UNIVERSITÉ PARIS 1 – PANTHÉON - SORBONNE –

L2 Sciences Economiques

Initiation à l'informatique - Mme Plihon-Heiwy

Epreuve sans document - Durée : 1 heure 30

Correction de l'examen du 14 Janvier 2015

Remarques importantes

- Les téléphones portables devront être arrêtés.
- Aucun document, aucune calculette et aucun ordinateur n'est autorisé.
- Les cartables et sacs doivent être rangés dans les couloirs. Ne conserver sur les pupitres que vos instruments d'écriture.

Pour que votre note reflète la totalité de vos connaissances, vous devez :

- Lire entièrement l'énoncé avant de répondre aux questions.
- Répondre aux questions dans l'ordre dans lequel elles sont posées, et rappeler leurs numéros, sinon une pénalité sera appliquée.
- Justifier chaque réponse par une explication claire, courte et néanmoins exhaustive. Les réponses seront notées en fonction de la qualité de l'explication, une réponse sans explication n'apportera aucun point.
- Ecrire lisiblement : comme l'écrivait l'illustre Maurice Allais, prix Nobel d'Economie, professeur de l'ancienne Faculté de Droit, sur ses textes d'examen, « les copies illisibles ne seront pas lues ».
- **Le projet VBA présenté ci-dessous comporte quatre feuilles de calcul nommées « Interface », « CatalogueVins », « HistoriqueCommandes » et « Calculs » ainsi que deux modules.**
  - **Le module nommé « ModulePrincipal », sur lequel porte les questions de cet examen, contient la fonction « totalTTCreduit » et les quatre procédures « initialiserFiltrage », « filtrageVins », « nouvelleCommande », « historiser ».**
  - *Le module nommé « ModuleComplementaire », fourni dans l'annexe 1, pour information, contient deux fonctions «PrixHT », « MontantTVA » et deux procédures « listerTypesVins » et « historiserEtReinitialiserCommande ».*
- ***Les questions 1 à 5 portent sur VBA. La question 6 porte sur Excel.***
- ***Les numéros de ligne utilisés dans les procédures ont été ajoutés pour faciliter la compréhension des questions du sujet.***
- ***Le barème est indicatif.***

Contexte:

Vers la mi-septembre, des foires aux vins sont organisées par des supermarchés en France. Pour faciliter la gestion des commandes durant ces foires, un fichier .xlsm a été élaboré.

L'objectif de ce fichier est:

- D'aider le consommateur à trouver les vins qu'il souhaite acheter en sélectionnant une couleur (rouge, blanc, rosé) ou un type de vin (par exemple bordeaux, beaujolais, etc.);
- De saisir les commandes;
- D'historiser les commandes passées.

La foire aux vins propose cette année une sélection de 13 vins à ses clients. Pour simplifier, on considère que cette limite est fixe.

Pendant la foire aux vins, une réduction est proposée aux clients.

Scanned by CamScanner

4 feuilles de calcul : « Interface », « CatalogueVins », « HistoriqueCommandes » et « Calculs »

