

```

#Segundo Parcial Modelo 2 del jueves
"""

Pregunta 2
Ingresar numeros positivos hasta cargar un arreglo con tres numeros pares multiplos de 7
y 3 numeros impares multiplos de 3 en ese orden.
mostrar el arreglo ejemplo se ingresan 21 14 2 15 28 31 7 42 27
el arreglo queda: 14 28 42 21 15 27
Ingresa números positivos hasta cargar un arreglo con
1. Tres números pares múltiplos de 7.
2. Tres números impares múltiplos de 3.
3. Muestra el arreglo resultante.
"""

def cargar_datos(arr_par7,arr_mult3,arreglo):

    cont_arr_par_7 = 0 # Contador para pares
    cont_arr_mult3 = 0
    i=0
    while len(arr_par7) < 3 or len(arr_mult3) < 3:
        numeros = int(input("Ingrese un número: "))
        if len(arr_par7) < 3:
            # Condición: Par (num % 2 == 0) Y Múltiplo de 7 (num % 7 == 0)
            if numeros % 2 == 0 and numeros % 7 == 0:
                arr_par7.append(numeros)
                cont_arr_par_7 += 1
                print(f"Agregado: {numeros}. Pares Múltiplos de 7 cargados: {cont_arr_par_7}/3")

        elif len(arr_par7) == 3 and len(arr_mult3) < 3:
            # Condición: Impar (numeros % 2 != 0) Y Múltiplo de 3 (numeros % 3 == 0)
            if numeros % 2 != 0 and numeros % 3 == 0:
                arr_mult3.append(numeros)
                cont_arr_mult3 += 1
                print(f"Agregado: {numeros}. Impares Múltiplos de 3 cargados: {cont_arr_mult3}/3")

        else:
            print(f"{numeros} no es impar múltiplo de 3. Ignorado.")
        numeros = int(input("Ingrese un número: "))
        arreglo.append(numeros)
        # Si el número no cumple la condición actual, no se hace nada y se pide otro
        # número
        print("Número ingresado no cumple con la condición actual o la condición ya está
completa.")

def mostrar_datos(arr_par7,arr_mult3):
    print(f"El arreglo resultante es: {arr_par7}+{arr_mult3}")

print("\n-----")
arreglo = []
pares7= []
multiplos3 = []
cargar_datos(pares7,multiplos3)

if len(pares7) >0:
    print("Cargando arreglo con 3 pares múltiplos de 7 y 3 impares múltiplos de 3...")
    print("\n-----")
    mostrar_datos(pares7,multiplos3,arreglo)
else:
    print("\n-----")
    print("No se pudieron cargar ambos arreglos completamente.")
    print(f"Pares/Múltiplos de 7 cargados: {len(pares7)}/3")

```

```
print(f"Impares/Múltiplos de 3 cargados: {len(multiplos3)}/3")
print("\n-----")
```