



**INSTITUTO TÉCNICO
DE ESTUDIOS
PROFESIONALES**

BOLETÍN DE EJERCICIOS DE PYTHON

Miguel Ángel

Rodríguez Dalí

BOLETÍN DE EJERCICIOS

1. Escribir un programa que muestre por pantalla ¡Hola Mundo!

```
print('World')
```

Tipos de datos

1. Desarrollar un programa que muestre por pantalla la cadena ¡Hola Mundo!

```
print("Hola Mundo")
```

2. Desarrollar un programa que almacene la cadena ¡Hola Mundo! en una variable y luego muestre por pantalla el contenido de la variable.

```
hola = "Hola Mundo"  
print(hola)
```

3. Desarrollar un programa que pregunte el nombre del usuario en la consola y después de que el usuario lo introduzca muestre por pantalla la cadena ¡Hola <nombre>!, donde <nombre> es el nombre que el usuario haya introducido.

```
nombreUsuario = input("Introduzca su nombre: ")  
print(f"Hola {nombreUsuario}")
```

4. Desarrollar un programa que muestre por pantalla el resultado de la siguiente operación aritmética $(\frac{3+2}{2*5})^{**2}$

```
operacionAritmetica = ((3+2)/(2*5))**2  
print(operacionAritmetica)
```

5. Desarrollar un programa que pregunte al usuario por el número de horas trabajadas y el coste por hora. Después debe mostrar por pantalla la paga que le corresponde

```
horasTrabajadas = int(input("¿Cuántas horas trabajadas tiene?: "))  
costeHora = float(input("¿Y a cuánto cobra cada hora?: "))  
pagaCorrespondiente = horasTrabajadas*costeHora  
print(f"Le corresponderían {pagaCorrespondiente}€")
```

6. Desarrollar un programa que lea un entero positivo, n , introducido por el usuario y después muestre en pantalla la suma de todos los enteros desde 1 hasta n . La suma de los primeros enteros positivos puede ser calculada de la siguiente forma:

$$suma = \frac{n(n+1)}{2}$$

```
while (True):
    entero = int(input("Introduzca un número entero positivo: "))
    if (entero > 0):
        break
suma = int(entero*(entero+1)/2)
print(suma)
```

7. Desarrollar un programa que pida al usuario su peso (en kg) y estatura (en metros), calcule el índice de masa corporal y lo almacene en una variable, y muestre por pantalla la frase **Tu índice de masa corporal es <imc>** donde **<imc>** es el índice de masa corporal calculado redondeado con dos decimales.

```
peso = float(input("Introduzca su peso (kg): "))
estatura = float(input("Introduzca su altura (m): "))
imc = round(peso/(estatura**2),2)
print(f"Tu índice de masa corporal es: {imc}")
```

8. Desarrollar un programa que pida al usuario dos números enteros y muestre por pantalla la **<n> entre <m> da un cociente <c> y un resto <r>** donde **<n>** y **<m>** son los números introducidos por el usuario, y **<c>** y **<r>** son el cociente y el resto de la división entera respectivamente

```
entero1 = int(input("Introduzca el primer entero: "))
entero2 = int(input("Introduzca el segundo entero: "))

cociente = entero1//entero2
resto = entero1%entero2
print(f"{entero1} entre {entero2} da un cociente {cociente} y un resto {resto}")
```

9. Escribir un programa que pregunte al usuario una cantidad a invertir, el interés anual y el número de años, y muestre por pantalla el capital obtenido en la inversión

```
cantInvertir = float(input("¿Cuánta cantidad de dinero desea invertir?: "))
interesAnual = float(input("¿Cuánto interés anual tiene?: "))
cantAños = int(input("¿Durante cuántos años?: "))

capitalObtenido = cantInvertir*interesAnual*cantAños
print(f"El capital que ha obtenido en la inversión es: {capitalObtenido}")
```

