



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### Ingeniero en computación

### Ingeniero en Software y tecnologías emergentes

**Materia:** Programación en Python / Clave COM432

**Alumno:** Emiliano Nevarez Palma

**Matrícula:** 375295

**Maestro:** Pedro Núñez Yépiz

**Actividad No. 5:**

**Tema - Unidad:** (try, for, range(), random())

**Ensenada Baja California a 10 de marzo del 2023**



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

### 1. INTRODUCCIÓN

Los temas que serán vistos serán las funciones random y range, ciclo For y la excepción Try Except

### 2. COMPETENCIA

Aprenderemos a dominar tanto las funciones mencionadas como el ciclo común, realizando problemas y practicando

### 3. FUNDAMENTOS

- Las funciones **random** en Python se utilizan para generar números aleatorios. El módulo de Python "**random**" proporciona una variedad de funciones para trabajar con números aleatorios. Estas funciones se utilizan para generar números enteros aleatorios, números de punto flotante aleatorios y secuencias aleatorias.
- el **try-except** se utiliza para manejar excepciones. Las excepciones son eventos que ocurren durante la ejecución de un programa que interrumpen el flujo normal de ejecución. El try-except permite al programador manejar estas excepciones y tomar medidas para resolverlas.
- La función **range** en Python se utiliza para generar una secuencia de números. La sintaxis de la función range es range (start, stop, step). La función range genera una secuencia de números que comienza en "start", termina en "stop" y se incrementa en "step". Por defecto, el valor de "start" es 0 y el valor de "step" es 1.
- El ciclo for en Python se utiliza para iterar sobre una secuencia (como una lista, una tupla, un rango, etc.) y ejecutar un conjunto de instrucciones para cada elemento en la secuencia. La sintaxis del ciclo for es:

for variable in secuencia:

#código a ejecutar

- El ciclo while en Python se utiliza para ejecutar un conjunto de instrucciones mientras se cumpla una condición. La sintaxis del ciclo while es:

while condición:

#código a ejecutar

### 4. PROCEDIMIENTO

**Los problemas a tratar serán los siguientes:**

**1.-** Programa en Python que genere 40 números aleatorios entre el 0 y 200, desplegar los números y la leyenda de cada número si es par o impar , la cantidad de los números pares e impares así como la suma de los números pares o impares.

**2.-** Programa en Python que despliegue la tabla de multiplicar de un número dado (número entre el 1 y 20).

Tabla del 5



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

5 \* 1 = 5

5 \* 2 = 10

..

5\*10=50

3.- Programa en Python que lea una calificación, la calificación deberá estar en el rango de 0 a 100, si hay un error de captura, mostrar mensaje de error. con la calificación correcta mostrar msg de aprobado reprobado.

4.- Programa en Python que lea n cantidad de números enteros dentro de un rango dado ( $> 0$ ), el programa deberá terminar cuando el usuario introduzca el número cero.  
desplegar la suma de números y la media.

5.- Programa en Python que sirva para leer el promedio de una materia. donde el usuario tendrá un máximo de 3 oportunidades de cursar la materia, si el promedio es aprobado, felicitarlo y continuar el siguiente semestre, si promedio es reprobado deberá salir mensaje de repetir materia o es baja académica si ha reprobado 3 veces.

### PARTE 2:

1.- función que lea n cantidad de números hasta que el usuario lo desee, desplegar la suma de los números, media y valor de los números mayores y menores.

2.- función que genere 15 números impares entre 10 y 60 o máximo de 25 números. desplegar la media de los pares y media de impares.

3.- función que sirva para leer y validar un número dentro de un rango dado por el usuario. repetir esta acción hasta que el usuario lo desee, desplegar cantidad de números y promedio de los números.

4.- función que reciba como parámetro los valores para el área de un triángulo y retorne su resultado

5.- función que sirva para validar un número dentro de un rango dado.

### 5. RESULTADOS Y CONCLUSIONES

Tuve algunas dificultades a la hora de realizar el código, pero al final de entender muy bien cada problema, se me fue facilitando cada vez mejor, aquí muestro el código:

R1:

```
import random as rm

par = 0;
impar = 0;
suma_par = 0;
suma_impar = 0;

for i in range(1, 41):
    num = rm.randint(0,200)
    if(num % 2 == 0):
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
print(f"{i} == {num} es par")
par += 1
suma_par = num + suma_par;
else:
    print(f"{i} == {num} es impar")
    impar += 1
    suma_impar = num + suma_impar;
print(f"La cantidad de los numeros pares es: {par} y la suma de todos los numeros pares es: {suma_par}")
print(f"La cantidad de los numeros impares es: {impar} y la suma de todos los numeros impares es: {suma_impar}")
```

