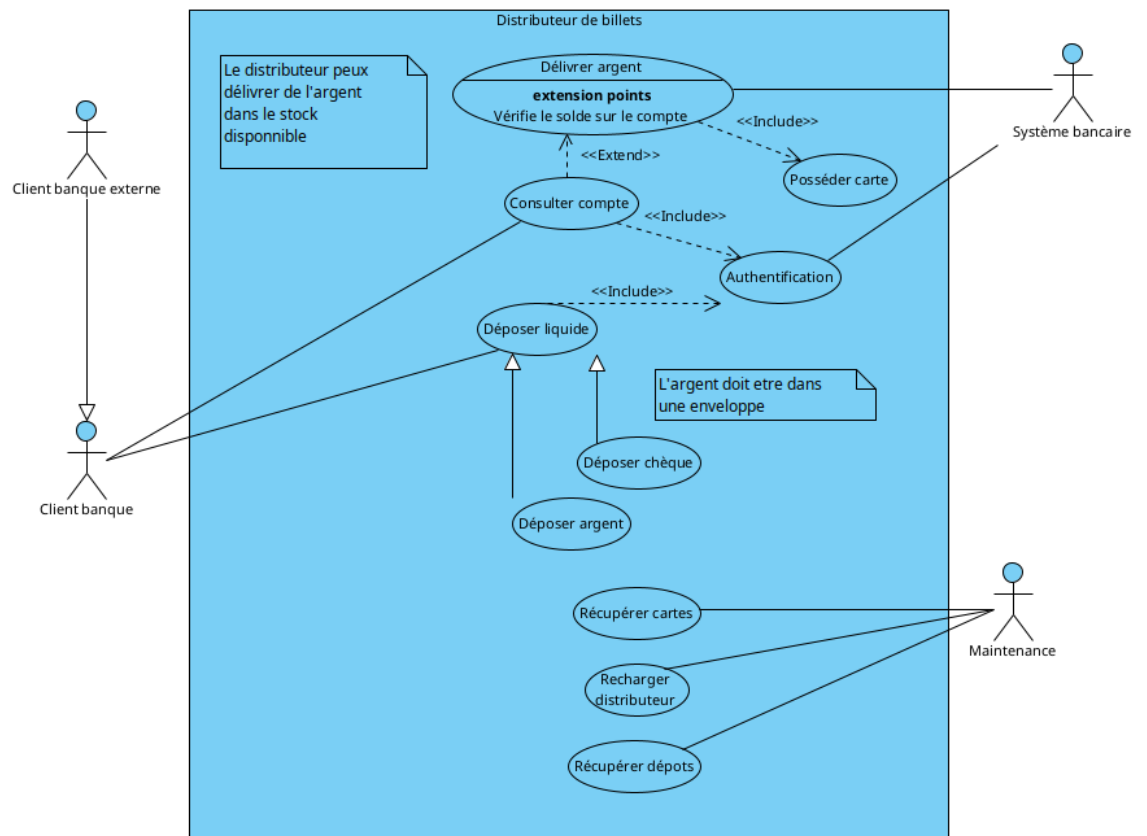


Exercice UML

CDA 25

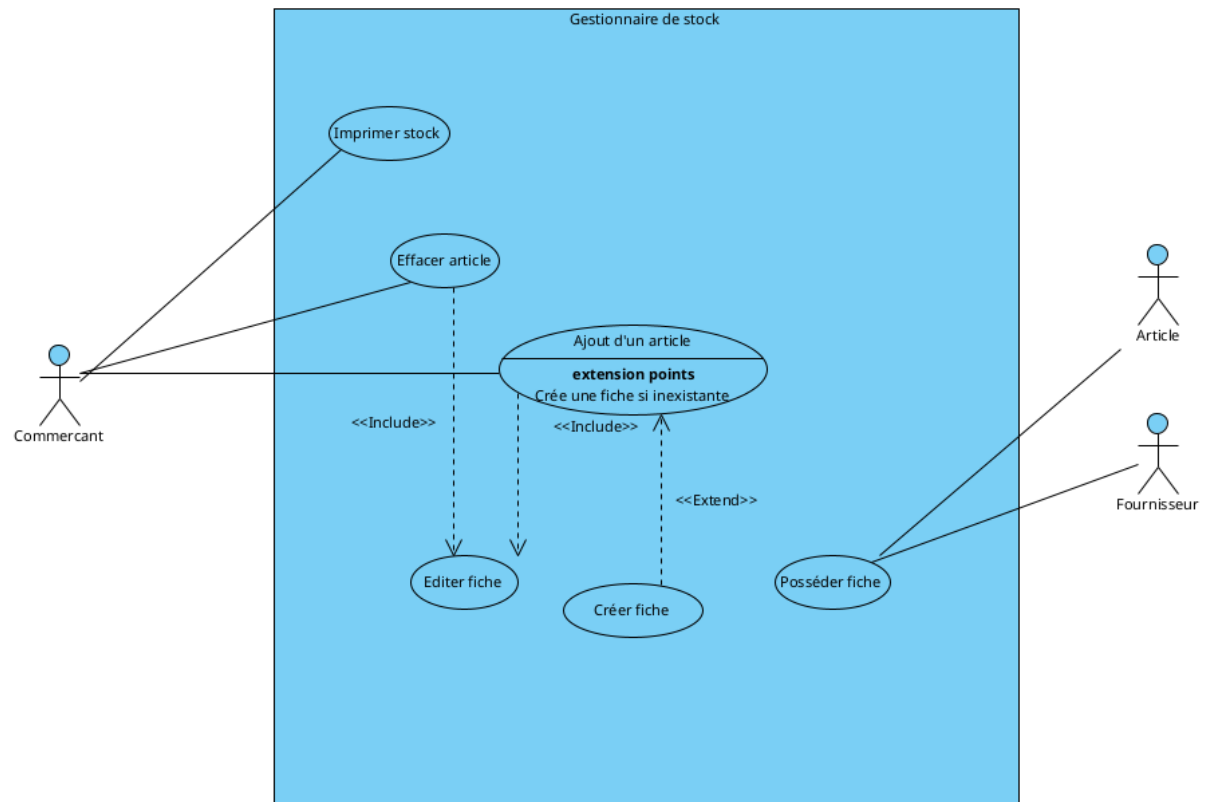
Section 1 Un distributeur de billets :