10. Una juguetería tiene mucho éxito en dos de sus productos: payasos y muñecas. Suele hacer venta por correo y la empresa de logística les cobra por peso de cada paquete así que deben calcular el peso de los payasos y muñecas que saldrán en cada paquete a demanda. Cada payaso pesa 112 g y cada muñeca 75 g. Escribir un programa que lea el número de payasos y muñecas vendidos en el último pedido y calcule el peso total del paquete que será enviado

```
cantPayasos = int(input("¿Cuántos payasos se han vendido?: "))
cantMuñecas = int(input("¿Cuántas muñecas se han vendido?: "))

payaso = 112
muñeca = 75
totalPaquete = cantPayasos*payaso + cantMuñecas*muñeca
print(f"El peso total del paquete que será enviado es de: {totalPaquete}")
```

11. Imagina que acabas de abrir una nueva cuenta de ahorros que te ofrece el 4% de interés al año. Estos ahorros debido a intereses, que no se cobran hasta finales de año, se te añaden al balance final de tu cuenta de ahorros. Escribir un programa que comience leyendo la cantidad de dinero depositada en la cuenta de ahorros, introducida por el usuario. Después el programa debe calcular y mostrar por pantalla la cantidad de ahorros tras el primer, segundo y tercer años. Redondear cada cantidad a dos decimales

```
dineroDepositado = float(input("¿Cuánto dinero ha depositado en la cuenta de ahorros?: "))

INTERESANUAL = 1.04 #4%+
primerAnio = round(dineroDepositado*INTERESANUAL, 2)
segundoAnio = round(dineroDepositado*(INTERESANUAL*2), 2)
tercerAnio = round(dineroDepositado*(INTERESANUAL*3), 2)

print("Durante los tres primeros años serían:")
print(f"1er año: {primerAnio}€; 2do año: {segundoAnio}€; 3er año: {tercerAnio}€")
```

12. Una panadería vende barras de pan a 3.49€ cada una. El pan que no es el día tiene un descuento del 60%. Escribir un programa que comience leyendo el número de barras vendidas que no son del día. Después el programa debe mostrar el precio habitual de una barra de pan, el descuento que se le hace por no ser fresca y el coste final total

```
BARRA_PAN = 3.49
PAN_NO_DIA = 0.60 #60%

barrasNoDelDia = int(input("¿Cuántas barras que no son del día ha vendido?: "))
total = barrasNoDelDia * PAN_NO_DIA

print(f"El precio habitual de la barra de pan es {BARRA_PAN}€, por no ser fresca se hace un descuento del 60%")
print(f"En total saldría por {total}€")
```

Cadenas

13. Calcula con una única expresión el valor absoluto del redondeo de -3.2 . (El resultado es 3.0 .)

```
valAbsoluto = abs(round(-3.2,0))  
print(valAbsoluto)
```

14. Convierte (en una única expresión) a una cadena el resultado de la división $5011/10000$ redondeado con 3 decimales.

```
resultado = round(5011/10000,3)  
print(resultado)
```

15. Diseña un programa que, a partir del valor de la base y de la altura de un triángulo (3 y 5 metros, respectivamente), muestre el valor de su área (en metros cuadrados). Recuerda que el área A de un triángulo se puede calcular a partir de la base b y la altura h como

$$A = \frac{1}{2} b \cdot h$$

```
base = 3  
altura = 5  
area = 0.5*base*altura  
  
print(area)
```

16. Diseña un programa Python que lea un carácter cualquiera desde el teclado, y muestre el mensaje “Es una MAYÚSCULA” cuando el carácter sea una letra mayúscula y el mensaje. “Es una MINÚSCULA” cuando sea una minúscula. En cualquier otro caso, no mostrará mensaje alguno. (Considera únicamente letras del alfabeto inglés.)