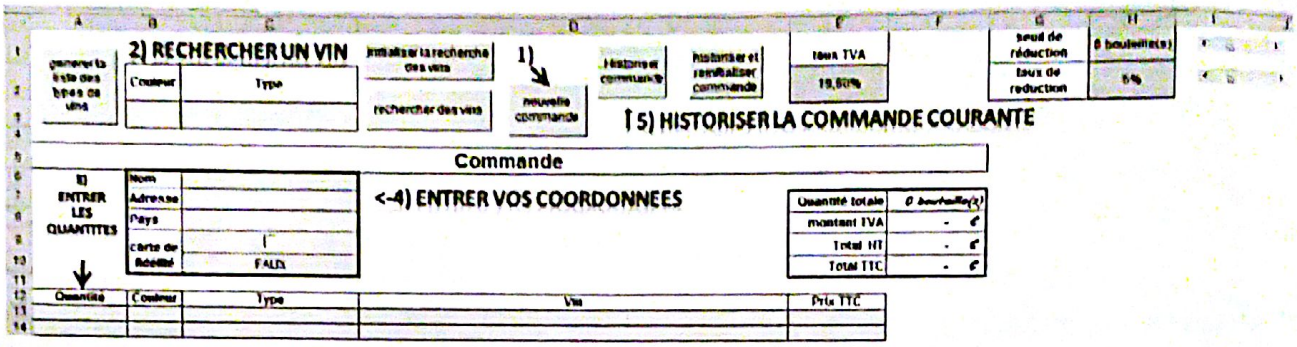


Figure 1 : La feuille « Interface »

N.B. : Dans la feuille "Interface" : 2 "barres de défilement" ont été créées pour permettre de modifier les conditions d'applications de la réduction (le seuil de réduction peut varier de 0 à 20 bouteilles et le taux de réduction peut varier de 0% à 20%).

La feuille « interface » comporte 6 boutons, chacun associés à une macro, comme présenté dans le tableau ci-dessous :

Nom du Bouton	*Macro associée*
generer la liste des types de vins	*listerTypesVin()*
Initialiser la recherche des vins	*initialiserFiltrage()*
Recherche des vins	*filtrageVins()*
Nouvelle commande	*nouvelleCommande()*
Historiser commande	*Historiser()*
Historiser et reinitialiser commande	*historiserEtReinitialiserCommande*

Table 1 : correspondance entre un bouton de commande et la macro qu'elle déclenche

	A	B	C	D	G	I
1	Couleur	Type	vin	Prix TTC	Type	Couleur
2	rouge	Bordeaux	Château du plantier 2009	2,95 €	Alsace Riesling	Rouge
3	rouge	Bordeaux	Château Les Tuileries 2010	3,90 €	Bandol	Blanc
4	rouge	Bordeaux	Château Sainte Marie 2010	5,50 €	Beaujolais	Rosé
5	rouge	Bordeaux	Château Grand Barail 2009	6,50 €	Bergerac	
6	rouge	Bergerac	Château Bel air 2010	2,95 €	Bordeaux	
7	rouge	Buzet	Marquis du Grez 2009	4,95 €	Bourgogne	
8	rouge	Madiran	Château de Viella	5,80 €	Brouilly	
9	rouge	Beaujolais	Domaine Romany 2010	3,95 €		

Figure 2 : La feuille « CatalogueVins »

	A	B	C	D	E	F	G	H
1	*Nom*	*Adresse*	*Pays*	*Carte de fidélité*	*Quantité totale commandée*	*montant TVA*	*Total H.T.*	*Total T.T.C.*

Figure 3 : La feuille « HistoriqueCommandes »

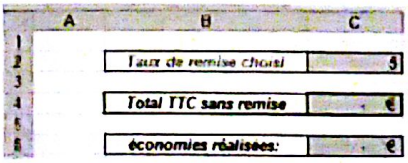
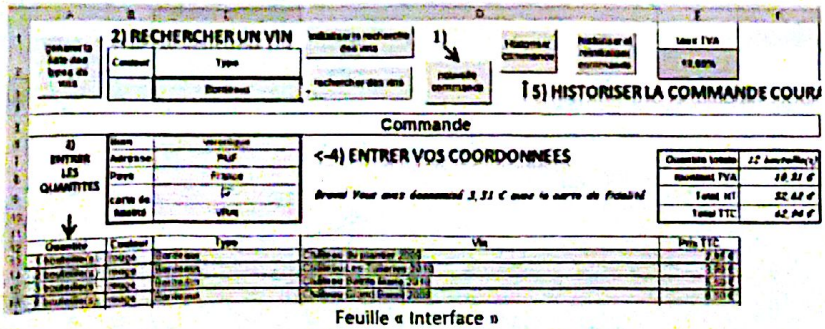
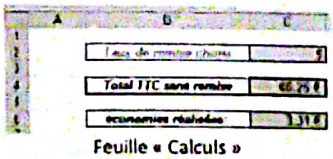


Figure 4 : la feuille « Calculs»



Feuille « Interface »



Feuille « Calculs »

Figure 5 : exemple de commande

La feuille « Calculs » contient des formules Excel qui permettent de connaitre le montant total de la commande TTC hors remise, ainsi que le montant de économies réalisées.

