

Exercice 1 :

```
1 public class Cours {
2     private String _code;
3     private String _intitule;
4     private float _volHoraire;
5
6     public Cours(String code, String intitule, int volHoraire){
7         this._code = code;
8         this._intitule = intitule;
9         this._volHoraire = volHoraire;
10    }
11
12    public String getCode(){
13        return this._code;
14    }
15    public String getIntitule(){
16        return this._intitule;
17    }
18    public float getVolHoraire(){
19        return this._volHoraire;
20    }
21    public void setCode(String code){
22        this._code = code;
23    }
24    public void setIntitule(String intitule){
25        this._intitule = intitule;
26    }
27    public void setVolHoraire(float volHoraire){
28        this._volHoraire = volHoraire;
29    }
30
31    public String toString(){
32        return ("Code du cours : " + this._code + " , intitulé du cours : " + this._intitule + " volume horaire : " + this._volHoraire);
33    }
34 }
35 }
```

[1] Implémentation de la classe Cours

Pour les variables de la classe on les précède d'un « _ » pour bien voir que ce sont celles de la classe. Pour les autres méthodes l'implémentation est classique.

Exercice 2 :

```
1  import java.util.ArrayList;
2
3  class Formation
4  {
5      private String _code;
6      private String _nom;
7      private ArrayList<Cours> _listeCours;
8
9      public Formation(String code, String nom, ArrayList<Cours> listeCours){
10         this._code = code;
11         this._nom = nom;
12         this._listeCours = listeCours;
13     }
14     public String getCode(){
15         return this._code;
16     }
17     public String getNom(){
18         return this._nom;
19     }
20     public ArrayList<Cours> getListeCours(){
21         return this._listeCours;
22     }
23     public void setCode(String code){
24         this._code = code;
25     }
26     public void setNom(String nom){
27         this._nom = nom;
28     }
29     public void setListeCours(ArrayList<Cours> listeCours){
30         this._listeCours = listeCours;
31     }
32
33     public String toString(){
34         String text;
35         text = "Code de la formation : " + this._code + ", nom de la formation : " + this._nom + " liste des cours :";
36         for (int k=0;k<_listeCours.size();k++) {
37             text = text + '\n' + "      " + _listeCours.get(k).toString();
38         }
39         return text;
40     }
41     public void addCours(Cours C){
42         _listeCours.add(C);
43     }
44 }
```

[2] Implémentation de la classe Formation

Ici aussi, c'est de l'implémentation classique avec des getters et des setters. De même les noms de variables de la classe sont précédés d'un « _ » pour pouvoir les reconnaître facilement.

Exercice 3 :

```
1  import java.util.ArrayList;
2
3  class Main
4  {
5      public static void main(String[] args) {
6          Cours c1 = new Cours("UE41","Informatique",20);
7          Cours c2 = new Cours("UE42","Electronique Numérique",35);
8          Cours c3 = new Cours("UE51","Ondes et propagation",40);
9          Cours c4 = new Cours("UE52","Certification CCNA1",30);
10         Cours c5 = new Cours("UE61","Traitement du signal",25);
11         Cours c6 = new Cours("UE62","Projet",24);
12         Cours c7 = new Cours("UE71","Anglais",12);
13
14         ArrayList<Cours> listeCours = new ArrayList<Cours>();
15
16         listeCours.add(c1);
17         listeCours.add(c2);
18         listeCours.add(c3);
19         listeCours.add(c4);
20         listeCours.add(c5);
21         listeCours.add(c6);
22         listeCours.add(c7);
23
24         Formation esirem3A = new Formation("3A_S6","Semestre 6, ESIREM",listeCours);
25
26         System.out.println(esirem3A);
27
28         listeCours.remove(1);
29
30         System.out.println(esirem3A);
31         System.out.println(esirem3A.getListeCours().get(1));
32     }
33 }
```

[3] classe Main

On crée 7 cours comme demandé dans le sujet puis fait plusieurs affichages pour constater que le deuxième élément est bien supprimé de la liste des cours.

```
Code de la formation :3A_S6, nom de la formation : Semestre 6, ESIREM liste des cours :
Code du cours : UE41 , intitulé du cours : Informatique, volume horaire : 20.0
Code du cours : UE42 , intitulé du cours : Electronique Numérique, volume horaire : 35.0
Code du cours : UE51 , intitulé du cours : Ondes et propagation, volume horaire : 40.0
Code du cours : UE52 , intitulé du cours : Certification CCNA1, volume horaire : 30.0
Code du cours : UE61 , intitulé du cours : Traitement du signal, volume horaire : 25.0
Code du cours : UE62 , intitulé du cours : Projet, volume horaire : 24.0
Code du cours : UE71 , intitulé du cours : Anglais, volume horaire : 12.0
Code de la formation :3A_S6, nom de la formation : Semestre 6, ESIREM liste des cours :
Code du cours : UE41 , intitulé du cours : Informatique, volume horaire : 20.0
Code du cours : UE51 , intitulé du cours : Ondes et propagation, volume horaire : 40.0
Code du cours : UE52 , intitulé du cours : Certification CCNA1, volume horaire : 30.0
Code du cours : UE61 , intitulé du cours : Traitement du signal, volume horaire : 25.0
Code du cours : UE62 , intitulé du cours : Projet, volume horaire : 24.0
Code du cours : UE71 , intitulé du cours : Anglais, volume horaire : 12.0
Code du cours : UE51 , intitulé du cours : Ondes et propagation, volume horaire : 40.0
```