Pista: aunque parezca una obviedad, recuerda que una letra es minúscula si está entre la 'a' y la 'z', y mayúscula si está entre la 'A' y la 'Z'.

```
caracter = input("Introduzca carácter: ")  
  
if (caracter.isupper()):  
    print("Es una MAYÚSCULA")  
elif (caracter.islower()):  
    print("Es una MINÚSCULA")  
else:  
    print("No es un carácter alfabético")
```

17. Amplía la solución al ejercicio anterior para que cuando el carácter introducido no sea una letra muestre el mensaje "No es una letra". (Nota: no te preocupes por las letras ñe, cedilla, vocales acentuadas, etc.)

```
caracter = input("Introduzca carácter: ")

if (len(caracter) > 1):
    print("No es una letra")
elif (caracter.isupper()):
    print("Es una MAYÚSCULA")
elif (caracter.islower()):
    print("Es una MINÚSCULA")
else:
    print("No es un carácter alfabético")
```

18. Haz un programa que pida el nombre de una persona y lo muestre en pantalla repetido 10 veces, pero dejando un espacio de separación entre aparición y aparición del nombre. (Utiliza los operadores de concatenación y repetición.)

```
nombre = input("Introduzca su nombre: ")

nombre = (nombre + " ")*10

print(nombre)
```

19. Diseña un programa que solicite el radio de una circunferencia y muestre su área y perímetro con sólo 2 decimales.

```
import math

radioCircunferencia = float(input("Introduzca el radio: "))
area = round(math.pi * radioCircunferencia**2, 2)
perimetro = round(math.pi*2*radioCircunferencia, 2)

print(f"El área es {area}")
print(f"El perímetro es {perimetro}")
```

20. Diseña un programa que lea un número flotante por teclado y muestre por pantalla el mensaje "El número es negativo" sólo si el número es menor que cero.

```
numeroFlotante = float(input("Introduzca número: "))

if (numeroFlotante < 0):
    print("El número es negativo")
```

21. Diseña un programa que lea un número flotante por teclado y muestre por pantalla el mensaje *"El número es positivo"* sólo si el número es mayor o igual que cero.

```
numeroFlotante = float(input("Introduzca número: "))

if (numeroFlotante >= 0):
    print("El número es positivo")
```

22. Escribir un programa que pida al usuario que introduzca una frase en la consola y muestre por pantalla la frase invertida

```
frase = input("Introduzca una frase: ")

print(frase[::-1])
```

Condicionales

23. Un usuario quiere aparcar en la calle pero solo puede aparcar los días del 1-15 en la acera A y del 16-31 en la acera B. Introducir por pantalla y imprimir según el día que meta en qué acera puede apartar para que el usuario no se confunda de lado de acera

```
dia = int(input("Introduzca el día del mes: "))

if (dia >= 1 and dia <= 15):
    print("Puede aparcar en la acera A")
elif (dia >= 16 and dia <= 31):
    print("Puede aparcar en la acera B")
```

24. Imprimir según el mes del año en que estación se encuentra. Si no pertenece a ninguna estación la devolución es "No existe el mes". Si existe el mes en la estación por ejemplo otoño deberá devolver por ejemplo: "estas en otoño"

```
mes = int(input("Introduzca el mes: "))

if mes >= 1 and mes < 3 or mes == 12:
    print("Estás en Invierno")
elif mes >= 3 and mes < 6:
    print("Estás en Primavera")
elif mes >= 6 and mes < 9:
    print("Estás en Verano")
elif mes >= 9 and mes < 12:
    print("Estás en Otoño")
else:
    print("No existe el mes")
```

25. Diseña un programa que, dado un número real que debe representar la calificación numérica de un examen, proporcione la calificación cualitativa correspondiente al número dado. La calificación cualitativa será una de las siguientes: "Suspenso" (nota menor que 5), "Aprobado" (nota mayor o igual que 5, pero menor que 7), "Notable" (nota mayor o igual que 7, pero menor que 8.5), "Sobresaliente" (nota mayor o igual que 8.5, pero menor que 10), "Matrícula de Honor" (nota 10).

```
calificacionExamen = float(input("Introduzca la calificación del examen: "))

if calificacionExamen < 5:
    print("Suspenso")
elif calificacionExamen < 7:
    print("Bien")
elif calificacionExamen < 8.5:
    print("Notable")
elif calificacionExamen < 10:
    print("Sobresaliente")
else:
    print("Matrícula de honor")
```

