumnClustered

.HasTitle = True

.ChartTitle.Text = "Répartition Fumeurs / Non-Fumeurs"

.SeriesCollection.NewSeries

.SeriesCollection(1).Name = "Fumeurs / Non-Fumeurs"

.SeriesCollection(1).XValues = Array("Fumeurs", "Non-Fumeurs")

.SeriesCollection(1).Values = Array(fumeurs, nonFumeurs)

```

```

End With

```

' Créer un graphique des classes dans la feuille "Graphique"



```
Set chartClass = graphiqueSheet.ChartObjects.Add(Left:=100,  
Width:=375, Top:=575, Height:=225)
```

```
With chartClass.chart
```

```
.ChartType = xlColumnClustered
```

```
.HasTitle = True
```

```
.ChartTitle.Text = "Répartition par Classes"
```

```
.SeriesCollection.NewSeries
```

```
.SeriesCollection(1).Name = "Classes"
```

```
.SeriesCollection(1).XValues = Array("1ère classe", "2ème classe")
```

```
.SeriesCollection(1).Values = Array(premiereClasse, deuxiemeClasse)
```

```
End With
```

```
' Définir le chemin d'exportation des images
```

```
exportPath = "C:\Users\Virtu\OneDrive\Bureau\BUT3\Faqir VBA\Projet  
Billet-Train\"
```

```
' Exporter chaque graphique en image (format JPEG)
```

```
On Error Resume Next
```

```
chartAge.chart.Export Filename:=exportPath & "Chart_Age.jpg",  
FilterName:="JPEG"
```

```
chartFum.chart.Export Filename:=exportPath & "Chart_Fumeur.jpg",  
FilterName:="JPEG"
```

```
chartClass.chart.Export Filename:=exportPath & "Chart_Class.jpg",  
FilterName:="JPEG"
```

```
On Error GoTo 0
```

' Importer les images dans les contrôles Image1, Image2 et Image3 de  
UserForm9

' S'assurer que le chemin d'accès est correct et que les images existent

On Error Resume Next ' Ignorer les erreurs si l'image ne se charge pas

UserForm9.Image1.Picture = LoadPicture(exportPath & "Chart\_Age.jpg")

UserForm9.Image3.Picture = LoadPicture(exportPath &  
"Chart\_Fumeur.jpg")

UserForm9.Image4.Picture = LoadPicture(exportPath &  
"Chart\_Class.jpg")

On Error GoTo 0 ' Réactiver la gestion des erreurs standard

' Afficher un message de confirmation

MsgBox "Les graphiques ont été exportés en images JPEG et importés  
dans UserForm9 !", vbInformation

End Sub

Private Sub btnRetourMenu\_Click()

Unload Me

UserForm1.Show

End Sub

```
' ***** UserForm10: Options *****
```

```
Private Sub btnModifier_Click()
```

```
    ' Validation des entrées utilisateur
```

```
    If Not IsNumeric(txtReductionEnfant.Text) Or  
    CDBl(txtReductionEnfant.Text) < 0 Then
```

```
        MsgBox "Veuillez entrer une valeur numérique valide pour la réduction  
enfant.", vbExclamation
```

```
        Exit Sub
```

```
    End If
```

```
    If Not IsNumeric(txtReductionSeniors.Text) Or  
    CDBl(txtReductionSeniors.Text) < 0 Then
```

```
        MsgBox "Veuillez entrer une valeur numérique valide pour la réduction  
seniors.", vbExclamation
```

```
        Exit Sub
```

```
    End If
```

```
    If Not IsNumeric(txtMajorationClasse1.Text) Or  
    CDBl(txtMajorationClasse1.Text) < 0 Then
```

```
        MsgBox "Veuillez entrer une valeur numérique valide pour la majoration  
1ère classe.", vbExclamation
```

```
        Exit Sub
```

```
    End If
```

```
    If Not IsNumeric(txtMajorationClasse2.Text) Or  
    CDBl(txtMajorationClasse2.Text) < 0 Then
```

MsgBox "Veuillez entrer une valeur numérique valide pour la majoration  
2ème classe.", vbExclamation

Exit Sub

End If

If Not IsNumeric(txtMajorationFumeur.Text) Or  
CDBl(txtMajorationFumeur.Text) < 0 Then

MsgBox "Veuillez entrer une valeur numérique valide pour la majoration  
fumeur.", vbExclamation

Exit Sub

End If

' Mise à jour des valeurs dans Excel (Feuil1)

With Worksheets("Parametres")

' Modifier les valeurs dans la ligne 2, colonnes correspondantes

.Cells(2, 1).Value = CDBl(txtReductionEnfant.Text) ' Réduction Enfant  
(Colonne 1)

.Cells(2, 2).Value = CDBl(txtReductionSeniors.Text) ' Réduction Seniors  
(Colonne 2)

.Cells(2, 3).Value = CDBl(txtMajorationClasse1.Text) ' Majoration 1ère  
Classe (Colonne 3)

.Cells(2, 4).Value = CDBl(txtMajorationClasse2.Text) ' Majoration 2ème  
Classe (Colonne 4)

.Cells(2, 5).Value = CDBl(txtMajorationFumeur.Text) ' Majoration  
Fumeur (Colonne 5)

End With

' Message de confirmation

MsgBox "Options modifiées avec succès!", vbInformation

End Sub

Private Sub btnRetourMenu\_Click()

' Retour au menu principal

Unload Me

UserForm1.Show

End Sub