**R2:**

```
import random as rm

num = rm.randint(1, 21)
for i in range(1, 11):
    print(f"{i} x {num} = {i * num}")
```

**R3:**

```
try:
    cal = input("Ingrese calificación del 0-100: ")
    if cal.isdigit():
        valor = int(cal)
        if valor < 0 or valor > 100:
            raise ValueError("La calificación debe estar entre 0 y 100.")
    else:
        raise TypeError("Los valores ingresados deben ser numéricos")
except ValueError as e:
    print("Error:", e)
except TypeError as er:
    print(f"Error:", er)
else:
    if valor < 60:
        print("Reprobado")
    else:
        print("Aprobado")
```

**R4:**

```
try:
    sum = 0
    contador = 0
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
n = input("Ingresa la cantidad de numeros a ingresar: ")
if n.isdigit():
    valor = int(n)
    if valor < 0:
        raise ValueError("La cantidad tiene que ser mayor a 0.")
    else:
        raise TypeError("Los valores ingresados deben ser númericos")
for i in range(valor):
    numero = input("Ingrese un número mayor a 0 ó 0 para terminar: ")
    if numero.isdigit():
        num = int(numero)
        if num < 0:
            raise ValueError("No se aceptan valores negativos.")
            continue
        sum += num
        contador += 1
    else:
        raise TypeError("Los valores ingresados deben ser númericos")
    continue
    if num == 0:
        break
except ValueError as e:
    print("Error:", e)
except TypeError as er:
    print("Error:", er)
else:
    if contador == 0:
        print("No se ingresaron números válidos.")
    else:
        media = sum / contador
        print("La suma de los números ingresados es:", sum)
        print("La media de los números ingresados es:", media)
```

**R5:**

```
oportunidad_max = 3
promedio_min = 60
oportunidad = 0
promedio = 0

for oportunidad in range(oportunidad_max):
    try:
        promedio = int(input("Ingrese el promedio de la materia "))
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
except:
    print("Error: Debe de ingresar solo valores numericos")
    continue
if promedio >= promedio_min:
    print("Felicitaciones, pasas al siguiente semestre")
    break
else:
    oportunidad += 1
    if oportunidad == oportunidad_max:
        print("Lo siento, es baja academica")
    else:
        print("Lo siento, has reprobado la materia, debes repetir")
```

### PARTE 2

R1:

```
def calcular():
    numeros = []
    while True:
        try:
            respuesta = input("¿Desea ingresar un número? (S/N): ")
            if respuesta.upper() == "N":
                break
            numero = float(input("Ingresa un número: "))
            numeros.append(numero)
        except ValueError:
            print("Error: Debe ingresar un número válido.")
            continue

    if len(numeros) == 0:
        print("No se ingresaron números.")
        return

    suma = sum(numeros)
    media = suma / len(numeros)
    mayor = max(numeros)
    menor = min(numeros)

    print("Suma de los números:", suma)
    print("Media de los números:", media)
    print("Valor máximo:", mayor)
    print("Valor mínimo:", menor)
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
calcular()
```

**R2:**

```
def generar_numeros():
    impares = []
    pares = []
    for numero in range(10, 61):
        if len(impares) == 15 or len(pares) + len(impares) == 25:
            break
        try:
            if numero % 2 == 0:
                pares.append(numero)
            else:
                impares.append(numero)
        except ValueError:
            continue

    impares_media = sum(impares) / len(impares) if len(impares) > 0 else 0
    pares_media = sum(pares) / len(pares) if len(pares) > 0 else 0

    print(f"Media de impares: {impares_media}")
    print(f"Media de pares: {pares_media}")

generar_numeros()
```

**R3:**

```
def leer_numeros_en_rango():
    numeros = []
    continuar = True
    rango_min = int(input("Introduce el límite inferior del rango: "))
    rango_max = int(input("Introduce el límite superior del rango: "))
    while continuar:
        num = int(input(f"Introduce un número entre {rango_min} y {rango_max}: "))
        try:
            if num < rango_min or num > rango_max:
                raise ValueError("El número introducido no está dentro del rango especificado")
            numeros.append(num)
        except ValueError as e:
            print("Error:", e)
        respuesta = input("¿Quieres introducir otro número? (s/n): ")
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
        if respuesta.lower() != "s":
            continuar = False
        cantidad_numeros = len(numeros)
        promedio = sum(numeros) / cantidad_numeros if cantidad_numeros > 0
    else 0
    print(f"Se introdujeron {cantidad_numeros} números.")
    print(f"El promedio de los números es: {promedio}")

leer_numeros_en_rango()
```

**R4:**

```
def calcular_area(base, altura):
    try:
        area = (base * altura) / 2
        return area
    except Exception as e:
        print("Error al calcular el área del triángulo:", e)

area = calcular_area(2, 10)
print(area)
```

