

DOCUMENTACIÓN DE PROYECTO INTEGRADOR DE APRENDIZAJE

Programación básica

Gráficas y Excel

Perla Cristal López Álvarez - 2225483 | Raquel Guadalupe Gloria Vázquez - 2011258

Grupo: 073

La API seleccionada fue: **Spotify**

Definición del Problema

Análisis de Coachella 2025, centrado en los artistas, así como sus seguidores, géneros musicales y nivel de popularidad de cada uno. Haciendo una comparación entre cada artista por día y las estadísticas generales de cada día. A continuación, se muestra el póster de Coachella 2025 que servirá de base para recolectar los artistas que se presentaron.



Almacenamiento de Datos en Excel.

En la documentación anterior se realizaron algunos cálculos estadísticos para poder realizar un análisis y comparación con ayuda de graficas en esta entrega.

Para poder ver de manera ordena los datos para las gráficas, lo que se realizo fue pasar las listas de artistas, seguidores por día del festival y la popularidad de cada artista por día previamente creadas aun libro de Excel con ayuda del módulo de Openpyxl.

Antes de realizar cualquier acción en Excel es importar el módulo, junto con las clases necesarias para poder manejar un libro en Excel.

```
from openpyxl import Workbook, load_workbook
```

Con un bloque try cargamos el libro de Excel llamado Coachella, en caso de que no exista con el except manejara el error, y creara en libro y activara la página que por default se crea cuando se genera un libro de Excel, y la llamaremos Artista del festival.

```
try:
    libro = load_workbook("Coachella.xlsx")
except FileNotFoundError:
    libro = Workbook()
    pagina = libro.active
    pagina.title = "Artistas_del festival"
```

Se define una función llamada Artistas_coachella con parámetro la lista eventos que tiene todos los artistas que se presentaron en el festival durante el fin de semana.

```
def Artistas_coachella(evento):
```

Dentro de la primera hoja generada en el libro vamos a escribir, así que definimos los encabezados para cada columna.

```
hoja = libro["Artistas_del festival"]

print("La página activa es:", hoja.title)

hoja["A1"] = "nombre"
hoja["B1"] = "id"
hoja["C1"] = "genero"
hoja["D1"] = "seguidores"
hoja["E1"] = "popularidad"
```

Mostramos en el intérprete de Python los encabezados de celda

```
print("Los encabezados de las celdas son:")
print(hoja["A1"].value, end="\t")
print(hoja["B1"].value, end="\t")
print(hoja["C1"].value, end="\t")
print(hoja["D1"].value, end="\t")
print(hoja["E1"].value)
```

Después con un ciclo for empezamos a escribir en la hoja de Excel.

```
count = 2
for artista in evento:
    hoja.cell(count, 1, artista['nombre'])
    hoja.cell(count, 2, artista['id'])
    hoja.cell(count, 3, artista['genero'])
    hoja.cell(count, 4, artista['seguidores'])
    hoja.cell(count, 5, artista['popularidad'])
    count += 1
```

Se declara otra función, pero ahora para los datos de seguidores por día del festival, así que se crea otra hoja con el nombre de seguidores Coachella.

```
def seguidores(lista_dia1_seguidores, lista_dia2_seguidores, lista_dia3_seguidores):
    hoja = libro.create_sheet("seguidores_coachella")
    print("La página activa es:", hoja.title)
```

Se define los encabezados de cada columna, que son los seguidores de día 1, 2 y 3, y se muestran en el intérprete el encabezado de las celdas

```
hoja["A1"] = "seguidores_dia1"
hoja["B1"] = "seguidores_dia2"
hoja["C1"] = "seguidores_dia3"

print("Los encabezados de las celdas son:")
print(hoja["A1"].value, end="\t")
print(hoja["B1"].value, end="\t")
print(hoja["C1"].value)
```

Y se escribe en la hoja de seguidores correspondiente al día.

```

count = 2
for i in range(len(lista_dia1_seguidores)):
    hoja.cell(count, 1, lista_dia1_seguidores[i])
    count += 1

count = 2
for i in range(len(lista_dia2_seguidores)):
    hoja.cell(count, 2, lista_dia2_seguidores[i])
    count += 1

count = 2

for i in range(len(lista_dia3_seguidores)):
    hoja.cell(count, 3, lista_dia3_seguidores[i])
    count += 1

```

Y por último se hace otra función, pero para la popularidad, creando otra hoja dentro del libro con el nombre de popularidad_coachella

```

hoja = libro.create_sheet("popularidad_coachella")

print("La página activa es:", hoja.title)