26. Diseña un programa que, dado un carácter cualquiera, lo identifique como vocal minúscula, vocal mayúscula, consonante minúscula, consonante mayúscula u otro tipo de carácter.

```
caracter = input("Introduzca su carácter: ")
vocales = ["a", "e", "i", "o", "u"]

if caracter in vocales:
    salida = "Vocal minúscula"
elif caracter in [v.upper() for v in vocales]:
    salida = "Vocal mayúscula"
elif caracter >= "b" and caracter <= "z":
    salida = "Consonante minúscula"
elif caracter >= "B" and caracter <= "Z":
    salida = "Consonante mayúscula"
else:
    salida = "Otro tipo de caracter"

print(salida)
```


27. Escribe un programa que según la opción que se le pase calcule el área o perímetro de una figura. Por ejemplo:

Escoge una opción:

- a) Calcular área triángulo
- b) Calcular perímetro triángulo
- c) Calcular área círculo
- d) Calcular perímetro círculo

```
import math

class Menu:
    print("a) Calcular el área del triángulo")
    print("b) Calcular perímetro triángulo")
    print("c) Calcular área círculo")
    print("d) Calcular perímetro círculo")

Menu()
opc = input("Escoja opción: ")

if opc == "a":
    base = float(input("Introduzca la base: "))
    altura = float(input("Introduzca la altura: "))

    area = base * altura / 2

    print(f"El área del triángulo es: {area}")
elif opc == "b":
    a = float(input("Introduzca la longitud del lado 1: "))
    b = float(input("Introduzca la longitud del lado 2: "))
    c = float(input("Introduzca la longitud del lado 3: "))

    perimetro = a + b + c

    print(f"El perímetro del triángulo es: {perimetro}")
elif opc == "c":
    radio = float(input("Introduzca el radio del círculo: "))

    area = math.pi * radio ** 2

    print(f"El área del círculo es: {area}")
elif opc == "d":
    diametro = float(input("Introduzca el diámetro del círculo: "))

    perimetro = math.pi * diametro

    print(f"El perímetro del círculo es: {perimetro}")
else:
    print("No ha introducido un valor correcto")
```

28. Desarrolla un programa que almacene una contraseña introducida previamente y la compruebe con una nueva que se le introduzca por segunda vez. Si las contraseñas coinciden debe devolver "la contraseña es correcta". Además agregarle que si la contraseña no es alfanumérica debe de volver que el nivel de la contraseña es débil. Si es alfanumérica debe de volver que su contraseña "es correcta y fuerte"

```
contrasenia = input("Introduzca una contraseña: ")
comprobante = input("Introduzca la contraseña de nuevo: ")

if contrasenia == comprobante:
    print("La contraseña es correcta")
    if contrasenia.isalnum() and not contrasenia.isdigit():
        print("Su contraseña es correcta y fuerte")
    else:
        print("La contraseña es débil")
```

29. Desarrolla un programa que calcule la división de dos números y que le pregunte al usuario si quiere el resto o el cociente.

```
dividendo = float(input("Introduzca el dividendo: "))
divisor = float(input("Introduzca el divisor: "))

resultado = input("¿Desea resto o cociente? (resto|cociente): ")

if resultado.lower() == "resto":
    salida = dividendo % divisor # resto
elif resultado.lower() == "cociente":
    salida = dividendo / divisor # cociente
else:
    print("No ha introducido un valor correcto")
    salida = ""

print(salida)
```

30. Desarrolla un programa que introduzca la fecha completa DD/MM/YYYY. Si la fecha no esta dentro del rango debe de devolver que la fecha no esta en el formato solicitado DD/MM/YYYY. Ejemplo de devolución:

53/12/2021 la devolución será: la fecha no es correcta porque el día 53 no esta dentro del rango de días de una fecha estipulada

