Appendix 1. Sample code for exercise 1

```
[1]  public class Beer {
[2]    String color = "gold";
[3]    static int number = 0;
[4]    public static int beersDrunk(){
[5]      number++;
[6]      System.out.println("Beers drunk "+number);
[7]      return number;
[8]    }
[9]    public void drinkMe(){
[10]     System.out.println("Drink me with anything");
[11]   }
[12]   public final void tasteMe() {
[13]           System.out.println("mmm!");
[14]   }
[15]   public static void main(String[] args) {
[16]     Beer amstel = new Beer();
[17]     Beer berliner = new Pilsner();
[18]     Beer pivo = new PilsnerUrquell();
[19]     Beer guinness = new Stout();
[20]     amstel.beersDrunk();
[21]     pivo.beersDrunk();
[22]     guinness.drinkMe();
[23]     guinnes.tasteMe();
[24]     Beer.drinkMe();
[25]     Beer.tasteMe();
[26]     Pilsner amstelBierPilsener;
[27]     amstelBierPilsener = (Pilsner)amstel;
[28]     amstelBierPilsener.beersDrunk();
[29]     amstelBierPilsener = new Pilsner();
[30]     amstelBierPilsener.beersDrunk();
[31]     amstelBierPilsener.color();
[32]     amstelBierPilsener.drinkMe();
[33]     amstel.drinkMe();
[34]     amstel.tasteMe();
[35]     amstel = (Beer)amstelBierPilsener;
[36]     amstel.color();
[37]     System.out.println("I am "+amstel.color);
[38]     amstel.beersDrunk();
[39]     pivo.drinkMe();
[40]     PilsnerUrquell ceske = (PilsnerUrquell)pivo;
[41]     ceske.color();
[42]     ceske.beersDrunk();
[43]     guinness = (Stout)ceske;
[44]     guinness.beersDrunk();
[45]     amstel = (Beer)guinness;
[46]     amstel.beersDrunk();
[47]     guinness.color();
[48]     Stout cooler = (Stout)guinness;
[49]     cooler.beersDrunk();
[50]     cooler.color();
[51]     System.out.println("I am "+cooler.color);
[52]     berliner.drinkMe();
[53]     System.out.println("I am "+berliner.color);
[54]     amstelBierPilsener  = (Pilsner)berliner;
```

```
      amstelBierPilsener.color();
      Stout.drinkMe();
   }
}

public class Pilsner extends Beer {
   static int number = 0;
   String color = "light ";
   public static int beersDrunk(){
      number++;
      System.out.println("pils' drunk "+number);
      return number;
   }
   public void drinkMe(){
      System.out.println("with meat or mild cheese");
   }
   public final void tasteMe() {
            System.out.println("So good!");
   }
   public void color(){
      System.out.println("I am "+color);
   }
}

public class PilsnerUrquell extends Pilsner {
   static int number = 0;
   public static int beersDrunk(){
      number++;
      System.out.println("Urquells drunk "+number);
      return number;
   }
   public void drinkMe(){
      System.out.println("or simply say: pivo");
   }
}

public class Stout extends Beer {
   static int number=0;
   String color = "deep black";
   public static int beersDrunk(){
      number++;
      System.out.println("Guinness' drunk "+number);
      return number;
   }
   public void drinkMe(){
      System.out.println("on St Patrick's Day");
   }
}
```

Appendix 2 Sample code for exercise 2

```
[1]  public class Guinness {
[2]    static int number = 0;
[3]    static int stock = 100;
[4]    int size=25;
[5]    public static int beersUsed(){
[6]      number++;
[7]      return number;
[8]    }
[9]    public void finish(){
[10]     int n=beersUsed();
[11]     stock -=n;
[12]   }
[13]   public static void main(String[] args) {
[14]     Guinness guinness = new Guinness();
[15]     guinness.finish();
[16]     Guinness.beersUsed();
[17]     Cooler cooler = new Cooler();
[18]     int measure = 2;
[19]     cooler.mix(measure);
[20]     int leftOver = cooler.size;
[21]     Guinness second;
[22]     second = (Guinness)cooler;
[23]     BlackVelvet bV= new BlackVelvet();
[24]     bV.mix(bV.size,measure);
[25]     leftOver+=bV.size;
[26]     second.finish();
[27]     {
[28]       Guinness cooler2 = new Cooler();
[29]       stock-=cooler2.beersUsed()/measure;
[30]       measure = 3;
[31]       cooler = (Cooler)cooler2;
[32]       leftOver += cooler.mix(measure);
[33]     }
[34]     second = (Guinness)bV;
[35]     bV.finish();
[36]     bV = new BlackVelvet();
[37]     bV.mix(leftOver,measure);
[38]     stock-=bV.beersUsed()/measure;
[39]   }
[40] }
[41] class BlackVelvet extends Guinness{
[42]   int jug;
[43]   public void finish(){
[44]    number++;
[45]   }
[46]   int mix(int glass, int measure){
[47]     int champagne = glass/measure;
[48]     size -= champagne;
[49]     jug = size+champagne;
[50]     return jug;
[51]   }
[52] }
```

```
[53] class Cooler extends Guinness{
[54]     int curacao = 1;
[55]     int cacao = 2;
[56]     static int number=0;
[57]     public static int beersUsed(){
[58]       number++;
[59]       return number;
[60]     }
[61]     int mix (int measure){
[62]       int ice = 5;
[63]       int jug = ice + curacao*measure;
[64]       jug+=cacao*measure;
[65]       jug = top(jug, this);
[66]       return size;
[67]     }
[68]     int top(int jug, Guinness g){
[69]       g.size-=jug;
[70]       jug+=g.size;
[71]       return jug;
[72]     }
[73] }
```