**R5:**

```
import random
def validar_num(rango_min, rango_max):
    try:
        numero = random.randint(1,51)
        if rango_min <= numero <= rango_max:
            print(f"El numero fue {numero} y esta dentro del rango")
        else:
            print(f"El numero fue {numero} y esta fuera del rango")
    except ValueError:
        print("Error: Los valores deben de ser numericos")

r_min = int(input("Ingresa un rango minimo: "))
r_max = int(input("Ingresa un rango maximo: "))
validar_num(r_min, r_max)
```

## 6. ANEXOS

**R1:**





# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
12 == 19 es impar
13 == 63 es impar
14 == 71 es impar
15 == 38 es par
16 == 43 es impar
17 == 150 es par
18 == 129 es impar
19 == 197 es impar
20 == 146 es par
21 == 132 es par
22 == 97 es impar
23 == 165 es impar
24 == 130 es par
25 == 188 es par
26 == 68 es par
27 == 64 es par
28 == 11 es impar
29 == 143 es impar
30 == 195 es impar
31 == 89 es impar
32 == 96 es par
33 == 92 es par
34 == 194 es par
35 == 119 es impar
36 == 63 es impar
37 == 184 es par
38 == 10 es par
39 == 78 es par
40 == 161 es impar
La cantidad de los numeros pares es: 18 y la suma de todos los numeros pares es: 1948
La cantidad de los numeros impares es: 22 y la suma de todos los numeros impares es: 2346
```

**R2:**



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

2.- Programa en Python que despliegue la tabla de multiplicar de un número dado (número entre el 1 y 20)

```
[ ] import random as rm

num = rm.randint(1, 21)
for i in range(1, 11):
    print(f"{i} x {num} = {i * num}")
```

```
1 x 13 = 13
2 x 13 = 26
3 x 13 = 39
4 x 13 = 52
5 x 13 = 65
6 x 13 = 78
7 x 13 = 91
8 x 13 = 104
9 x 13 = 117
10 x 13 = 130
```

**R3:**

3.- Programa en Python que lea una calificación, las calificación deberá estar en el rango de 0 a 100, si hay un error de captura, mensaje de error. con la calificación correcta mostrar msg de aprobado reprobado.

```
[ ] try:
    cal = input("Ingrese calificación del 0-100: ")
    if cal.isdigit():
        valor = int(cal)
        if valor < 0 or valor > 100:
            raise ValueError("La calificación debe estar entre 0 y 100.")
        else:
            raise TypeError("Los valores ingresados deben ser numéricos")
    except ValueError as e:
        print("Error:", e)
    except TypeError as er:
        print(f"Error:", er)
    else:
        if valor < 60:
            print("Reprobado")
        else:
            print("Aprobado")
```

```
Ingrese calificación del 0-100: 88
Aprobado
```

**R4:**



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
if numero.isdigit():
    num = int(numero)
    if num < 0:
        raise ValueError("No se aceptan valores negativos.")
        continue
    sum += num
    contador += 1
else:
    raise TypeError("Los valores ingresados deben ser numéricos")
    continue
if num == 0:
    break
except ValueError as e:
    print("Error:", e)
except TypeError as er:
    print("Error:", er)
else:
    if contador == 0:
        print("No se ingresaron números válidos.")
    else:
        media = sum / contador
        print("La suma de los números ingresados es:", sum)
        print("La media de los números ingresados es:", media)
```

☞ Ingresa la cantidad de numeros a ingresar: d  
Error: Los valores ingresados deben ser numéricos

R5:



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

5.- Programa en Python que sirva para leer el promedio de una materia. donde el usuario tendrá un máximo de 3 oportunidades de cursar materia, si el promedio es aprobado, felicitarlo y continuar el siguiente semestre, si promedio es reprobado deberá salir mensaje de rep materia o es baja académica si ha reprobado 3 veces.

```
oportunidad_max = 3
promedio_min = 60
oportunidad = 0
promedio = 0

for oportunidad in range(oportunidad_max):
    try:
        promedio = int(input("Ingrese el promedio de la materia "))
    except:
        print("Error: Debe de ingresar solo valores numericos")
        continue
    if promedio >= promedio_min:
        print("Felicitaciones, pasas al siguiente semestre")
        break
    else:
        oportunidad += 1
        if oportunidad == oportunidad_max:
            print("Lo siento, es baja academica")
        else:
            print("Lo siento, has reprobado la materia, debes repetir")
```

```
Ingresa el promedio de la materia 50
Lo siento, has reprobado la materia, debes repetir
Ingresa el promedio de la materia 50
Lo siento, has reprobado la materia, debes repetir
Ingresa el promedio de la materia 40
Lo siento, es baja academica
```

