```
1
2  public class PatientInfo {
3
4      public static void main(String[] args) {
5          //Creates a new patient with the default constructor
6          Patient Timmy = new Patient();
7
8          //Prints out the information in a formatted way
9          System.out.println("Patient ID: " + Timmy.getID() +
10                 "\nPatientAge: " + Timmy.getAge());
11         //Displays the blood type using the enumerated values
12         Timmy.displayBlood();
13
14         //Creates a new patient using the non default constructor and fills in
15         //information by using enumerated types
16         Patient Spike = new Patient(1337, 19, RhFactor.NEGATIVE, BloodType.AB);
17
18         //Prints out the information in a formatted way
19         System.out.println("PatientID: " + Spike.getID() +
20                 "\nPatient Age: " + Spike.getAge());
21         //Displays the blood type using the enumerated values
22         Spike.displayBlood();
23     }
24
25 }
26
```

```java

public class Patient {

    //Variable list for the patient class
    private int ID_number;
    private int age;
    private BloodData blood_data;

    //Default constructor for the patient class
    Patient(){
        ID_number = 0;
        age = 0;
        blood_data = new BloodData();
    }

    //Non-Default constructor for the patient class, sets the values
    Patient(int IDf, int agef, RhFactor Rhf, BloodType bloodf){
        ID_number = IDf;
        age = agef;
        blood_data = new BloodData(bloodf, Rhf);
    }

    //Returns the ID number
    int getID(){ return ID_number;}

    //Returns the Age
    int getAge() { return age;}

    //Displays the blood information based on enumerated values
    void displayBlood() {System.out.println(blood_data.getBlood());}

    //Class for the blood data
    public class BloodData{

        //values used inside of the class, cannot be accessed unless called by a
        //method in the class
        private BloodType blood_type;
        private RhFactor Rh;

        //Default constructor for the blood data class
        BloodData(){
            blood_type = BloodType.O;
            Rh = RhFactor.POSITIVE;
        }
```

```
45
46          //Non-default constructor for the blood data class
47          BloodData(BloodType bloodf, RhFactor Rhf){
48              blood_type = bloodf;
49              Rh = Rhf;
50          }
51
52          //Creates a string so that it will be formatted when it is printed
53          String getBlood() {
54              return "Blood Type: " + blood_type + Rh.getSign();
55          }
56      }
57  }
```

```
1
2  public enum RhFactor {
3      POSITIVE('+'), NEGATIVE('-');
4
5      private char sign;
6
7      private RhFactor(char signf) {
8          sign = signf;
9      }
10
11     public char getSign() {
12         return sign;
13     }
14
15 }
16
```

```
1
2  public enum BloodType {
3      O, A, B, AB;
4  }
5
```

```
Patient ID: 0
PatientAge: 0
Blood Type: O+
PatientID: 1337
Patient Age: 19
Blood Type: AB-
```