uc [distributeur_billets]



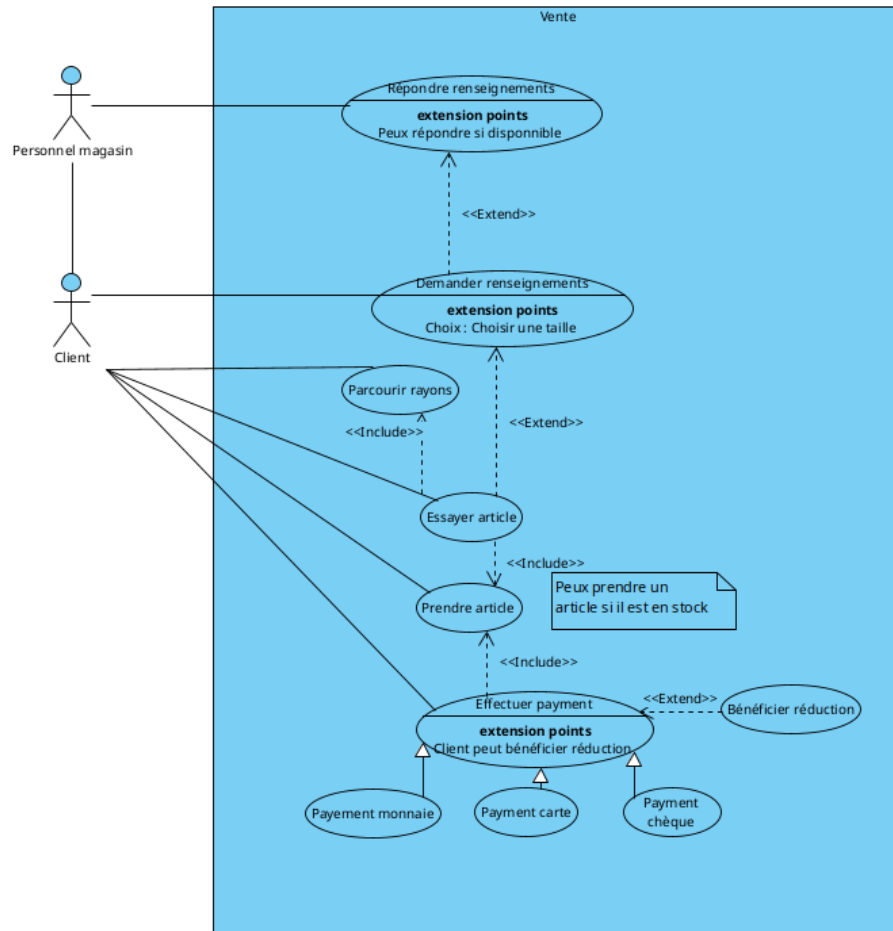
Section 2 Gestion d'articles dans un stock :