Scanned by CamScanner

Procédures et fonctions du module « ModulePrincipal »

```
Option Explicit
Function totalTTCreduit(Total_PrixTTC As Double, QteCmd As Integer, TauxReduction As Single, _
SeuilReduction As Integer, Fidelité As Boolean) As Double
If Fidelité = True Then
  If QteCmd >= SeuilReduction Then
    totalTTCreduit = Total_PrixTTC * (1 - TauxReduction)
  Else
    totalTTCreduit = Total_PrixTTC
  End If
Else
  totalTTCreduit = Total_PrixTTC
End If
End Function

Sub initialiserFiltrage()
   Range("B3:C3").Select
   Selection.ClearContents
End Sub

Sub filtrageVins()
   Sheets("CatalogueVins").Range("A1:D31").AdvancedFilter Action:=xlFilterCopy, _
      CriteriaRange:=Range("B2:C3"), CopyToRange:=Range("InterfaceIExtract"), _
      Unique:=False
End Sub

Sub nouvelleCommande()
   Sheets("Interface").Select
   Call initialiserFiltrage
   Range("C6:C8").Select
   Selection.ClearContents
   Range("A13:E42").Select
   Selection.ClearContents
End Sub

Sub historiser()
Dim i As Integer, NumLigneStockageNouvelleCommande As Long
Sheets("HistoriqueCommandes").Activate
i = 2
Do While (Not (IsEmpty(Cells(i, 1).Value))) Or (Not (IsEmpty(Cells(i, 2).Value))) Or (Not (IsEmpty(Cells(i, 3).Value))) _
    Or (Not (IsEmpty(Cells(i, 4).Value))) Or (Not (IsEmpty(Cells(i, 5).Value))) Or (Not (IsEmpty(Cells(i, 6).Value))) _
    Or (Not (IsEmpty(Cells(i, 7).Value))) Or (Not (IsEmpty(Cells(i, 8).Value)))
  i = i + 1
Loop
NumLigneStockageNouvelleCommande = i
'MsgBox ("numéro de ligne de stockage de la nouvelle commande = ", & NumLigneStockageNouvelleCommande)

ActiveSheet.Cells(NumLigneStockageNouvelleCommande, 1) = Sheets("Interface").Range("C6")
ActiveSheet.Cells(NumLigneStockageNouvelleCommande, 2) = Sheets("Interface").Range("C7")
ActiveSheet.Cells(NumLigneStockageNouvelleCommande, 3) = Sheets("Interface").Range("C8")
ActiveSheet.Cells(NumLigneStockageNouvelleCommande, 4) = Sheets("Interface").Range("C10")
ActiveSheet.Cells(NumLigneStockageNouvelleCommande, 5) = Sheets("Interface").Range("F7")
ActiveSheet.Cells(NumLigneStockageNouvelleCommande, 6) = Sheets("Interface").Range("F8")
ActiveSheet.Cells(NumLigneStockageNouvelleCommande, 7) = Sheets("Interface").Range("F9")
ActiveSheet.Cells(NumLigneStockageNouvelleCommande, 8) = Sheets("Interface").Range("F10")

Sheets("Interface").Activate
End Sub
```


Scanned by CamScanner

Questions

Question 1 (1 point) :

A la ligne 1, expliquer à quoi sert l'instruction **option explicit.**

Elle sert à rendre obligatoire la déclaration des variables utilisées dans les programmes VBA.

