

REQUISITIVOS IMPRESCINDIBLES. No cumplir con alguno de los siguientes requisitos implica la evaluación de la tarea correspondiente con una nota de 0.

- Todo el código ha de estar correctamente indentado
- Se han de seguir escrupulosamente la normas de nomenclatura (CamelCase o SNAKE_CASE,)
- Los nombres de paquetes deben estar basados en un nombre de dominio propio.
- Entregar un fichero comprimido del proyecto completo para cada ejercicio.
- Solo se admiten ficheros comprimidos con extensiones **.zip**, **.tgz** o **.rar** y con nombre del tipo:

nombreApellido1Apellido2xx

siendo **xx** el número de ejercicio con dos dígitos.

Si alguna de las tareas del AV no ha sido entregada antes del comienzo del examen y corregida con un APTO, la nota del examen será un 0.

Los ejercicios 1-6 puntúan sobre 1. Los ejercicios 7 y 8 puntúan sobre 2

1. Tenemos que almacenar un menú para 12 personas para cada uno de los días de la semana. El menú en si se puede codificar como un String, y sus datos introducidos hard-coded. Sin utilizar ninguna clase auxiliar ni crear ninguna nueva, almacenar toda la información en una sola variable. Comprobar la corrección pidiendo una persona (número del 0 al 11), un día de semana (número del 0 al 6) y mostrando el menú correspondiente.
2. Crear un vector de 100 enteros. Rellenar los elementos con índices del 37 al 59 con números aleatorios comprendidos entre 0 y el máximo números entero posible. Imprimir los valores generados en orden inverso de índices.
3. Programa que copie un fichero binario, recibiendo los parámetros (fcheroACopiar, ficheroNuevo) por la línea de comandos.
4. Programa que cree una BD llamada **examenXX** (siendo **XX** los dos últimos dígitos de vuestra IP) en el SXBD del servidor. Bajo esta BD, crear una tabla de nombre libre, con tres campos, uno de tipo texto y dos numéricos (a vuestra elección) e insertar 2 registros en la misma.

5. Recuperar los datos creados en el ejercicio 4) (si no lo habéis hecho, podéis crear la BD, tabla y registros del modo que prefiráis – Worbench, phpMyAdmin, ...) y volcarlos en un fichero binario de nombre **misDatos** en el directorio **/home/alumno/Descargas**
6. Crear un proyecto para el código que se muestra a continuación e internacionalizarlo (Español, Inglés y Francés)

```
import java.sql.*;

public class CreandoBD {

    public static void main(String[] args) {
        Connection con = null;
        String user = "root";
        String passwd = "root";
        String url = "jdbc:mysql://10.0.3.100/";
        Statement stm = null;
        String bd = "mysql";
        try {
            con = DriverManager.getConnection(url, user, passwd);
            System.out.println("Conexión establecida");
            stm = con.createStatement();
            stm.executeUpdate("use "+ bd);
            System.out.println("Base de Datos "+bd+" seleccionada");
            stm.close();
            con.close();
        } catch (SQLException ex) {
            System.out.println("Error en la conexión con el SXBD");
        }
    }
}
```

7. Aplicación JavaFX con tres ficheros:

- App.java
- Control.java
- ventana.fxml

que debe permitir poner nota (1-10 en función de la dificultad) a cada uno de los módulos de 1ºDAW. El módulo se escoge mediante un ChoiceBox, la nota se debe introducir en un TextField, y habrá un botón de validación. Cada vez que se pulsa el botón ha de comprobarse que la nota no está vacía; de ser así se actualizará una colección en la que llevaremos, para cada módulo, el número de votaciones y la media. Habrá otro botón que nos permitirá volcar a un fichero de texto los resultados.

8. Realizar mediante **Maps** y **Sets**. Todos los datos han de ser introducidos por teclado. Introducir los nombres (abreviaturas) de los todos los módulos de los ciclos que se imparten en el instituto, clasificándolos correctamente por ciclo (no importa el curso). Sacar un listado ordenado por orden alfabético de los módulos que se imparten en 2 ciclos distintos (SMR/ASIR o DAW/ASIR) pero no en los 3 y otro de los que se imparten en los 3 ciclos. El listado será de la forma: abreviatura – descripción. Serializar las colecciones de modo que la próxima vez que se ejecute el programa no sea necesario volver a introducir los datos

Los módulos son los siguientes (las descripciones las podéis consultar en el AV o directamente inventáros las):

SMR	DAW	ASIR
SOM	SINF	LM
MMEI	LM	IMSO
AO	FOL	BD
RL	BD	PAR
FOL	CD	FH
AW	P	FOL
SER	DWCS	SRI
SI	DAW	SAD
SOR	DWCC	ASO
EIE	DIW	EIE
IAW	EIE	ASXBD
		IAW

Nota: IAW no se da en SMR, está puesto solo para forzar el resultado esperado