uc [gestion_articles]

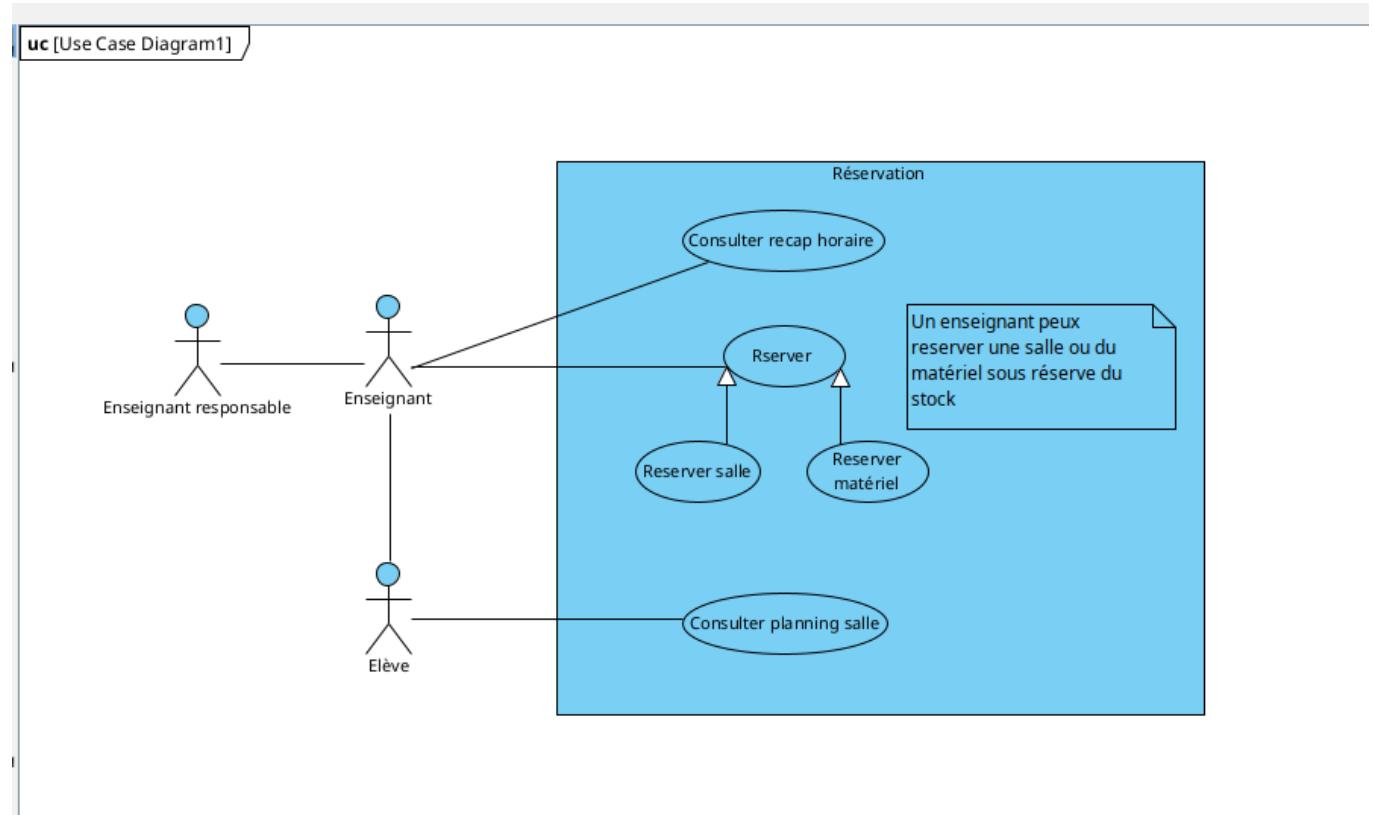


Section 3 