Question 2 (1 point) :

Dans le module « **ModulePrincipal** », citez le nom d'une procédure créée par enregistrement de macro.

initialiserFiltrage ou filtrageVins

Question 3 (4 points)

Dans la fonction « **totalTTCreduit** » (lignes 2 à 13) :

- A quel type d'instructions appartient **if... then... else... end if?**
  C'est une instruction conditionnelle.
- Citez une autre instruction appartenant au même type.
  Select case ... end select
- Sur quelle ligne se trouve le **if** qui est fermé par le **end if** de la ligne 12 ?
  Sur la ligne 4
- Sur quelle ligne se trouve le **if** qui est fermé par le **end if** de la ligne 9 ?
  Sur la ligne 5

- La fonction « **totalTTCreduit** » est appelée dans la formule de la cellule **F10** de la feuille « **Interface** » : **=totalTTCreduit(Calculs!C4;F7;H2;H1;C10).**
- En vous aidant du code VBA de la fonction « **totalTTCreduit** », de la formule saisie dans la cellule **F10** de la feuille « **interface** » et de la **figure 5, explicitez la règle d'attribution de la remise.**
- Si le client possède une carte de fidélité (C10=VRAI), Et s'il a acheté plus de 6 bouteilles de vins (H1), alors il bénéficiera d'une réduction de 5% (H2) ; dans tous les autres cas, pas de réduction.

Question 4 (4 points) :

Concernant la procédure « **nouvelleCommande** » (lignes 23 à 30) :

- Expliquez ce que fait cette procédure, instruction par instruction.
  L23 : début de la procédure nommée nouvelleCommande
  L24 : sélection de la feuille de calcul « interface »
  L25 : Call appelle la procédure initialiser Filtrage, qui elle sélectionne les cellules B3 et C3 et efface leur contenu
  L26 : sélection des cellules C6 à C8
  L27 : effacement des cellules sélectionnées
  L28 : sélection des cellules A13 à E42
  L29 : effacement des cellules sélectionnées
  L30 : fin de la procédure.
- A quoi sert l'instruction **call** (ligne 25) ?
  A appeler une procédure

Question 5 (8 points) :

Concernant la procédure « **historiser** » (lignes 31 à 51) :

- Expliquez **Dim I As Integer, NumLigneStockageNouvelleCommande As Long** (ligne 32). Expliquez en particulier ce qui représente Dim, i, NumLigneStockageNouvelleCommande, integer, Long.
  Dim : mot clé utilisé pour déclarer une variable
  I : nom de variable, de type Integer
  NumLigneStockageNouvelleCommande : nom de variable, de type Long

Integer : type de données entier numérique dont les valeurs sont comprises en -32768 et 32767
Long : type de données numérique dit « entier long » dont les valeurs sont comprises entre **–2 000 000 000** et **+2 000 000 000**, plus précisément entre -2^{31} et $+2^{31}-1$.

- **Cette ligne est-elle obligatoire? Justifiez votre réponse brièvement mais clairement.**
  Oui, à cause de l'instruction option explicit présente sur la première ligne du module.

