

```
1
2 /*
3  Creare una classe che vada a gestire un set di numeri
   interi appartenenti ad un range prefissato.
4
5  Ogni gestore di numeri interi istanziato conoscerà il
   valore minimo, il valore massimo (in riferimento al
   range da gestire) e se deve accettare numeri pari o
   numeri dispari.
6
7  La classe dovrà presentare i seguenti metodi:
8  Costruttori:
9  Default: istanzia un oggetto che accetta numeri pari
   compresi tra 0 e 200
10 Con parametri
11 Di copia
12 Getters e Setters
13 Un metodo per aggiungere un numero al set
14 Un metodo che restituisce una stringa csv di tutti i
   numeri del set
15 Un metodo che restituisce la media dei numeri del set
16 Un metodo che restituisce, in base ad un valore
   passato come parametro, il numero massimo o il numero
   minimo, tra i numeri del set
17
18 Crea un programma di prova che verifichi il
   funzionamento della classe.
19 */
20
21 import java.util.Scanner;
22
23 public class Main {
24     public static void main(String[] args) {
25         int m,num,val;
26         int vmin,vmax;
27         boolean pari,max;
28         String tipo,numUni;
29         double media;
30         vmin=leggiInt("inserisci valore minimo",
Integer.MIN_VALUE,Integer.MAX_VALUE);
31         vmax=leggiInt("inserisci valore massimo",vmin
```

```

31 ,Integer.MAX_VALUE);
32     do {
33         tipo = leggiStr("pari o dispari?");
34         }while(!tipo.equals("pari") && !tipo.equals("
dispari"));
35         if(tipo.equals("pari"))
36             pari=true;
37         else
38             pari=false;
39         Numeri set=new Numeri(vmin,vmax,pari);
40         do {
41             m = leggiInt("Inserisci:\n1.per inseire
un nuovo numero\n2.per visulizzare tutti i numeri del
set\n3.per visualizzare la media di tutti i numeri
del set\n4.per conoscere il valore max o min del set\
n5.termina\n", 1, 5);
42             switch (m) {
43                 case 1:
44                     do {
45                         num = leggiInt("inserisci
numero", vmin, vmax);
46                     }while((pari && num%2!=0) || (!
pari && num%2==0));
47                     set.aggiungiNum(num);
48                     break;
49                 case 2:
50                     numUni=set.uniNum();
51                     System.out.println(numUni);
52                     break;
53                 case 3:
54                     media=set.mediaN();
55                     System.out.println(media);
56                     break;
57                 case 4:
58                     do {
59                         tipo = leggiStr("valore
minimo o massimo?");
60                     }while(!tipo.equals("minimo"
) && !tipo.equals("massimo"));
61                     if(tipo.equals("massimo")){
62                         max=true;

```

```

63             }else {
64                 max = false;
65             }
66             try {
67                 val = set.maxOMin(max);
68                 if (max)
69                     System.out.println("il
numero max del set è " + val);
70                 else
71                     System.out.println("il
numero min del set è " + val);
72             }catch(IllegalStateException e){
73                 System.out.println(e.
getMessage());
74             }
75         }
76         }while(m!=5);
77     }
78     public static int leggiInt(String msg, int vmin
, int vmax) {
79         Scanner input = new Scanner(System.in);
80         int n;
81         System.out.println(msg);
82         do {
83             n = input.nextInt();
84             if (n < vmin || n > vmax)
85                 System.out.println("errore,
reinserire");
86         } while (n < vmin || n > vmax);
87         return n;
88     }
89
90     public static String leggiStr(String msg) {
91         Scanner input = new Scanner(System.in);
92         System.out.println(msg);
93         String s;
94         do{
95             s=input.nextLine();
96             if(s.isEmpty())
97                 System.out.println("errore,
reinserire");