Achat dans un magasin :

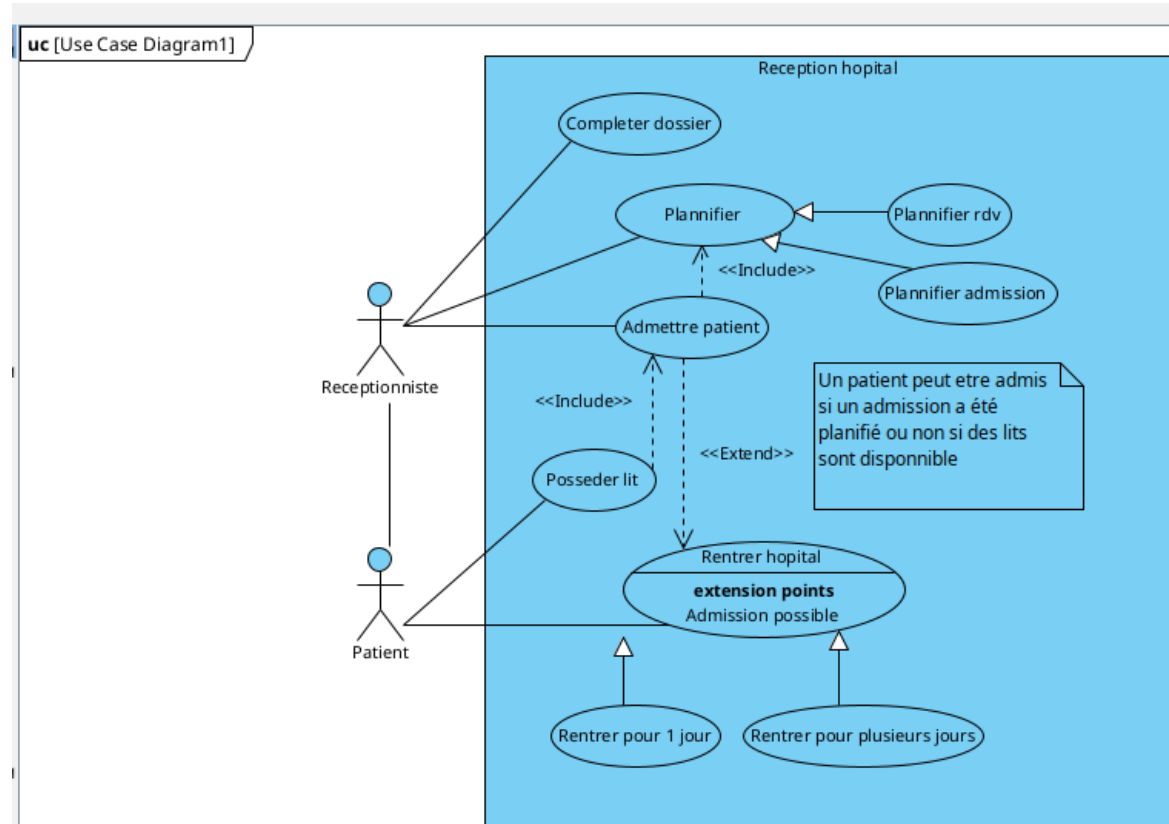
uc [achat]