- **A quel type d'instructions appartient Do While ... Loop (lignes 35 à 39)?**
  Il s'agit d'une instruction répétitive ou boucle
- **Dans quelles conditions l'instruction i=i+1 (ligne 38) est-elle executée ?**
  Si l'une des cellules de chaque ligne de la feuille « HistoriqueCommandes », à partir de la ligne 2 est non vide
- **Expliquez ce que fait l'instruction i=i+1 (ligne 38)**
  On affecte à la variable i son ancienne valeur augmentée de 1. L'ancienne valeur de i est alors perdue.
- **Expliquez l'instruction 'MsgBox ("numero de ligne de stockage de la nouvelle commande= " & NumLigneStockageNouvelleCommande) de la ligne 41, en particulier ', MsgBox, et &.**
- **Quelle est sa particularité à l'exécution ?**
  Cette instruction est précédée d'une **simple cote**, il s'agit donc d'un commentaire, une instruction qui ne sera pas exécutée lors de l'appel du programme.
  Si cette instruction n'était pas en commentaire, elle afficherait une boite de dialogue (grace à la procédure **MsgbBox**) avec le message « numero de ligne de stockage de la nouvelle commande= » suivi de la valeur du numéro de la 1ere ligne entièrement ligne de la feuille « HistoriqueCommandes ».
  Le & est l'opérateur de concaténation. Il permet d'accoler deux chaines de caractères (type String)
- **Expliquez en détail l'instruction ActiveSheet.Cells(NumLigneStockageNouvelleCommande, 1) = Sheets("Interface").Range("C6") de la ligne 42. Expliquez en particulier ce qui représente ActiveSheet, cells ; NumLigneStockageNouvelleCommande, 1, Sheets("Interface") et Range("C6").**
  ActiveSheet : variable predéfinie VBA qui sert à designer la feuille active
  Cells : méthode de la classe worksheet
  NumLigneStockageNouvelleCommande : N° de ligne de la cellule désignée par cells
  1 : N° de colonne de la cellule désignée par cells ; soit ici A
  Sheets("Interface") : objet représentant la feuille « interface »
  Range("C6") : objet désignant la cellule C6, de la feuille « Interface »

  Globalement, cette instruction affecte à la cellule de N° de ligne NumLigneStockageNouvelleCommande et de N° de la colonne de la feuille « HistoriqueCommandes » la valeur de la cellule C6 de la feuille « interface ».
- **Après avoir passé la commande présentée à la figure 5, puis avoir cliqué sur le bouton « historiser Commande » de la feuille «Interface » (voir figure 1 et table 1), quel sera le contenu de la feuille « HistoriqueCommandes »? Vous préciserez les noms des cellules modifiées.**
  En A2 :Veronique
  En B2 : PMF
  En C2 : France
  En D2 : VRAI
  En E2 : 12 bouteille(s)
  En F2 : 10,31€
  En G2 :52,62€
  En H2 : 62,94€

Question 6 (2 points)

Quelle formule Excel faut-il saisir dans la cellule D9 de la feuille « Interface », pour voir apparaitre le message présenté en D9, dans la figure 5? Attention, le message qui s'affiche dans la cellule D9 doit s'adapter en fonction de la commande effectuée.

=si(F10<Calculs !C4 ; « Bravo ! Vous avez économisé »&Calculs !C6& « avec la carte de fidelité » ; « »)

Scanned by CamScanner

Annexe 1

Procédures et fonctions du module « ModuleComplementaire »

```
52. Function PrixHT(PrixTTC As Double, TauxTVA As Single) As Double
53. PrixHT = PrixTTC / (1 + TauxTVA)
54. End Function

55. Function MontantTVA(PrixTTC As Double, TauxTVA As Single) As Double
56. MontantTVA = (PrixTTC * TauxTVA) / (1 + TauxTVA)
57. End Function

58. Sub listerTypesVins()
59.     Sheets("CatalogueVins").Select
60.     Range("B1:B31").AdvancedFilter Action:=xlFilterCopy, CriteriaRange:=Range( _
61.         "F2:F3"), CopyToRange:=Range("G2"), Unique:=True
62.         Range("G3:G26").Select
63.     ActiveWorkbook.Worksheets("CatalogueVins").sort.SortFields.Clear
64.     ActiveWorkbook.Worksheets("CatalogueVins").sort.SortFields.Add Key:=Range( _
65.         "G3:G26"), SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:= _
66.         xlSortNormal
67.     With ActiveWorkbook.Worksheets("CatalogueVins").sort
68.         .SetRange Range("G3:G26")
69.         .Header = xlGuess
70.         .MatchCase = False
71.         .Orientation = xlTopToBottom
72.         .SortMethod = xlPinYin
73.         .Apply
74.     End With
75. End Sub

76. Sub historiserEtReinitialiserCommande()
77. Call historiser
78. Call nouvelleCommande
79. End Sub
```


Scanned by CamScanner