[4] résultat obtenu après l'exécution de la classe Main

Exercice 4 :

```
1  public class Cours implements Comparable{
2      private String _code;
3      private String _intitule;
4      private float _volHoraire;
5
6      public Cours(String code, String intitule, int volHoraire){
7          this._code = code;
8          this._intitule = intitule;
9          this._volHoraire = volHoraire;
10     }
11
12     @Override
13     public int compareTo(Object obj) {
14         String intitule = ((Cours)obj).getIntitule();
15         return this._intitule.compareTo(intitule);
16     }
```

[5] Ajout de la méthode compareTo à la class Cours

```
1  import java.util.ArrayList;
2  import java.util.Collections;
3
4  class Main
5  {
6      public static void main(String[] args) {
7          Cours c1 = new Cours("UE41", "Informatique", 20);
8          Cours c2 = new Cours("UE42", "Electronique Numérique", 35);
9          Cours c3 = new Cours("UE51", "Ondes et propagation", 40);
10         Cours c4 = new Cours("UE52", "Certification CCNA1", 30);
11         Cours c5 = new Cours("UE61", "Traitement du signal", 25);
12         Cours c6 = new Cours("UE62", "Projet", 24);
13         Cours c7 = new Cours("UE71", "Anglais", 12);
14
15         ArrayList<Cours> listeCours = new ArrayList<Cours>();
16
17         listeCours.add(c1);
18         listeCours.add(c2);
19         listeCours.add(c3);
20         listeCours.add(c4);
21         listeCours.add(c5);
22         listeCours.add(c6);
23         listeCours.add(c7);
24
25         Formation esirem3A = new Formation("3A_S6", "Semestre 6, ESIREM", listeCours);
26
27         //System.out.println(esirem3A);
28         //listeCours.remove(1);
29         //System.out.println(esirem3A);
30         //System.out.println(esirem3A.getListeCours().get(1));
31
32         System.out.println(esirem3A);
33
34         Collections.sort(esirem3A.getListeCours());
35
36         System.out.println(esirem3A);
37     }
38 }
```

[6] Modification de la classe Main

```

Code de la formation :3A_S6, nom de la formation : Semestre 6, ESIREM liste des cours :
  Code du cours : UE41 , intitulé du cours : Informatique, volume horaire : 20.0
  Code du cours : UE42 , intitulé du cours : Electronique Numérique, volume horaire : 35.0
  Code du cours : UE51 , intitulé du cours : Ondes et propagation, volume horaire : 40.0
  Code du cours : UE52 , intitulé du cours : Certification CCNA1, volume horaire : 30.0
  Code du cours : UE61 , intitulé du cours : Traitement du signal, volume horaire : 25.0
  Code du cours : UE62 , intitulé du cours : Projet, volume horaire : 24.0
  Code du cours : UE71 , intitulé du cours : Anglais, volume horaire : 12.0
Code de la formation :3A_S6, nom de la formation : Semestre 6, ESIREM liste des cours :
  Code du cours : UE71 , intitulé du cours : Anglais, volume horaire : 12.0
  Code du cours : UE52 , intitulé du cours : Certification CCNA1, volume horaire : 30.0
  Code du cours : UE42 , intitulé du cours : Electronique Numérique, volume horaire : 35.0
  Code du cours : UE41 , intitulé du cours : Informatique, volume horaire : 20.0
  Code du cours : UE51 , intitulé du cours : Ondes et propagation, volume horaire : 40.0
  Code du cours : UE62 , intitulé du cours : Projet, volume horaire : 24.0
  Code du cours : UE61 , intitulé du cours : Traitement du signal, volume horaire : 25.0

```

[7] Résultat après exécution

Dans la figure [5] on ajoute simplement la méthode qui est donné dans le sujet. Ensuite figure [6] on modifie le main pour afficher la formation avant est après le tri des cours par ordre alphabétique de l'intitulé des cours. Enfin dans la figure [7], on constate bien que les cours on était trié par ordre alphabétique de leur intitulé.