12/53/2021 la devolución sería: la fecha no es correcta porque el mes 53 no esta dentro del rango de meses de una fecha estipulada

```
fecha = input("Introduzca la fecha deseada DD/MM/YYYY: ")
vFecha = fecha.split("/")

if int(vFecha[0]) < 1 or int(vFecha[0]) > 31:
    print(f"La fecha no es correcta porque el día {vFecha[0]} no está dentro del rango de días de una fecha estipulada")
elif int(vFecha[1]) < 1 or int(vFecha[1]) > 12:
    print(
        f"La fecha no es correcta porque el mes {vFecha[1]} no está dentro del rango de meses de una fecha estipulada")
```

Bucles

31. Diseña un programa que solicite la lectura de un número entre 0 y 10 (ambos inclusive). Si el usuario teclea un número fuera del rango válido, el programa solicitará nuevamente la introducción del valor cuantas veces sea menester.

```
numero = int(input("Introduzca un número entre 0 y 10: "))

while (numero not in range(0,10+1)):
    numero = int(input("Introduzca un número entre 0 y 10: "))
```

32. Diseña un programa que solicite la lectura de un texto que no contenga letras mayúsculas. Si el usuario teclea una letra mayúscula, el programa solicitará nuevamente la introducción del texto cuantas veces sea preciso.

```
texto = input("Introduzca un texto sin letras mayúsculas: ")

while (texto != texto.lower()):
    texto = input("Introduzca un texto sin letras mayúsculas: ")
```

33. Un vector en un espacio tridimensional es una tripleta de valores reales (x, y, z). Deseamos confeccionar un programa que permita operar con dos vectores. El usuario verá en pantalla un menú con las siguientes opciones:

- * Introducir el primer vector
- * Introducir el segundo vector
- * Calcular la suma
- * Calcular la diferencia
- * Calcular el producto escalar
- * Calcular el producto vectorial
- * Calcular el ángulo (en grados) entre ellos
- * Calcular la longitud
- * Finalizar

Puede que necesites que te refresquemos la memoria sobre los cálculos a realizar. Si es así, la tabla 1 te será de ayuda:

Operación	Cálculo
Suma: $(x_1, y_1, z_1) + (x_2, y_2, z_2)$	$(x_1 + x_2, y_1 + y_2, z_1 + z_2)$
Diferencia: $(x_1, y_1, z_1) - (x_2, y_2, z_2)$	$(x_1 - x_2, y_1 - y_2, z_1 - z_2)$
Producto escalar: $(x_1, y_1, z_1) \cdot (x_2, y_2, z_2)$	$x_1x_2 + y_1y_2 + z_1z_2$
Producto vectorial: $(x_1, y_1, z_1) \times (x_2, y_2, z_2)$	$(y_1z_2 - z_1y_2, z_1x_2 - x_1z_2, x_1y_2 - y_1x_2)$
Ángulo entre (x_1, y_1, z_1) y (x_2, y_2, z_2)	$\frac{180}{\pi} \cdot \arccos \left(\frac{x_1x_2 + y_1y_2 + z_1z_2}{\sqrt{x_1^2 + y_1^2 + z_1^2} \sqrt{x_2^2 + y_2^2 + z_2^2}} \right)$
Longitud de (x, y, z)	$\sqrt{x^2 + y^2 + z^2}$

Tras la ejecución de cada una de las acciones del menú éste reaparecerá en pantalla, a menos que la opción escogida sea la número 9. Si el usuario escoge una opción diferente, el programa advertirá al usuario de su error y el menú reaparecerá.

Las opciones 4 y 6 del menú pueden proporcionar resultados distintos en función del orden de los operandos, así que, si se escoge cualquiera de ellas, deberá mostrarse un nuevo menú que permita seleccionar el orden de los operandos. Por ejemplo, la opción 4 mostrará el siguiente menú:

- * Primer vector menos segundo vector
- * Segundo vector menos primer vector

Nuevamente, si el usuario se equivoca, se le advertirá del error y se le permitirá corregirlo.

La opción 8 del menú principal conduciría también a un submenú para que el usuario decida sobre cuál de los dos vectores se aplica el cálculo de longitud.

