

# Ejercicio práctico I

# Enunciado

➤ Queremos realizar un programa para la gestión de notas de una escuela. Cuando se inicie el programa se mostrará el siguiente menú:

1. Introducir nota.
2. Total aprobados.
3. Nota mayor.
4. Nota inferior.
5. Salir.

Elige opción:

➤ Al elegir la opción 1, se solicitará la introducción de una nota que será guardada por el programa (las notas estarán entre 0 y 10). Con la opción 2, nos mostrará el número de aprobados hasta el momento, mientras que con las opciones 3 y 4

# Enunciado

el programa nos mostrará la nota mayor y menor introducida hasta el momento, respectivamente.

➤Tras el procesamiento de una opción se volverá mostrar de nuevo el menú, así hasta que elija la opción de salir

# Ejercicio práctico II

# Enunciado

➤ Queremos realizar una agenda de contactos, donde cada contacto está definido por un nombre, email y telefono. Al iniciar la aplicación, se mostrará el siguiente menú:

1. Nuevo contacto.
2. Buscar contacto.
3. Eliminar contacto.
4. Mostrar contactos.
5. Salir.

Elige opción:

➤ Al elegir la opción 1, se solicitará la introducción de los datos de un nuevo contacto y este será añadido a la agenda.

# Enunciado

Con la opción 2, se realizará una búsqueda de contacto por email, mientras que con la opción 3 se elimina también a partir de su email . Al elegir la opción 4, se mostrarán los datos de todos los contactos.

➤Tras el procesamiento de una opción se volverá mostrar de nuevo el menú, así hasta que elija la opción de salir

# Nota de implementación

- La agenda de contactos será una lista de diccionarios, donde cada diccionario contendrá los datos de un contacto:

```
[{"nombre":"contacto1", "email":"contacto1@gmail.com","telefono":1234},  
 {"nombre":"contacto2", "email":"contacto2@gmail.com","telefono":5456},  
 {"nombre":"contacto3", "email":"contacto3@gmail.com","telefono":19224},  
 ....  
 ]
```

# Ejercicio práctico III



# Enunciado

➤ Realizar una nueva versión de la agenda de contactos que, en lugar de guardar los contactos en una lista los guarde en un fichero, separando los datos del contacto por una barra vertical, y escribiendo cada contacto en una línea

```
nombre|email|telefono  
nombre|email|telefono  
nombre|email|telefono
```

# Ejercicio práctico IV

# Enunciado

➤ Tenemos en un fichero un documento JSON que almacena datos diarios de contagios de una determinada enfermedad. El aspecto del fichero es el siguiente:

```
[{"pais": "España", "fecha": "2020-10-30", "contagios": 14300},  
{"pais": "Francia", "fecha": "2020-11-07", "contagios": 29400},  
{"pais": "EEUU", "fecha": "2020-10-30", "contagios": 65000},  
{"pais": "España", "fecha": "2020-11-04", "contagios": 21000},  
{"pais": "Gran Bretaña", "fecha": "2020-10-20", "contagios": 9830},  
{"pais": "Italia", "fecha": "2020-11-02", "contagios": 32000},  
:  
]
```

# Enunciado

➤ Se pide realizar un programa que nos muestre el siguiente menú:

1. Contagios de país
2. Contagios posteriores a fecha
3. País con más contagios
4. Salir

Introduce opción:

➤ Si se elige la opción 1, se solicitará la introducción del nombre de un país y el programa deberá mostrarnos el total de contagios registrados para ese país. Mediante la opción 2, se nos pide una fecha (años-mes-día) y se deberá mostrar el total de contagios producidos con posterioridad a esa fecha. Con la opción 3, se mostrará el nombre del país con mayor número de contagios registrados.

# Ejercicio práctico V

# Enunciado

- Realizar un programa que al ser ejecutado vuelque el contenido del fichero del programa anterior en una base de datos MongoDB.
  - Lo cargará en una colección que se regenerará cada vez que se ejecute el programa.
- 