

**Contrôle**  
(Module : Java)  
(2 heures)

**Exercice 1 (3 points) :**

Compléter le tableau suivant :

Vocabulaire	Signification
interface	.....
Exception	.....
private	.....
Héritage	.....

**Exercice 2 (17 points) :**

Un centre d'immatriculation au Maroc souhaite automatiser l'enregistrement des véhicules (Neuf et Occasion) via un système informatique. Ce système doit assurer les tâches suivantes : l'enregistrement des véhicules, l'élaboration des cartes grises, ... etc. Chaque véhicule est identifié par les informations suivantes : Nom et prénom du propriétaire, CIN du propriétaire, Marque, Modèle, Carburant (Gasoil ou Essence), Etat (Neuf ou Occasion), Puissance de fiscale et Immatriculation. La gestion du centre s'effectue à l'aide des classes suivantes :

1) Ecrire une classe Vehicule contenant :

- Le nom et le prénom du propriétaire (nom : une chaîne de caractères) ✓ (5 pts)
- Le numéro de CIN (cin : une chaîne de caractères) ✓
- La marque du véhicule (marque : une chaîne de caractères) ✓
- Le modèle du véhicule (mod[3] : un tableau de type entier) (Exemple : le modèle suivant: 25-4-2015 est stockée comme suit : mod[0] = 25, mod[1] = 4 et mod[2] = 2015) ✓
- Le carburant (Gasoil / Essence) (carb : une chaîne de caractères) ✓
- L'état du véhicule (Neuf / Occasion) (etat : une chaîne de caractères) ✓
- La puissance de fiscalité (fiscale : int) ✓
- L'immatriculation du véhicule (mat[3] : un tableau de type String) (Exemple : l'immatriculation suivante: 1234-A-61 est stockée comme suit : mat[0] = "1234", mat[1] = "A" et mat[2] = "61") ✓
- Un constructeur Vehicule() qui prend en paramètre l'immatriculation du véhicule
- Une méthode clavier() qui permette de saisir et retourner une chaîne de caractères

LST( Info+HEA)+FI

\*\*\*\*\* Bonne chance \*\*\*\*\*



- ✓ Une méthode lecture() qui permet de saisir le reste des informations (nom, cin, marque, mod, carb, etat et fiscale) ✓
- ✓ Une méthode toString() qui permet de retourner une chaîne de caractères contenant toutes les informations du véhicule. ✓

Chaque donnée membre et chaque méthode devront être définies avec les modificateurs suivants : private, public ou static.

- ✓ 2) Tester la classe Vehicule en écrivant une application JAVA (dans une autre classe) capable de créer un objet de type Vehicule, de saisir les informations et d'afficher les informations de l'objet créé. (1 pt)

- 3) Ecrire une classe Centre répondant au cahier des charges suivant (4 pts)

- ✓ Le nombre maximum des véhicules (max: un entier)
- ✓ Un tableau des véhicules (table[]: Vehicule) →
- ✓ Le nombre des véhicules enregistrés (nbe: entier)
- ✓ Un constructeur qui prend en paramètre deux entiers pour initialiser le nombre maximum des véhicules que peut contenir le centre et le nombre des enregistrements
- ✓ Une méthode int recherchevehicule(String[] im) qui recherche un véhicule identifié par l'immatriculation « im » dans le tableau « table ». Cette fonction retourne l'indice dans le cas de réussite et -1 dans le cas inverse. → getElement (int i)
- ✓ Une méthode boolean ajoutevehicule(Vehicule v) qui ajoute un véhicule au tableau « table », la méthode renvoie true si l'ajout s'est correctement effectué, false sinon. Dans le cas où le véhicule existe dans le tableau « table », il faut faire une mise à jour concernant le nouveau propriétaire (nom et cin).
- ✓ Une méthode afficher(Vehicule[] A, int d) qui permettra d'afficher l'ensemble des véhicules enregistrés dans un tableau de type Vehicule

Chaque donnée membre et chaque méthode devront être définies avec les modificateurs suivants : private, public ou static. (1 pt)

- ✓ 4) Modifier l'application JAVA pour tester la classe Centre.

