

01. Fundamentos de Python - Ejercicios

1. Variables y Tipos de Datos

1. Escribe un archivo `variables.py` que defina tres variables: una entera, una de punto flotante y una cadena de texto. Luego, imprime sus valores y tipos usando `type()`.
 2. Escribe un archivo `operaciones_basicas.py` que declare dos variables numéricas y realice las operaciones básicas (+, -, *, /, %, //, **). Imprime los resultados usando `f-strings`.
 3. Escribe un archivo `conversion_temperatura.py` que convierta grados Celsius a Fahrenheit y viceversa.
-

2. Estructuras de Control (Condicionales y Bucles)

4. Escribe un archivo `numero_positivo.py` que solicite al usuario un número e indique si es positivo, negativo o cero.
 5. Escribe un archivo `bucles.py` que imprima los números del 1 al 10 usando un bucle `for`, y luego haga lo mismo con un bucle `while`.
 6. Escribe un archivo `fibonacci.py` que solicite un número al usuario y genere la secuencia de Fibonacci hasta ese número.
 7. Escribe un archivo `validar_contraseña.py` que pida al usuario una contraseña y verifique que cumpla los siguientes criterios:
 - Al menos 8 caracteres
 - Contiene mayúsculas y minúsculas
 - Contiene al menos un número
-

3. Funciones

8. Escribe un archivo `suma_funcion.py` que defina una función `sumar(a, b)` que reciba dos números y retorne su suma.

9. Escribe un archivo `numero_primo.py` que contenga una función `es_primo(n)` para determinar si un número es primo.
 10. Escribe un archivo `filtrar_pares.py` que contenga una función `filtrar_pares(lista)`, la cual recibe una lista de números y devuelve otra lista con solo los números pares.
 11. Escribe un archivo `factorial.py` que defina una función que reciba un número `n` y retorne su factorial usando recursión.
-

4. Manejo de Cadenas (Strings) y F-Strings

12. Escribe un archivo `invertir_palabras.py` que pida al usuario una frase y devuelva la misma frase con cada palabra en orden inverso.
 13. Escribe un archivo `contar_vocales.py` que cuente cuántas veces aparece cada vocal en una cadena ingresada por el usuario.
 14. Escribe un archivo `iniciales.py` que solicite al usuario su nombre y apellidos y devuelva sus iniciales en mayúsculas.
 15. Escribe un archivo `fstrings_basico.py` que reciba el nombre y edad de una persona e imprima:
"Hola, mi nombre es {nombre} y tengo {edad} años." usando `f-strings`.
 16. Escribe un archivo `fstrings_operaciones.py` que reciba un número ingresado por el usuario e imprima su cuadrado y su raíz cuadrada con formato de dos decimales usando `f-strings`.
 17. Escribe un archivo `fstrings_tabla.py` que imprima una tabla con números del 1 al 10 y su respectivo cuadrado, alineando correctamente las columnas con `f-strings`.
-

5. Listas y Tuplas

18. Escribe un archivo `lista_cuadrados.py` que cree una lista con los números del 1 al 5 y use un bucle para imprimir cada elemento elevado al cuadrado.
 19. Escribe un archivo `buscar_frutas.py` que cree una tupla con nombres de frutas y permita al usuario ingresar un nombre para verificar si está en la tupla.
-

6. Diccionarios

20. Escribe un archivo `capitales.py` que cree un diccionario con países como clave y sus capitales como valor. Luego, permite al usuario ingresar un país y muestra su capital.
 21. Escribe un archivo `frecuencia_palabras.py` que cuente la frecuencia de palabras en un texto ingresado por el usuario.
-

7. Módulos y Manejo de Archivos

22. Escribe un módulo `calculos.py` que contenga una función para calcular el área de un círculo. Luego, crea un archivo `uso_calculos.py` que lo importe y lo utilice.
 23. Escribe un archivo `leer_archivo.py` que lea un archivo llamado `datos.txt` y cuente cuántas líneas tiene.
 24. Escribe un archivo `escribir_json.py` que guarde un diccionario en un archivo JSON llamado `datos.json`. Luego, crea otro archivo `leer_json.py` que lo lea e imprima su contenido.
-

8. Programación Orientada a Objetos (POO)

25. Escribe un archivo `clase_persona.py` que defina una clase `Persona` con atributos `nombre` y `edad`, y un método que imprima un saludo con su nombre.
 26. Escribe un archivo `clase_coche.py` que defina una clase `Coche` con atributos `marca` y `modelo`, y un método que imprima su descripción.
-

9. Expresiones Regulares

27. Escribe un archivo `validar_email.py` que valide si un correo electrónico ingresado por el usuario tiene un formato correcto usando **expresiones regulares**.
-

10. Uso de Librerías Estándar

28. Escribe un archivo `random_numeros.py` que genere 10 números aleatorios entre 1 y 100 usando la librería `random`.
29. Escribe un archivo `fecha_actual.py` que muestre la fecha y hora actual en formato `dd-mm-yyyy HH:MM:SS` usando `datetime`.

30. Escribe un archivo `math_funciones.py` que use la librería `math` para calcular la raíz cuadrada y el logaritmo en base 10 de un número ingresado por el usuario.