

Exemple pour illustrer les associations UML et les collections Java

Nous donnons ici un petit exemple pour illustrer les associations et leur mise en œuvre en java.

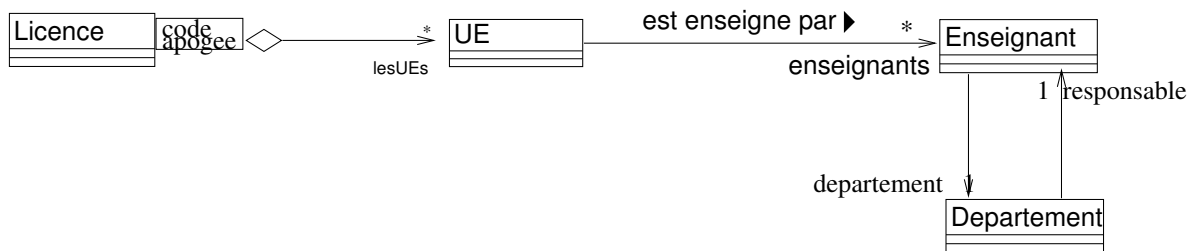


FIG. 1 – Exemple en UML, sans détails

Listing 1 – Licence.java

```

1 package exempleCollections;
2
3 import java.util.*;
4
5 public class Licence{
6     private String specialite;
7     private int nbInscriptions;
8     private Hashtable<String,UE> lesUEs=new Hashtable<String,UE>();
9
10    public String getSpecialite(){
11        return specialite;
12    }
13    public void setSpecialite(String spe){
14        specialite=spe;
15    }
16    public int getNbInscriptions(){
17        return nbInscriptions;
18    }
19    public void setNbInscriptions(int nb){
20        nbInscriptions=nb;
21    }
22
23    public UE getUEByCode(String codeApogee){
24        return lesUEs.get(codeApogee);
25    }
26    public void ajoutUE(UE ue){
27        lesUEs.put(ue.getCodeApogee(), ue);
28    }
29 }

```

Listing 2 – UE.java

```
1 package exempleCollections;
2 import java.util.*;
3
4
5 public class UE{
6     private String codeApogee;
7     private String nom;
8     private Vector<Enseignant> enseignants=new Vector<Enseignant>();
9
10    public String getCodeApogee(){
11        return codeApogee;
12    }
13    public void setCodeApogee(String code){
14        codeApogee=code;
15    }
16    public String getNom(){
17        return nom;
18    }
19    public void setNom(String nom){
20        this.nom=nom;
21    }
22
23    public int nbEnseignants(){
24        return enseignants.size();
25    }
26    public Enseignant getEnseignantAt(int index){
27        return enseignants.get(index);
28    }
29    public void addEnseignant(Enseignant e){
30        enseignants.add(e);
31    }
32    public void trieEnseignantParNom ( ){
33        Collections.sort(enseignants) ;
34    }
35    public void trieEnseignantParDepartement( ){
36        Collections.sort(enseignants , new ComparatorEnseignantParDepartement())
37        ;
38    }
39    public void trieEnseignantParAnciennete ( ){
40        Collections.sort(enseignants , new ComparatorEnseignantParAnciennete ())
41        ;
42    }
}
```

Listing 3 – Enseignant.java

```
1 package exempleCollections;
2 import java.util.*;
3
4
5 public class Enseignant implements Comparable<Enseignant >{
```

```
6  private String nom;
7  private String prenom;
8  private int anneeRecrutement ; // pour determiner les plus anciens pour
   des taches administratives particulieres
9  private Departement departement ;
10
11 public int compareTo (Enseignant e) // sur le nom
12 {
13     return this.getNom().compareTo(e.getNom());
14 }
15
16 public String getNom(){
17     return nom;
18 }
19 public void setNom(String nom){
20     this.nom=nom;
21 }
22 public String getPrenom(){
23     return prenom;
24 }
25 public void setPrenom(String prenom){
26     this.prenom=prenom;
27 }
28
29 public int getAnneeRecrutement() {
30     return anneeRecrutement ;
31 }
32
33 public void setAnneeRecrutement(int anneeRecrutement) {
34     this.anneeRecrutement=anneeRecrutement ;
35 }
36
37 public Departement getDepartement() {
38     return departement ;
39 }
40
41 public void setDepartement(Departement d) {
42     this.departement=d ;
43 }
44
45 public Enseignant(String nom, String prenom, Departement departement,
   int anneeRecrutement) {
46     this.nom=nom ;
47     this.prenom=prenom ;
48     this.departement=departement ;
49     this.anneeRecrutement=anneeRecrutement ;
50 }
51
52 }
```