- ✓ 5) Enrichir la classe Centre d'une méthode float fiscale\_prix() qui calcule et retourne le prix total des fiscalités concernant les véhicules enregistrés dans le tableau « table ». (2pts)

Carburant	Puissance Fiscale (PF)			
	PF<8	8≤PF≤10	11≤PF≤14	PF≥15
Essence	350 DH	650 DH	3000 DH	8000 DH
Gasoil	700 DH	1500 DH	6000 DH	20000 DH

- ✓ 6) Ajouter à la classe Centre une méthode Vehicule[] supprime\_vehicules() : dans le tableau « table », supprimer les véhicules qui ont dépassés vingt ans de circulation par rapport à la date 21/11/2016. Cette fonction retourne un tableau contenant les véhicules supprimés. (2 pts)

- ✓ 7) Ajouter à la classe Centre une méthode int fichier\_vehicule(String[] im) qui permet de générer un fichier texte contenant les informations d'un véhicule identifié par l'immatriculation « im ». Cette fonction retourne 1 dans le cas de réussite et 0 dans le cas inverse. throw IOException (2 pts)

\*\*\*\*\* Bonne chance \*\*\*\*\*

LST( Info+IEEA)+FI



## Exercice 1

**Interface** : Une sorte d'interface ne contient que des prototypes de méthodes et des constantes ayant les modificateurs `static` et `final`.

**Exception** : Une interruption de l'exécution d'un programme suite à une erreur.

**private** : Une variable / Une méthode privée ne peut être utilisée que dans les méthodes de même classe.

**Héritage** : Une dérivée hérite des propriétés d'une classe.

## Exercice 2

```
import java.util.*;  
import java.io.*;
```

1%

```
public class Vehicule  
{
```

```
    private String nom;  
    private String cin;  
    private String marque;  
    int[] mod;
```



private String carb;  
private String etat;  
private int fiscale;  
String[] mat = new String[3];  
public Vehicule (String[] mat) // constructeur  
{  
 this.mat = mat;  
 mod = new int[3];  
}

// méthode clavier()  
static String clavier()  
{

Scanner clav = new Scanner(System.in);  
String ch;  
ch = clav.nextLine();  
return ch;  
}

// méthode Lire()

public void lecture()  
{

System.out.println("Veuillez entrer le nom:");  
nom = clavier();  
System.out.println("Veuillez entrer le Cin:");

```

cin = clavier();
System.out.println("La marque de vehicule : ");
marque = clavier();
System.out.println("Veuillez entrer le modèle : ");

for( int i=0 ; i < 3 ; i++)
{
    mod[i] = Integer.parseInt(clavier());
}
System.out.println("Veuillez entrer la puissance
de fiscalite ");
fiscale = Integer.parseInt(clavier());
System.out.println("de Carburant : ");
carb = clavier();
System.out.println("L'etat de vehicule : ");
etat = clavier();
}

```

shs

```

private String carb;
private String etat;
private int fiscale;
String[] mat = new String[3];
public Vehicule (String[] mat) // constructeur
{
    this.mat = mat;
    mod = new int[3];
}
// méthode clavier()
static String clavier()
{
    Scanner clav = new Scanner(System.in);
    String ch;
    ch = clav.nextLine();
    return ch;
}
// méthode Lire()
public void Lecture()
{
    System.out.println("Veuillez entrer le nom:");
    nom = clavier();
    System.out.println("Veuillez entrer le C.in:");

```



```

cin = clavier();
System.out.println("La marque de vehicule :");
marque = clavier();
System.out.println("Veuillez entrer le modèle :");

for( int i=0 ; i < 3 ; i++)
{
    mod[i] = Integer.parseInt(clavier());
}
System.out.println("Veuillez entrer la puissance  
de fiscalite ");
fiscale = Integer.parseInt(clavier());
System.out.println("de Carburant :");
carb = clavier();
System.out.println("L'etat de vehicule :");
etat = clavier();
}

```

```
// méthode toString()  
public String toString()  
{ String ch;
```

```
    ch = " de nom de propriétaire : " + nom + "\n  
    cin : " + cin + "\n da marque de vehicule : " +  
    marque + " de modele : \n" + mod[0] + " - " +  
    mod[1] + " - " + mod[2] + "\n de carburant " +  
    carb + "\n d'état de vehicule : " + etat + " da  
    puissance : " + fiscale + "\n d'immatricule : "  
    + mat[0] + " - " + mat[1] + " - " + mat[2];  
    return ch;  
}
```



*/\* des getters \*/*

```
public String getNom() { return nom; }  
public String getCin() { return cin; }  
public String getMarque() { return marque; }  
public int[] getModel() { return mod; }  
public int getModelElement(intindice) { return mod[indice]; }  
public String getCarb() { return carb; }  
public String getEtat() { return etat; }  
public int getFiscale() { return fiscale; }  
public String[] getMat() { return mat; }  
public String getMatElement(int indice) { return mat[indice]; }
```

*/\* des setters \*/*

```
public void setNom(String nom) { this.nom = nom; }  
public void setCin(String cin) { this.cin = cin; }
```

/\* Classe Centre \*/

```
public class Centre  
{
```

```
    private int max;
```

```
    public Vehicule[] table ;
```

```
    private int nbe ;
```

// Constructeur

```
    public Centre(int max , int nbe)
```

```
    {
```

```
        this.max = max;
```

```
        this.nbe = nbe;
```

```
        table = new Vehicule(max);
```

```
    }
```