```

Se definen los encabezados.

```

hoja["A1"] = "Popularidad_dia1"
hoja["B1"] = "Popularidad_dia2"
hoja["C1"] = "Popularidad_dia3"

print("Los encabezados de las celdas son:")
print(hoja["A1"].value, end="\t")
print(hoja["B1"].value, end="\t")
print(hoja["C1"].value)

```

Y se escribe en el Excel.

```

count = 2
for i in range(len(lista_dia1_popularidad)):
    hoja.cell(count, 1, lista_dia1_popularidad[i])
    count += 1

count = 2
for i in range(len(lista_dia2_popularidad)):
    hoja.cell(count, 2, lista_dia2_popularidad[i])
    count += 1

count = 2

for i in range(len(lista_dia3_popularidad)):
    hoja.cell(count, 3, lista_dia3_popularidad[i])
    count += 1

```

Llamamos a las funciones y guardamos el archivo.

```

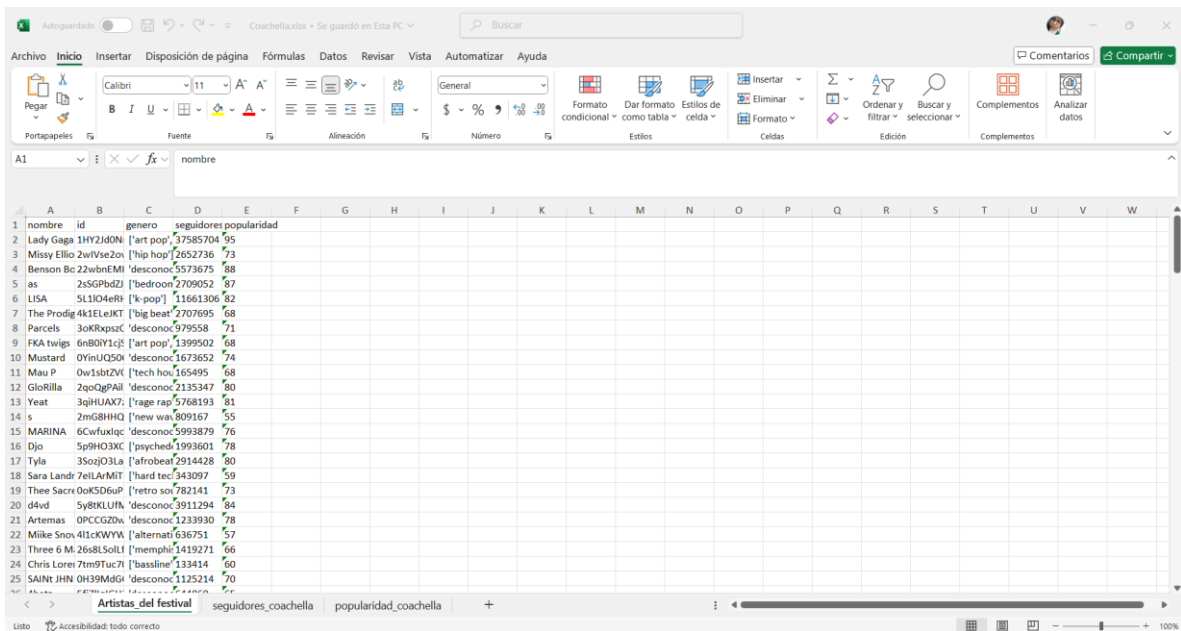
Artistas_coachella(evento)
seguidores(lista_dia1_seguidores, lista_dia2_seguidores, lista_dia3_seguidores)