## PARTE 2

### R1:

```
if len(numeros) == 0:
    print("No se ingresaron números.")
    return
suma = sum(numeros)
media = suma / len(numeros)
mayor = max(numeros)
menor = min(numeros)

print("Suma de los números:", suma)
print("Media de los números:", media)
print("Valor máximo:", mayor)
print("Valor mínimo:", menor)

calcular()
```

```
¿Desea ingresar un número? (S/N): s
Ingresa un número: 3
¿Desea ingresar un número? (S/N): s
Ingresa un número: 32
¿Desea ingresar un número? (S/N): n
Suma de los números: 35.0
Media de los números: 17.5
Valor máximo: 32.0
Valor mínimo: 3.0
```



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

**R2:**

2.- función que genere 15 números impares entre 10 y 60 o máximo de 25 números. desplegar la media de los pares y media de impa

```
def generar_numeros():  
    impares = []  
    pares = []  
    for numero in range(10, 61):  
        if len(impares) == 15 or len(pares) + len(impares) == 25:  
            break  
        try:  
            if numero % 2 == 0:  
                pares.append(numero)  
            else:  
                impares.append(numero)  
        except ValueError:  
            continue  
  
    impares_media = sum(impares) / len(impares) if len(impares) > 0 else 0  
    pares_media = sum(pares) / len(pares) if len(pares) > 0 else 0  
  
    print(f"Media de impares: {impares_media}")  
    print(f"Media de pares: {pares_media}")  
  
generar_numeros()
```

```
Media de impares: 22.0  
Media de pares: 22.0
```

**R3:**



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

```
def leer_numeros_en_rango():
    numeros = []
    continuar = True
    rango_min = int(input("Introduce el límite inferior del rango: "))
    rango_max = int(input("Introduce el límite superior del rango: "))
    while continuar:
        num = int(input(f"Introduce un número entre {rango_min} y {rango_max}: "))
        try:
            if num < rango_min or num > rango_max:
                raise ValueError("El número introducido no está dentro del rango especificado")
            numeros.append(num)
        except ValueError as e:
            print("Error:", e)
        respuesta = input("¿Quieres introducir otro número? (s/n): ")
        if respuesta.lower() != "s":
            continuar = False
    cantidad_numeros = len(numeros)
    promedio = sum(numeros) / cantidad_numeros if cantidad_numeros > 0 else 0
    print(f"Se introdujeron {cantidad_numeros} números.")
    print(f"El promedio de los números es: {promedio}")
    leer_numeros_en_rango()
```

```
Introduce el límite inferior del rango: 10
Introduce el límite superior del rango: 50
Introduce un número entre 10 y 50: 40
¿Quieres introducir otro número? (s/n): n
Se introdujeron 1 números.
El promedio de los números es: 40.0
```

R4:

4.- función que reciba como parámetro los valores para el área de un triángulo y retorne su resultado

```
def calcular_area(base, altura):
    try:
        area = (base * altura) / 2
        return area
    except Exception as e:
        print("Error al calcular el área del triángulo:", e)

area = calcular_area(2, 10)
print(area)
```

```
10.0
```

R5:



# Universidad Autónoma de Baja California

## Facultad de Ingeniería Arquitectura y Diseño

5.- función que sirva para validar un número dentro de un rango dado.



```
import random
def validar_num(rango_min, rango_max):
    try:
        numero = random.randint(1,51)
        if rango_min <= numero <= rango_max:
            print(f"El numero fue {numero} y esta dentro del rango")
        else:
            print(f"El numero fue {numero} y esta fuera del rango")
    except ValueError:
        print("Error: Los valores deben de ser numericos")

r_min = int(input("Ingresa un rango minimo: "))
r_max = int(input("Ingresa un rango maximo: "))
validar_num(r_min, r_max)
```

[https://colab.research.google.com/drive/17eiA8FxydJYQWHeemP45M93qSkZEM-qL?usp=share\\_link](https://colab.research.google.com/drive/17eiA8FxydJYQWHeemP45M93qSkZEM-qL?usp=share_link)

## 7. REFERENCIAS

Referencia colaboratory:

[https://colab.research.google.com/drive/17eiA8FxydJYQWHeemP45M93qSkZEM-qL?usp=share\\_link](https://colab.research.google.com/drive/17eiA8FxydJYQWHeemP45M93qSkZEM-qL?usp=share_link)

<https://www.freecodecamp.org/espanol/news/guia-de-funciones-de-python-con-ejemplos/>

<https://www.programarya.com/Cursos/Python/Ciclos/Ciclo-while>

<https://codingornot.com/07-python-ciclos-ejemplos>

<https://controlautomaticoeducacion.com/python-desde-cero/ciclo-while-en-python/>

<https://uniwebsidad.com/libros/python/capitulo-4/definiendo-funciones>