**package** q1;

**public** **interface** DisplayElement {

**public** **void** display(monkey m);

}

**package** q1;

**public** **class** Driver {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

monkey m= **new** monkey("mohammed");

monkey m2= **new** monkey("ahmad");

toffee d1=**new** toffee();

m2.registerObserver(d1);

m.registerObserver(d1);

Thread t1=**new** Thread (m);

Thread t2=**new** Thread (m2);

t1.start();

t2.start();

}

}

**package** q1;

**import** java.util.ArrayList;

**import** java.util.Random;

**public** **class** monkey **implements** Subject ,Runnable {

**private** ArrayList observers;

**float** temp;

String name;

**public** monkey(String name ){

**this**.name = name;

observers = **new** ArrayList();

}

**public** **void** setMeasurements(**float** temp)

{

**this**.temp=temp;

mChanged();

}

**public** **void** notifyObservers() {

// **TODO** A to u -generated method st b generated method stub

**for** (**int** i =0; i<observers.size(); i++)

{

Observer observer = (Observer)observers.get(i);

**synchronized**(**this**) {

observer.update(**this**);

}

}

}

**public** **void** registerObserver(Observer o) {

// **TODO** Auto-generated method stub

observers.add(o);

}

**public** **void** removeObserver(Observer o) {

// **TODO** Auto-generated method stub

**int** i = observers.indexOf(o);

**if** (i>=0) observers.remove(i);

}

**void** mChanged(){

notifyObservers();

}

@Override

**public** **synchronized** **void** run() {

**while** (**true**) {// **TODO** Auto-generated method stub

**int** r = **new** Random().nextInt(2);

setMeasurements(r);

**try** {

Thread.*sleep*(1000);

} **catch** (InterruptedException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

}

**package** q1;

**public** **interface** Observer {

**public** **void** update(monkey p);

}

**package** q1;

**public** **interface** Subject {

**public** **void** registerObserver(Observer o);

**public** **void** removeObserver(Observer o);

**public** **void** notifyObservers();

}

**package** q1;

**import** java.util.\*;

**public** **class** toffee **implements** Observer, DisplayElement {

**int** toffeecounter = 10;

**public** **void** update(monkey m) {

// **TODO** Auto-generated method stub

**if** (toffeecounter > 0){

**if** (m.temp == 1) {

display(m);

toffeecounter--;

}

**else** {

System.***out***.println( m.name +" grabbed toffee number"+ toffeecounter + " \n" + m.name +" returned toffee to box");

}

}

**else** {System.***out***.println("no more toffee");

}

}

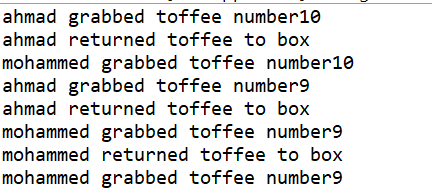
**public** **void** display(monkey m) {

// **TODO** Auto-generated method stub

System.***out***.println(m.name +" grabbed toffee number"+ toffeecounter);

}

}



Q2

**package** q2;

**public** **abstract** **class** bear **implements** Observer {

**public** **void** update(river m) {

// **TODO** Auto-generated method stub

**else** {

System.***out***.println( this.thread +" jumped to water");

}

}

**package** q2;

**public** **interface** Observer {

**public** **void** update(river b);

}

**package** q2;

**import** java.util.ArrayList;

**import** java.util.Random;

**import** q1.Observer;

**public** **abstract** **class** river **implements** Subject , Runnable {

**private** ArrayList observers;

**float** splash;

**public** river(){

observers = **new** ArrayList();

}

**public** **void** setSplash(**float** temp)

{

**this**.splash=temp;

mChanged();

}

**public** **void** notifyObservers() {

// **TODO** A to u -generated method st b generated method stub

**for** (**int** i =0; i<observers.size(); i++)

{

Observer observer = (Observer)observers.get(i);

**synchronized**(**this**) {

observer.update(**this**);

}

}

}

**public** **void** registerObserver(Observer o) {

// **TODO** Auto-generated method stub

observers.add(o);

}

**public** **void** removeObserver(Observer o) {

// **TODO** Auto-generated method stub

**int** i = observers.indexOf(o);

**if** (i>=0) observers.remove(i);

}

**void** mChanged(){

notifyObservers();

}

@Override

**public** **synchronized** **void** run() {

**while** (**true**) {// **TODO** Auto-generated method stub

**int** r = **new** Random().nextInt(2);

setSplash(r);

**try** {

Thread.*sleep*(1000);

} **catch** (InterruptedException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

}

**package** q2;

**public** **interface** Subject {

**public** **void** registerObserver(Observer o);

**public** **void** removeObserver(Observer o);

**public** **void** notifyObservers();

}