

```
In [ ]: import pandas as pd
        from pandas import ExcelWriter
        from pandas import ExcelFile

        grupos_totales=3

        #Se importa el listado de la población ( muestra con numeros aleatorios en Excel)
        #respetando las primisas del problema planteado

        df1 = pd.read_excel('Listado_Muestras.xlsx', sheet_name='Hoja1')
        df1

In [ ]: # Acomodo de forma ascendente los valores de la columna 'Peso':
        df1 = df1.sort_values(by='PESO', ascending=True)
        df1

In [ ]: #Realizo el acomodo del ID para mantener de menor a mayor la tabla
        df1.ID = range(df1.shape[0])
        df1

In [ ]: # Ordenar por los valores de la columna 'Peso' para la tabla:
        tabla = df1.sort_values(by='PESO', ascending=True)

In [ ]: # Ordenar por los valores de la columna 'Peso' para la tabla2:
        tabla2 = df1.sort_values(by='PESO', ascending=True)

In [ ]: # Ordenar por los valores de la columna 'Peso' para la tabla3:
        tabla3 = df1.sort_values(by='PESO', ascending=True)

In [ ]: #Realizo la cuenta de cuantos especimenes hay en el archivo para saber de cuantos
        #especimenes van a ser mis grupos

        cuenta = 0
        for i in df1.index:
            cuenta = cuenta + 1
        print(cuenta)

In [ ]: # Realizo la división de los especimenes totales y los grupos totales que de acuerdo
        #al problema son 3
        muestras = cuenta // grupos_totales
        print(muestras)

In [ ]: #Calculo el residuo de la division de los especimenes vs grupos totales
        sobrante =cuenta % 3
        print(sobrante)
```

```
In [ ]: #Realizo una comparación de los especímenes para tomar los primeros elementos
        de la variable "muestras"
        #y agrego una columna para el grupo 1

        tabla['Grupo_1'] = tabla['ID'].apply(lambda x: 'True' if x <= muestras else 'False')

        # Ordeno los valores de la columna 'Grupo 1' y elimino, los que estaría fuera
        del rango de acuerdo a la variable muestras

        grupo1 = tabla.sort_values(by='Grupo_1', ascending=False)
        grupo1 = grupo1.drop(grupo1[grupo1.ID >= muestras].index)

        #Exporto el listado
        grupo1.to_excel("Grupo_1.xlsx", sheet_name='Grupo_1')

        #Imprimo la tabla
        grupo1
```

```
In [ ]: #Realizo una comparación de los especímenes para tomar los primeros elementos
        de la variable "muestras"
        #y agrego una columna para el grupo 2

        tabla2['Grupo_2'] = tabla2['ID'].apply(lambda x: 'True' if ((x >= muestras) &
        (x <=(muestras*2+sobranante))) else 'False')

        # Ordeno los valores de la columna 'Grupo 2' y elimino los que estaría fuera d
        el rango de acuerdo a la variable muestras

        grupo2 = tabla2.sort_values(by='Grupo_2', ascending=False)
        grupo2 = grupo2.drop(grupo2[grupo2.ID < (muestras)].index)
        grupo2 = grupo2.drop(grupo2[grupo2.ID >= (muestras*2+sobranante)].index)

        #Exporto el listado
        grupo2.to_excel("Grupo_2.xlsx", sheet_name='Grupo_2')

        #Imprimo la tabla
        grupo2
```

```
In [ ]: #Realizo una comparación de los especímenes para tomar los primeros elementos
        de la variable "muestras"
        #y agrego una columna para el grupo 3

        tabla3['Grupo_3'] = tabla3['ID'].apply(lambda x: 'True' if ((x >= muestras*2+s
        obrante)) else 'False')

        # Ordeno los valores de la columna 'Grupo 3' y elimino los que estaría fuera d
        el rango de acuerdo a la variable muestras

        grupo3 = tabla3.sort_values(by='Grupo_3', ascending=False)
        grupo3 = grupo3.drop(grupo3[grupo3.ID < (muestras*2+sobrante)].index)

        #Exporto el listado
        grupo3.to_excel("Grupo_3.xlsx", sheet_name='Grupo_3')

        #Imprimo la tabla
        grupo3
```

De esa manera se realizan los 3 grupos indicados en la actividad con los pesos lo más parecidos.