Ten en cuenta que tu programa debe contemplar y controlar toda posible situación excepcional: divisiones por cero, raíces con argumento negativo, etcétera. (Nota: La función arco seno se encuentra disponible en el módulo math y su identificador es acos.)

34. Realiza un programa que proporcione el desglose en billetes y monedas de una cantidad entera de euros. Recuerda que hay billetes de 500, 200, 100, 50, 20, 10 y 5 € y monedas de 2 y 1 €. Debes “recorrer” los valores de billete y moneda disponibles con uno o más bucles

```
#No es seguro si esta es la respuesta correcta o de la forma en la que se pedía hacerlo.
cantidadEuros = int(input("Introduzca la cantidad de euros: "))
billetes = [500, 200, 100, 50, 20, 10, 5, 2, 1]

for index in billetes:
    restante = cantidadEuros // index
    cantidadEuros %= index

    if restante > 0:
        print("Le queda " + str(restante) +
              (" billete" if index > 2 else " moneda") +
              ("s " if restante > 1 else " ") + str(index))
```

35. Haz un programa que vaya leyendo números y mostrándose por pantalla hasta que el usuario introduzca un número negativo. En ese momento, el programa mostrará un mensaje de despedida y finalizará su ejecución

```
while (True):
    valor = float(input("Escriba un número: "))
    if valor < 0:
        print("Sayonara")
        break
    else:
        print(valor)
```

36. Haz un programa que vaya leyendo números hasta que el usuario introduzca un número negativo. En ese momento, el programa mostrará por pantalla el número mayor de cuantos ha visto

```
mayor = 2.2250738585072014e-308
while (True):
    valor = float(input("Escriba un número: "))
    if valor > mayor:
        mayor = valor
    elif valor < 0:
        print(mayor)
        break
```

Estructura de datos

37. Diseña un programa que lea una lista de 10 enteros, pero asegurándose de que todos los números introducidos por el usuario son positivos. Cuando un número sea negativo, lo indicaremos con un mensaje y permitiremos al usuario repetir el intento cuantas veces sea preciso.

```
index = 0
lista = []

while index < 10:
    num = int(input("Ingrese número: "))
    if num > 0:
        lista.append(num)
        index += 1
    else:
        print("El número que ha introducido es negativo, no será contado como válido")
print(lista[:])
```

38. Diseña un programa que elimine de una lista todos los elementos de índice par y muestre por pantalla el resultado. (Ejemplo: si trabaja con la lista [1, 2, 1, 5, 0, 3], ésta pasará a ser [2, 5, 3].)

```
lista = [1, 2, 1, 5, 0, 3]
lista2 = []
esImpar = False
index = 0

for i in lista:
    if esImpar:
        lista2.append(lista[index])
    esImpar = not esImpar
    index += 1
lista = lista2

print(lista[:])
```

39. Diseña un programa que elimine de una lista todos los elementos de valor par y muestre por pantalla el resultado. (Ejemplo: si trabaja con la lista [1, -2, 1, -5, 0, 3], ésta pasará a ser [1, 1, -5, 3].)

```
lista = [1, -2, 1, -5, 0, 3]
lista2 = []
index = 0

for i in lista:
    if lista[index] % 2 != 0:
        lista2.append(lista[index])
    index += 1
lista = lista2

print(lista[:])
```

40. Una matriz nula es aquella que sólo contiene ceros. Construye una matriz nula de 5 filas y 5 columnas.

```
matriz = []
for i in range(5):
    matriz.append([]) # Filas
    for j in range(5):
        matriz[i].append(0) # Columnas

for i in range(len(matriz)):
    print(matriz[i])
```

41. Una matriz identidad es aquella cuyos elementos en la diagonal principal, es decir, accesibles con una expresión de la forma $M[i][i]$, valen uno y el resto valen cero. Construye una matriz identidad de 4 filas y 4 columnas.

```
matriz = []
for i in range(4):
    matriz.append([]) # Filas
    for j in range(4):
        if i == j:
            matriz[i].append(1) # Columnas
        else:
            matriz[i].append(0) # Columnas

for i in range(len(matriz)):
    print(matriz[i])
```