```

```

98         }while(s.isEmpty());
99         return s;
100     }
101
102     public static double leggiDouble(String msg,
double vmin,double vmax) {
103         Scanner input = new Scanner(System.in);
104         System.out.println(msg);
105         double n;
106         do{
107             n=input.nextDouble();
108             if(n<vmin || n>vmax)
109                 System.out.println("errore,
reinserire");
110         }while(n<vmin || n>vmax);
111         return n;
112     }
113 }
```

```
1  /*
2  Creare una classe che vada a gestire un set di numeri
    interi appartenenti ad un range prefissato.
3  Ogni gestore di numeri interi istanziato conoscerà il
    valore minimo, il valore massimo (in riferimento al
    range da gestire) e se deve accettare numeri pari o
    numeri dispari.
4  La classe dovrà presentare i seguenti metodi:
5  Costruttori:
6  Default: istanzia un oggetto che accetta numeri pari
    compresi tra 0 e 200
7  Con parametri
8  Di copia
9  Getters e Setters
10 Un metodo per aggiungere un numero al set
11 Un metodo che restituisce una stringa csv di tutti i
    numeri del set
12 Un metodo che restituisce la media dei numeri del set
13
14 Un metodo che restituisce, in base ad un valore
    passato come parametro, il numero massimo o il numero
    minimo, tra i numeri del set
15 Crea un programma di prova che verifichi il
    funzionamento della classe.
16 */
17 /*
18 UML
19
20 classe Numeri
21
22 + classe ContoCorrente
23 attributi
24 - vmin: int
25 - vmax-int
26 - num- int[]
27 -pari- bool
28
29 costruttori
30 +Numeri()
31 +Numeri(int vmin,int vmax,boolean pari)
32 +Numeri(Numeri n)
```

```
33
34 metodi
35 + getVmin():int
36 + public getVmax():int
37 + getPari(): boolean
38 + setVmin: void
39 +setVmax:void
40 +setPari:void
41 +aggiungiNum(int):void
42 +uniNum():String
43 +mediaN():float
44 maxOMin(boolean):int
45
46 descr. metodi
47 int getVmin()
48 restituisce il valore di vmin di un oggetto
49
50 int getVmax()
51 restituisce il valore di vmax di un oggetto
52
53 boolean getPari()
54 restituisce il valore dell'attributo pari di un
   oggetto
55
56 void setVmin()
57 serve per modificare il valore di vmin di un oggetto
58
59 void setVmax()
60 serve per modificare il valore di vmax di un oggetto
61
62 void setPari()
63 serve per modificare il valore di pari di un oggetto
64
65 void aggiungiNum(int num)
66 se(num%2==0 && pari || num%2!=0 && !pari) && se(num>=
   vmin && num<=vmax) aggiungi numero al set e n++
67
68 String uniNum()
69 se(n>0) inserisci tutti i numeri del set in un'unica
   stringa altrimenti scrivi "nessun numero presente"
70
```

```

71 float mediaN()
72 se(n==) ritorna 0 altrimenti calcola la media dei
    numeri del set
73
74 int max0Min(boolean pari)
75 se(max) allora cerca il numero massimo del set e
    mettilo in val, altrimenti cerca il numero minimo
    del set e mettilo in val. Ritorna val.
76 */
77 public class Numeri {
78     private int vmin,vmax,n=0;
79     private int[] num;
80     private boolean pari;
81     public Numeri(){
82         this.vmin=0;
83         this.vmax=200;
84         this.pari=true;
85         this.num=new int[20];
86     }
87     public Numeri(int vmin,int vmax,boolean pari){
88         if(vmin>vmax) {
89             this.vmin = 0;
90             this.vmax = 200;
91             this.pari = true;
92         }else{
93             this.vmin=vmin;
94             this.vmax=vmax;
95             this.pari=pari;
96         }
97         this.num=new int[20];
98     }
99     public Numeri(Numeri n){
100         if (n!=null){
101             this.vmin = n.vmin;
102             this.vmax = n.vmax;
103             this.pari = n.pari;
104         }else {
105             this.vmin = 0;
106             this.vmax = 200;
107             this.pari = true;
108         }

```

```
109         this.n=0;
110         this.num=new int[20];
111     }
112     public int getVmin(){
113         return vmin;
114     }
115     public int getVmax(){
116         return vmax;
117     }
118     public boolean getPari(){
119         return pari;
120     }
121     public void setVmin(int vmin){
122         if(vmin<this.vmax)
123             this.vmin=vmin;
124         else
125             this.vmin=0;
126     }
127     public void setVmax(int vmax){
128         if(vmax>this.vmin)
129             this.vmax=vmax;
130         else
131             this.vmax=0;
132     }
133     public void setPari(boolean pari){
134         this.pari=pari;
135     }
136     public void aggiungiNum(int num){
137         if(num%2==0 && pari || num%2!=0 && !pari)
138             if(num>=vmin && num<=vmax){
139                 this.num[n]=num;
140                 n++;
141             }
142     }
143     public String uniNum(){
144         String numUni="";
145         if(n!=0)
146             for(int i=0;i<n;i++)
147                 numUni=numUni+num[i]+",";
148         else
149             numUni="nessun numero nel set";
```



```
150         return numUni;
151     }
152     public float mediaN(){
153         float media;
154         int tot=0;
155         if(n==0)
156             return 0;
157         for(int i=0;i<n;i++)
158             tot=tot+num[i];
159         media=tot/n;
160         return media;
161     }
162     public int maxOMin(boolean max){
163         int nMin=Integer.MAX_VALUE,nMax=Integer.
MIN_VALUE,val;
164         if(max) {
165             for (int i = 0; i < n; i++) {
166                 if (this.num[i] > nMax)
167                     nMax = this.num[i];
168             }
169             val=nMax;
170         }else {
171             for (int i = 0; i < n; i++) {
172                 if (this.num[i] < nMin)
173                     nMin = this.num[i];
174             }
175             val=nMin;
176         }
177         return val;
178     }
179 }
180
```