Listing 4 – Departement.java

```
1 package exempleCollections ;
2
```

```

3
4 public class Departement implements Comparable<Departement>
5 {
6     private String nom;
7     private Enseignant responsable;
8
9     public String getNom() {
10         return nom ;
11     }
12
13     public void setNom (String nom) {
14         this.nom=nom ;
15     }
16
17     public Enseignant getResponsable(){
18         return responsable ;
19     }
20
21     public void setResponsable(Enseignant responsable){
22         this.responsable=responsable;
23     }
24
25     public Departement(String n){
26         nom=n ;
27     }
28
29     public Departement (String n, Enseignant r){
30         nom=n;
31         responsable=r;
32     }
33
34     public int compareTo(Departement d ){ // sur le nom
35         return this.getNom().compareTo(d.getNom());
36     }
37 }

```

Listing 5 – ComparatorEnseignantParAnciennete.java

```

1 package exempleCollections;
2 import java.util.Comparator;
3
4 public class ComparatorEnseignantParAnciennete implements Comparator<
5     Enseignant>
6 {
7     // retourne un nb >0 si e1 est plus ancien que e2
8     public int compare(Enseignant e1, Enseignant e2) {
9         return e1.getAnneeRecrutement()-e2.getAnneeRecrutement();
10    }
11 }

```

Listing 6 – ComparatorEnseignantParDepartement.java

```

1 package exempleCollections;

```

```

2
3 import java.util.Comparator;
4
5 public class ComparatorEnseignantParDepartement implements Comparator<
    Enseignant>
6 {
7     public int compare(Enseignant e1, Enseignant e2){
8         int resultat=e1.getDepartement().compareTo(e2.getDepartement());
9         if (resultat ==0){
10             resultat=e1.compareTo(e2);
11         }
12         return resultat;
13     }
14 }

```

Listing 7 – TestLicence.java

```

1 package exempleCollections;
2
3 import java.util.*;
4
5
6 public class TestLicence{
7
8     public static void main(String [] args){
9         Departement info = new Departement(" Informatique" );
10        Departement math = new Departement(" Mathematique" );
11
12        Licence l=new Licence ();
13        l.setSpecialite(" info");
14        UE ue=new UE();
15        ue.setCodeApogee(" ULIN407");
16        ue.setNom(" Conception et programmation par objets");
17        Enseignant mh=new Enseignant(" Huchard", " Marianne", info, 1992);
18        Enseignant cn=new Enseignant(" Nebut", " Clementine", info, 2005);
19        Enseignant cm=new Enseignant(" Cuer", " Michel", math, 2000);
20        ue.addEnseignant(mh);
21        ue.addEnseignant(cn);
22        ue.addEnseignant(cm);
23        l.ajoutUE(ue);
24
25        UE uebis=l.getUEByCode(" ULIN407");
26        System.out.println(" l'UE de code apogee "+uebis.getCodeApogee()+" se '
27            appelle "+uebis.getNom()+" et est dans la licence "+l.getSpecialite
28            ());
29        System.out.println(" Ses enseignants pour cette annee sont : ");
30        for (int i=0;i<uebis.nbEnseignants();i++){
31            Enseignant e=uebis.getEnseignantAt(i);
32            System.out.println(e.getPrenom()+" "+e.getNom());
33        }
34
35        System.out.println(" Liste trie par dpt -----" );
36        uebis.trieEnseignantParDepartement();
37        for (int i =0; i<uebis.nbEnseignants();i++){

```

```
36     Enseignant e=uebis.getEnseignantAt(i);
37     System.out.println(e.getPrenom()+" "+e.getNom()+" "+e.getDepartement
38         (.getNom()));
39     }
40 }
41 }
```