42. Crea la siguiente matriz utilizando la técnica del bucle

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

```
matriz = []
for i in range(4):
    matriz.append([]) # Filas
    for j in range(4):
        if i == j:
            matriz[i].append(1) # Columnas
        else:
            matriz[i].append(0) # Columnas

for i in range(len(matriz)):
    print(matriz[i])
```

43. Haz un programa que pida un entero positivo n y almacene en una variable M la matriz identidad de $n \times n$ (la que tiene unos en la diagonal principal y ceros en el resto de celdas).

```
while (True):
    n = int(input("Introduzca un entero positivo: "))
    if n > 0:
        break

m = []
for i in range(n):
    m.append([]) # Filas
    for j in range(n):
        if i == j:
            m[i].append(1) # Columnas
        else:
            m[i].append(0) # Columnas

for i in range(len(m)):
    print(m[i])
```

44. Diseña un programa que lea dos matrices y calcule la diferencia entre la primera y la segunda.

```
n1 = int(input("¿De qué tamaño es la primera matriz?: "))
m1 = []
for i in range(n1):
    m1.append([])
    for j in range(n1):
        p = int(input(f"Valor para la posición {i},{j}: "))
        m1[i].append(p)
print("Matriz primera:")
for i in range(len(m1)):
    print(m1[i])

n2 = int(input("¿De qué tamaño es la segunda matriz?: "))
m2 = []
for i in range(n2):
    m2.append([])
    for j in range(n2):
        p = int(input(f"Valor para la posición {i},{j}: "))
        m2[i].append(p)
print("Matriz primera:")
for i in range(len(m2)):
    print(m2[i])

# La parte de hacer la diferencia no sé cómo hacerlo aún
# Creo que habría de hacerlo restando con cada j en vez de con las i...
print(set(m1).difference(set(m2)))
```


45. Diseña un programa que lea una matriz y un número y devuelva una nueva matriz: la que resulta de multiplicar la matriz por el número. (El producto de un número por una matriz es la matriz que resulta de multiplicar cada elemento por dicho número.)

```
n = int(input("¿De qué tamaño es la matriz?: "))
m1 = []
for i in range(n):
    m1.append([])
    for j in range(n):
        p = int(input(f"Valor para la posición {i},{j}: "))
        m1[i].append(p)
print("Matriz:")
for i in range(len(m1)):
    print(m1[i])

n2 = int(input("¿Qué número desea multiplicar?: "))
m2 = []
for i in range(n):
    m2.append([])
    for j in range(n):
        m2[i].append(m1[i][j] * n2)
print("Matriz nueva:")
for i in range(len(m2)):
    print(m2[i])
```

46. La traspuesta de una matriz A de dimensión $m \times n$ es una matriz A^T de dimensión $n \times m$ tal que $A_{ij} = A_{ji}$. Por ejemplo, si:

$$A = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 12 & 6 \\ 1 & 0 & -3 \\ 10 & -1 & 0 \end{pmatrix}$$

Entonces:

$$A^T = \begin{pmatrix} 1 & 2 & 1 & 10 \\ 2 & 12 & 0 & -1 \\ 3 & 6 & -3 & 0 \end{pmatrix}$$

Diseña un programa que lea una matriz y muestre su traspuesta.

47. Diseña un programa tal que lea una matriz A de dimensión $m \times n$ y muestre un vector v de talla n tal que

$$v_i = \sum_{j=1}^m A_{ij}$$

para i entre 1 y n .

48. Diseña un programa que lea una cadena y muestre por pantalla una lista con todas sus palabras en minúsculas. La lista devuelta no debe contener palabras repetidas.

Por ejemplo: ante la cadena

`'Una_frase_formada_con_palabras._Otra_frase_con_otras_palabras.'`,

el programa mostrará la lista

`['una', 'frase', 'formada', 'con', 'palabras', 'otra', 'otras']`.

Observa que en la lista no aparece dos veces la palabra “frase”, aunque sí aparecía dos veces en la cadena leída.