

```

1  /*Creacion y gestion vuelos*/
2  package newairlineseller;
3
4  import java.io.BufferedReader;
5  import java.io.IOException;
6  import java.io.InputStreamReader;
7
8  public class Vuelos {
9      String fechaVuelo, horaVuelo, ciudadOrigen, ciudadDestino, aux, color;
10     double tarifaBase;
11     boolean si, dispon;
12     int cantNaves = 10, disp = 96;
13     Manejador man = new Manejador();
14     BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
15
16     void crearVuelo() throws IOException{//Crear vuelo
17         man.crear("vuelo");//Crear archivo vuelo.txt
18         int num = man.buscar("vuelo")+1;//busca en el numero del ultimo vuelo creado y le
19         suma 1 para crear el siguiente vuelo.
20         System.out.println("\033[30mVuelo #\033[31m"+num+"\033[30m.");{//Imprime en pantalla
21         el numero de vuelo
22         do{//bucle para verificacion de datos ingresados para la cracion del vuelo
23             System.out.print("Fecha del Vuelo: ");
24             aux = br.readLine();
25             if(man.validarFecha(aux)){//evalua mediante una funcion si el dato fue ingrado
26             en el formato correcto.
27                 fechaVuelo = aux;
28                 si = true;
29                 //Si el dato es correcto se carga el valor de la variable auxiliar en la
30                 variable definitiva.
31                 //la variable "si" se evalua a verdadero.
32             }else{
33                 System.out.println("Fecha invalida.");
34                 si = false;
35                 //si el dato no tiene un formato valido, le indica al usuario que el dato no
36                 paso la verificacion
37                 //evalua la variable "si" a falso para que el bucle se repita y permita
38                 ingresarl el dato nuevamente.
39             }
40             }while(!si);//El bucle se reperita mientras la variable "si" se evalua falsa.
41         do{//bucle de virificacion del formato de la hora.
42             System.out.print("Hora de salida: ");
43             aux = br.readLine();
44             if(man.validarHora(aux)){
45                 horaVuelo = aux;
46                 si = true;
47             }else{
48                 System.out.println("Hora invalida.");
49                 si = false;
50             }
51         }while(!si);
52
53         System.out.println("Ciudad de origen: ");
54         ciudadOrigen = man.selecCiudad();//mediante la fucion selecCiudad carga un codigo de
55         ciudad en la variable correspondiente
56
57         System.out.println("Ciudad destino: ");

```

```

51     ciudadDestino = man.selecCiudad();
52
53     do{
54         System.out.print("Tarifa base: ");
55         aux = br.readLine();
56         if(man.validarTarifa(aux)){
57             tarifaBase = Double.parseDouble(aux);
58             si = true;
59         }else{
60             System.out.println("Fecha invalida.");
61             si = false;
62         }
63     }while(!si);
64     String data = num+"-"+fechaVuelo+"-"+horaVuelo+"-"+ciudadOrigen
65         +"-"+ciudadDestino+"-"+tarifaBase;
66     man.guardar("vuelo", data);
67 }
68
69 void verVuelos() throws IOException{//muestra los vuelos creados
70     System.out.print("Introduzca el numero del vuelos: ");
71     aux = br.readLine();
72     String[] datos = man.buscar("vuelo", aux);
73     //si la funcion buscar() devuelve un null significa que el archivo esta vacio
74     if(datos!=null){
75         System.out.println("\033[34m|-----|");
76         System.out.println("\033[34m|Vuelo #\033[31m"+datos[0]+" \033[34m\t\t\t\t\t|");
77         System.out.println("\033[34m|Con fecha: \033[31m"+datos[1]+" \033[34m - a las
78 \033[31m"+datos[2]+" \033[34m |");
79         System.out.println("\033[34m|De \033[31m"+datos[3]+" \033[34m a
80 \033[31m"+datos[4]+" \033[34m\t\t\t\t\t|");
81         System.out.println("\033[34m|Con una tatira base de
82 \033[31m"+datos[5]+" \033[34m\t\t\t\t\t|");
83         System.out.println("\033[34m|-----|");
84     }else{
85         System.out.println("Vuelo no encontrado...!");
86     }
87
88     boolean verificarVuelo(String n){//verifica la existencia de un vuelo en le archivo
89     vuelo.txt
90     String[] data = man.buscar("vuelo", n);
91     if(data!=null){
92         if(data[0].equals(n)){
93             return true;
94         }else{
95             return false;
96         }
97     }else{
98         return false;
99     }
100
101     String[] infoVuelo(String n){//envia un arreglo con la informacion de un buelo buscado
102     String[] data;
103     data = man.buscar("vuelo",n);
104     return data;
105 }

```

```

106
107     void disponibilidad(String numV) {
108         /*imprime en pantalla una matriz con la disposicion de los diferentes asientos
109         * en un vuelo. Se incican en color negro los asientos disponibles, y en
110         * color rojo los asientos ocupados*/
111         int i,j,x;
112
113         for(x=1; x<=16; x++){
114             if(x<=9) {
115                 System.out.print("\033[32m |"+x+"| ");
116             }else{
117                 System.out.print("\033[32m|"+x+"| ");
118             }
119         }
120
121         System.out.println();
122         System.out.println("\033[32m-----");
123         for (i=0;i<=5;i++) {
124             if(i==3) {
125                 System.out.println();
126             }
127             for (j=0;j<=15;j++) {
128                 //funcion de ver disponibilidad
129                 if(man.verDispon(j, i, numV)){
130                     dispon = false;
131                 }else{
132                     dispon = true;
133                 }
134                 if(dispon) {
135                     disp--;
136                     color = "\033[31m";
137                 }else{
138                     color = "\033[30m";
139                 }
140                 switch(i) {
141                     case 0 : System.out.print(color+" |A| "); break;
142                     case 1 : System.out.print(color+" |B| "); break;
143                     case 2 : System.out.print(color+" |C| "); break;
144                     case 3 : System.out.print(color+" |D| "); break;
145                     case 4 : System.out.print(color+" |E| "); break;
146                     case 5 : System.out.print(color+" |F| "); break;
147                 }
148             }
149             System.out.println();
150         }
151         System.out.println("\033[32m-----");
152         System.out.println("Lugares disponibles: \033[31m"+disp);
153         disp = 96;
154     }
155
156 }

```