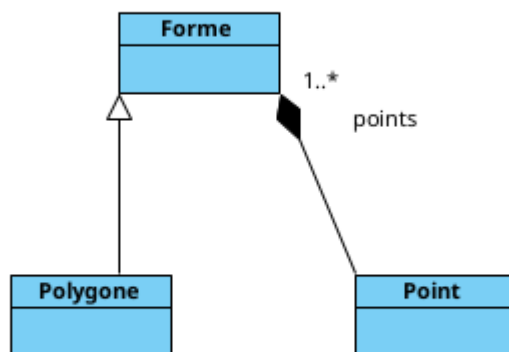
Section 4 Gestion de salle de cours :



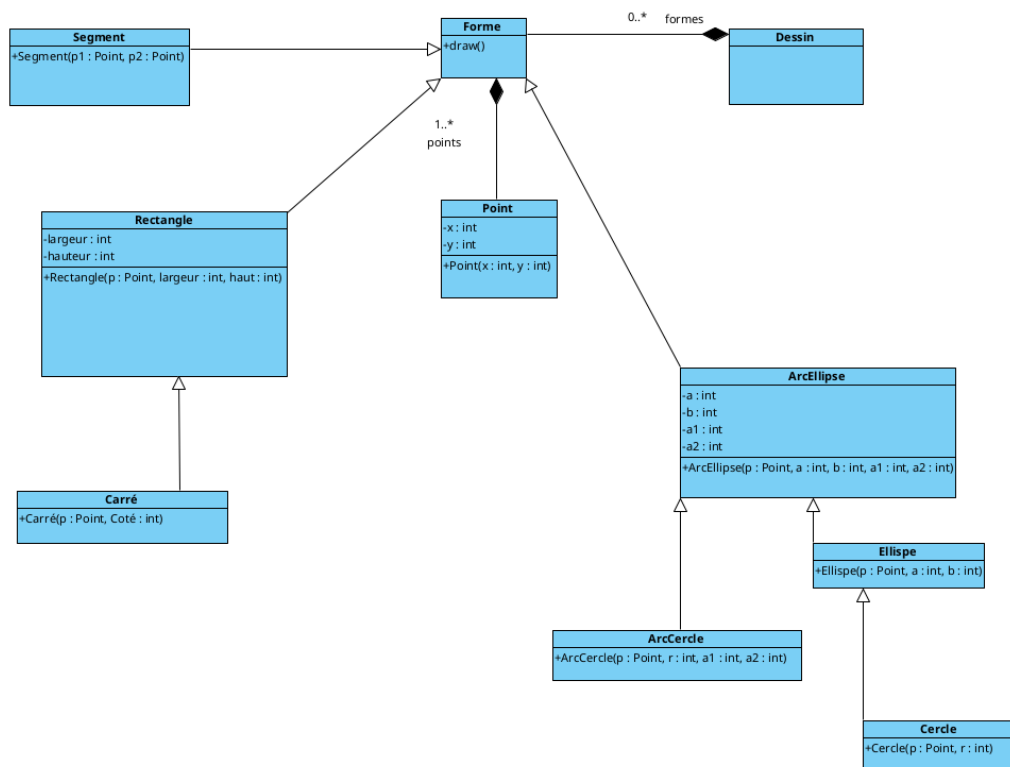
Section 5 Travail du réceptionniste dans un hôpital :



