

## PRÁCTICA 7\_02.01

Realiza los siguientes scripts en Python:

1. Escribir un programa que almacene la cadena de caracteres **contraseña** en una variable, pregunte al usuario por la contraseña e imprima por pantalla si la contraseña introducida por el usuario coincide con la guardada en la variable sin tener en cuenta mayúsculas y minúsculas. (*if -else*)
2. Escribir un programa que pida al usuario dos números y devuelva su división. Si el usuario no introduce números debe devolver un aviso de error y si el divisor es cero también. (*if -else*)
3. Escribir un programa que pida al usuario un número entero y muestre por pantalla si es par o impar. (*if -else*)
4. Los alumnos de un curso se han dividido en dos grupos A y B de acuerdo al sexo y el nombre. El grupo A esta formado por las mujeres con un nombre anterior a la M y los hombres con un nombre posterior a la N y el grupo B por el resto. Escribir un programa que pregunte al usuario su nombre y sexo, y muestre por pantalla el grupo que le corresponde. (*if -else*)
5. Los tramos impositivos para la declaración de la renta en un determinado país son los siguientes:

Renta	Tipo impositivo
Menos de 10000€	5%
Entre 10000€ y 20000€	15%
Entre 20000€ y 35000€	20%
Entre 35000€ y 60000€	30%
Más de 60000€	45%

Escribir un programa que pregunte al usuario su renta anual y muestre por pantalla lo que tiene que pagar. (*if -else*)

6. Escribir un programa que almacene la cadena de caracteres **contraseña** en una variable, pregunte al usuario por la contraseña hasta que introduzca la contraseña correcta. (*while*)
7. Escribir un programa en el que se pregunte al usuario por una frase y una letra, y muestre por pantalla el número de veces que aparece la letra en la frase. (*for - if*)

8. Escribir un programa que almacene las asignaturas de un curso (por ejemplo Matemáticas, Física, Química, Historia y Lengua) en una **lista**, pregunte al usuario la nota que ha sacado en cada asignatura, y después las muestre por pantalla con el mensaje **En <asignatura> has sacado <nota>** donde **<asignatura>** es cada una de las asignaturas de la lista y **<nota>** cada una de las correspondientes notas introducidas por el usuario. (*for*)
9. Escribir un programa que almacene las asignaturas de un curso (por ejemplo Matemáticas, Física, Química, Historia y Lengua) en una **lista**, pregunte al usuario la nota que ha sacado en cada asignatura y elimine de la lista las asignaturas aprobadas. Al final el programa debe mostrar por pantalla las asignaturas que el usuario tiene que repetir. (*for-if*)
10. Escribir un programa que guarde en un **diccionario** los precios de las frutas de la tabla, pregunte al usuario por una fruta, un número de kilos y muestre por pantalla el precio de ese número de kilos de fruta. Si la fruta no está en el diccionario debe mostrar un mensaje informando de ello. (*if -else*)

Fruta	Precio
Plátano	1.35
Manzana	0.80
Pera	0.85
Naranja	0.70

11. Escribir un programa que almacene el **diccionario** con los créditos de las asignaturas de un curso {'Matemáticas': 6, 'Física': 4, 'Química': 5} y después muestre por pantalla los créditos de cada asignatura en el formato **<asignatura> tiene <créditos> créditos**, donde **<asignatura>** es cada una de las asignaturas del curso, y **<créditos>** son sus créditos. Al final debe mostrar también el número total de créditos del curso. (*for*)
12. Escribir un programa que cree un **diccionario** vacío y lo vaya llenando con información sobre una persona (por ejemplo nombre, edad, sexo, teléfono, correo electrónico, etc.) que se le pida al usuario. Cada vez que se añada un nuevo dato debe imprimirse el contenido del diccionario. (*while*)

13. Escribir un programa que cree un **diccionario** simulando una cesta de la compra. El programa debe preguntar el artículo y su precio y añadir el par al diccionario, hasta que el usuario decida terminar. Después se debe mostrar por pantalla la lista de la compra y el coste total, con el siguiente formato: (*while - for*)

**Lista de la compra**

Artículo 1	Precio
Artículo 2	Precio
Artículo 3	Precio
...	...
Total	Coste

14. Escribir un programa que permita gestionar la base de datos de clientes de una empresa. Los clientes se guardarán en un diccionario en el que la clave de cada cliente será su NIF, y el valor será otro diccionario con los datos del cliente (nombre, dirección, teléfono, correo, preferente), donde preferente tendrá el valor True si se trata de un cliente preferente. El programa debe preguntar al usuario por una opción del siguiente menú: (1) Añadir cliente, (2) Eliminar cliente, (3) Mostrar cliente, (4) Listar todos los clientes, (5) Listar clientes preferentes, (6) Terminar. En función de la opción elegida el programa tendrá que hacer lo siguiente:

- (1) Preguntar los datos del cliente, crear un diccionario con los datos y añadirlo a la base de datos.
- (2) Preguntar por el NIF del cliente y eliminar sus datos de la base de datos.
- (3) Preguntar por el NIF del cliente y mostrar sus datos.
- (4) Mostrar lista de todos los clientes de la base datos con su NIF y nombre.
- (5) Mostrar la lista de clientes preferentes de la base de datos con su NIF y nombre.
- (6) Terminar el programa.

15. El directorio de los clientes de una empresa está organizado en una cadena de texto como la de más abajo, donde cada línea contiene la información del nombre, email, teléfono, nif, y el descuento que se le aplica. Las líneas se separan con el carácter de cambio de línea \n y la primera línea contiene los nombres de los campos con la información contenida en el directorio.

```
"nif;nombre;email;teléfono;descuento\n01234567L;Luis  
González;luisgonzalez@mail.com;656343576;12.5\  
n71476342J;Macarena Ramírez;macarena@mail.com;692839321;8\  
n63823376M;Juan José Martínez;juanjo@mail.com;664888233;5.2\  
n98376547F;Carmen Sánchez;carmen@mail.com;667677855;15.7"
```

Escribir un programa que genere un diccionario con la información del directorio, donde cada elemento corresponda a un cliente y tenga por clave su nif y por valor otro diccionario con el resto de la información del cliente. Los diccionarios con la información de cada cliente tendrán como claves los nombres de los campos y como valores la información de cada cliente correspondientes a los campos. Es decir, un diccionario como el siguiente

```
{'01234567L': {'nombre': 'Luis González', 'email':  
luisgonzalez@mail.com, 'teléfono': '656343576', 'descuento':  
12.5}, '71476342J': {'nombre': 'Macarena Ramírez', 'email':  
macarena@mail.com, 'teléfono': '692839321', 'descuento': 8.0},  
'63823376M': {'nombre': 'Juan José Martínez', 'email':  
juanjo@mail.com, 'teléfono': '664888233', 'descuento': 5.2},  
'98376547F': {'nombre': 'Carmen Sánchez', 'email':  
carmen@mail.com, 'teléfono': '667677855', 'descuento': 15.7}}
```

**Entrega:** apellido1\_apellido2\_nombre.zip (con ejercicio1.py, ejercicio2.py...)