/\* la méthode recherchevehicule \*/

```
    public int recherchevehicule(String[] im)
```

```
    {
```

```
        int pos = -1;
```

```
        for(int i=0 ; i < nbe ; i++)
```

```
        {
```

```
            for(int j=0 ; j < 3 ; j++)
```

```
            {
```

```
                if(table[i].getMatElement(j).equals(im[j]))
```

```
                { pos = i ; break ; }
```

```
            }
```

```
        }
```

```
    } return pos;
```

6

/\* la méthode ajoutevehicule \*/

```
{  
public boolean ajoutevehicule(Vehicule V)
```

```
  
    boolean res = false ;
```

```
    int i = recherchevehicule( V.getMat() );
```

```
    if( nbe < max )
```

```
    {
```

```
        if( i == -1 )
```

```
        {
```

```
            table[ nbe ] = V ;
```

```
            nbe ++ ;
```

```
            res = true ;
```

```
        }
```

```
    }  
    elseif( i != -1 )
```

```
    {
```

```
        table[ i ].setNom( V.getNom() );
```

```
        table[ i ].setCin( V.getCin() );
```

```
        res = true ;
```

```
    }
```

```
}
```

```
return res ;
```

```
}
```



/\* La méthode afficher \*/

```
public void afficher ( Vehicule[] A, int d)
{
    System.out.println(" Les vehicules sont : \n");
    for (int i = 0 ; i < d ; i++)
    {
        System.out.println(" Vehicule " + (i+1) + " :");
        System.out.println( A[i].toString() + "\n");
    }
}
```

```
public float fiscale_prix ( )
{
    float prix = 0;
    float totale = 0;
    for ( int i = 0 ; i < nbe ; i++)
    {
        int pf = table[i].getFiscale();
        if ( table[i].getCarb().equals("Essence") )
        {
            if ( pf < 8 ) { prix = 350; }
            elseif ( pf >= 8 && pf <= 10 ) { prix = 650; }
            elseif ( pf >= 11 && pf <= 14 ) { prix = 3000; }
        }
    }
}
```

```
else { prix = 8000 ; }  
}
```

```
if ( table[i].getCarb() . equals ("Gasoil") )
```

```
{  
    if ( pf < 8 ) { prix = 700 ; }
```

```
    elseif ( pf >= 8 && pf <= 10 ) { prix = 1500 ; }
```

```
    elseif ( pf >= 11 && pf <= 14 ) { prix = 6000 ; }
```

```
    else { prix = 20000 ; }
```

```
}
```

```
total = total + prix ;
```

```
} // fin for
```

```
return total ;
```

```
} // fin methode
```

/\* Methode supprime\_vehicule \*/

```
public Vehicule[] supprime_vehicule()  
{  
    Vehicule[] V;  
    V = new Vehicule[nbe];  
    for(int i=0 ; i < nbe ; i++)  
    {  
        for (int j=0 ; j < nbe ; j++)  
        {  
            if (table[i].getModElement[3] == 1996)  
            {  
                V[j] = tab[i];  
                tab[i] = tab[i+1];  
                nbe --;  
            }  
        }  
    }  
    return V;  
}
```



```
/* Méthode fichier_vehicule */  
public int fichier_vehicule(String[] im) throws  
IOException
```

```
{
```

```
    int res = 0;
```

```
    File myFile = new File("Vehicule.txt");
```

```
    FileWriter fw = new FileWriter(myFile);
```

```
    PrintWriter pw = new PrintWriter(fw);
```

```
    int indice = recherchevehicule(im);
```

```
    if (indice != -1)
```

```
    {
```

```
        String ch = table[indice].toString();
```

```
        pw.println(ch);
```

```
        res = 1;
```

```
    }
```

```
    pw.close();
```

```
    return res;
```

```
}
```

```
} // fin classe Centre
```

## // Classe principale

```
public class PrincipalVehicule
{
    public static void main(String[] args)
    {
        /* Q2 */ String[] mat = new String[3];
        System.out.println("Veuillez entrer votre  
immatriculation :");
        for(int i=0 ; i < 3 ; i++)
        {
            mat[i] = clavier();
        }
        Vehicule mycar = new Vehicule(mat);
        mycar.lecture();
        System.out.println(mycar.toString());
        /* Question 4 */
        System.out.println("Veuillez entrer le nombre maximum  
des vehicule :");
        int max = Integer.parseInt(clavier());
        System.out.println("Veuillez entrer le nombre des  
véhicules à enregistrés");
        int nbe = Integer.parseInt(clavier());
    }
}
```

Center c = new Center(max, nbe);

C.ajoutervehicule (mycar);

C.afficher (C.table, nbe);

}

Sana  
KHAYOU