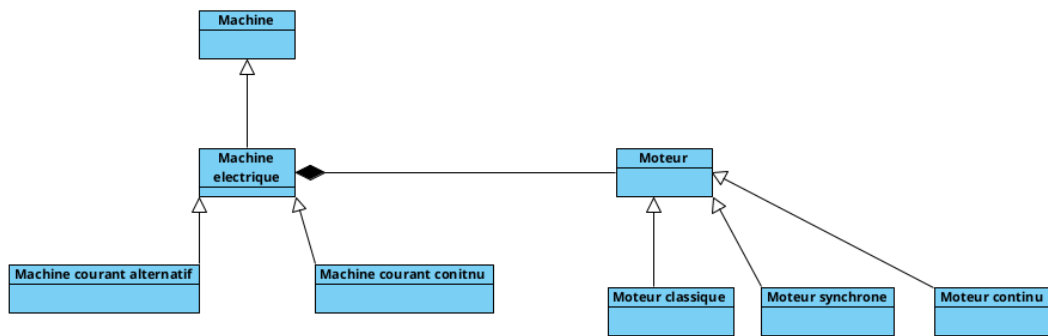
Section 6 Le polygone



Section 7 Les figures



Section 8 Machine électrique

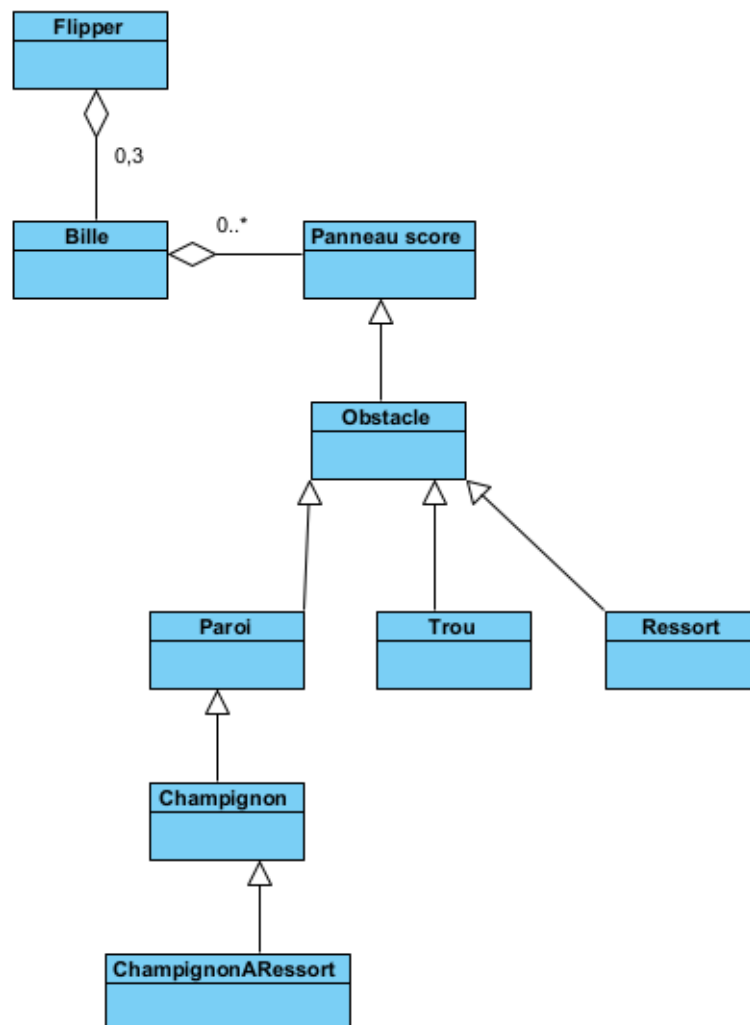


Section 9 Système de réservation

Section 10 Analyse de code

Le code va boucler au là cause de fObj.f();

Section 11 Le Flipper



Section 12 Programmation objet

```
Main.java  Voiture.java  Roue.java
1  import java.util.Arrays;
2
3  public class Voiture { 1 usage
4      private String modele; 2 usages
5      private Roue[] roues; 2 usages
6
7      public Voiture(String modele, double pressionAvant, double pressionArriere) { 1 usage
8          this.modele = modele;
9          this.roues = new Roue[]{
10              new Roue(pressionAvant, position: "Avant Gauche"),
11              new Roue(pressionAvant, position: "Avant Droit"),
12              new Roue(pressionArriere, position: "Arrière Gauche"),
13              new Roue(pressionArriere, position: "Arrière Droit"),
14              new Roue(Math.max(pressionAvant, pressionArriere), position: "De secours")
15          };
16      }
17
18      @Override
19      public String toString() {
20          return "Voiture{" +
21              "modele=" + modele + '\n' +
22              ", roues=" + Arrays.toString(roues) +
23              '\n';
24      }
25  }
```

```
Main.java  Voiture.java  Roue.java
1  public class Roue { 7 usages
2      private double pression; 5 usages
3      private String position; 2 usages
4
5      public Roue(double pression, String position) { 5 usages
6          this.pression = pression;
7          this.position = position;
8      }
9
10     @Override
11     public String toString() {
12         return "Roue{" +
13             "pression=" + pression +
14             ", position=" + position + '\n' +
15             '\n';
16     }
17
18     public void gonfler(double pression) { no usages
19         this.pression += pression;
20     }
21
22     public void degonfler(double pression) { no usages
23         this.pression = Math.max(0, this.pression - pression);
24     }
25 }
```

```
Main.java x  Voiture.java  Roue.java
1  public class Main {
2      public static void main(String[] args) {
3
4          Voiture myCar = new Voiture( modele: "R4", pressionAvant: 2.5, pressionArriere: 2.2);
5          System.out.printf(myCar.toString());
6      }
7  }
```

Section 13 Héritage

```
1 public class VoitureDeSport extends Voiture { 2 usages
2     private double vitesseMax; 2 usages
3
4     public VoitureDeSport(String modele, double pressionAvant, double pressionArriere, double vitesseMax) { 1 usage
5         super(modele, pressionAvant, pressionArriere);
6         this.vitesseMax = vitesseMax;
7     }
8
9     @Override
10    public String toString() {
11        return "VoitureDeSport{" +
12            "vitesseMax=" + vitesseMax +
13            "}" + super.toString();
14    }
15 }
```

```
1 public class Main {
2     public static void main(String[] args) {
3         Voiture myCar = new Voiture( modele: "R4", pressionAvant: 2.5, pressionArriere: 2.2);
4         System.out.printf(myCar.toString() + "\n");
5
6         VoitureDeSport mySuperCar = new VoitureDeSport( modele: "Ferrari", pressionAvant: 2.8, pressionArriere: 2.6, vitesseMax: 350);
7         System.out.printf(mySuperCar.toString() + "\n");
8     }
9 }
10
```

Section 14 Le polymorphisme

```
Main.java x Voiture.java Roue.java VoitureDeSport.java
2 import java.util.List;
3
4 public class Main {
5     public static void main(String[] args) {
6         Voiture myCar1 = new Voiture( modele: "R4", pressionAvant: 2.5, pressionArriere: 2.2);
7         Voiture myCar2 = new Voiture( modele: "C4", pressionAvant: 2.4, pressionArriere: 2.3);
8         Voiture mySuperCar1 = new VoitureDeSport( modele: "Ferrari", pressionAvant: 2.8, pressionArriere: 2.6, vitesseMax: 350);
9         Voiture mySuperCar2 = new VoitureDeSport( modele: "Bugatti", pressionAvant: 3.0, pressionArriere: 2.7, vitesseMax: 320);
10
11         List<Voiture> voitures = new ArrayList<>();
12
13         voitures.add(myCar1);
14         voitures.add(myCar2);
15         voitures.add(mySuperCar1);
16         voitures.add(mySuperCar2);
17
18         for (Voiture voiture : voitures) {
19             System.out.printf(voiture.toString() + "\n");
20         }
21     }
22 }
23
```

Section 15 Un avion en détresse