popularidad(lista_dia1_popularidad, lista_dia2_popularidad, lista_dia3_popularidad)

|
print(libro.sheetnames)
libro.save("Coachella.xlsx")

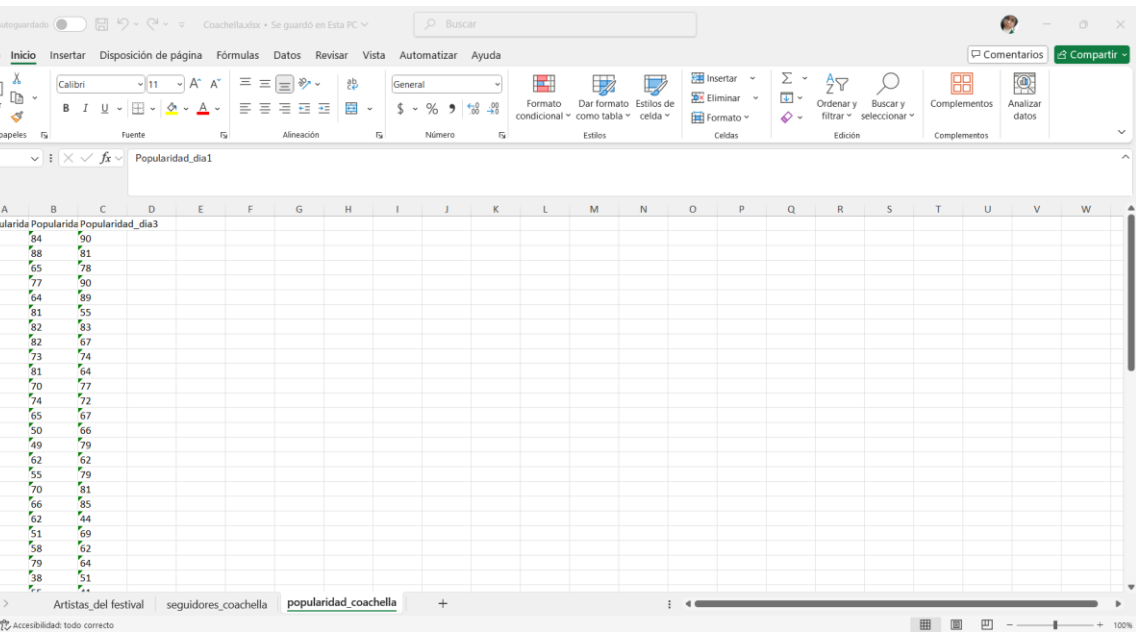
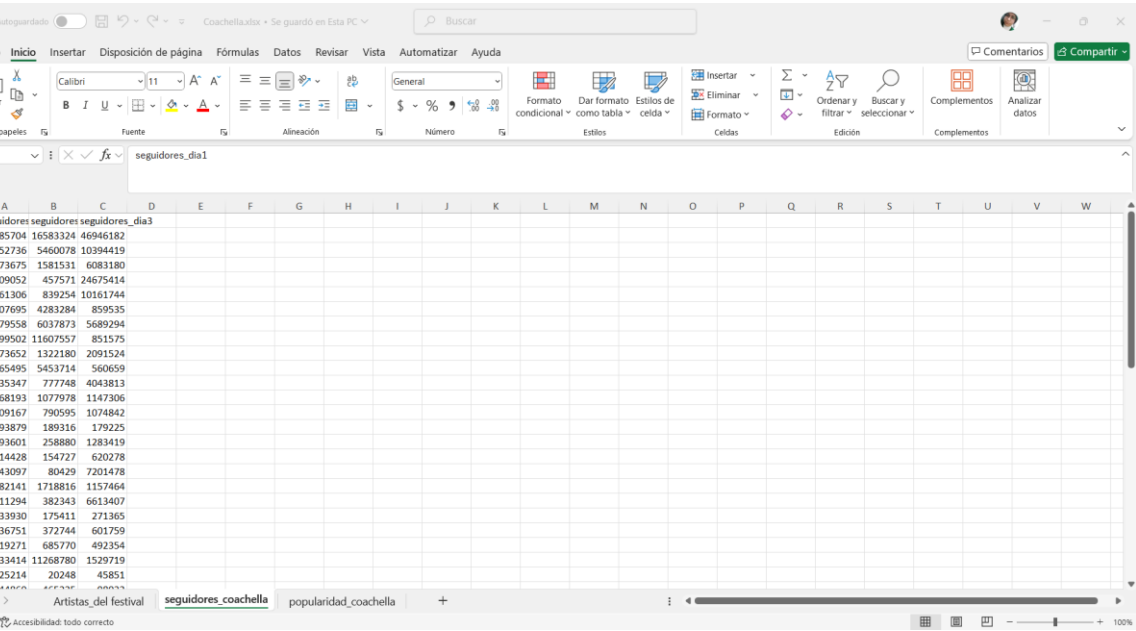
```

Una vez que se ejecute el script, se generara el archivo de Excel.

Artistas del festival



nombre	id	genero	seguidores popularidad
Lady Gaga	1HY2Jd0N	[art pop]	37585704
Missy Elliot	2wVse2on	[hip hop]	2652736
Benson Bo	22wbnEMl	[desconoc]	5573679
as	2SGOpb2j	[bedroom]	2709052
LISA	5L1IO4eR9	[k-pop]	11661306
The Prodig	4k1ELeJKT	[big beat]	2707695
Parcels	3oKRpxszC	[desconoc]	979558
FKA twigs	6n8QY1cj5	[art pop]	1399502
Mustard	0YinUQ50i	[desconoc]	1673652
Mau P	0w1sbZVv	[tech hou]	165495
GloRilla	2zoQgPail	[desconoc]	2135347
Yeat	3qHUAx7j	[rage rap]	5768193
s	2mGBHHQ	[new wa]	809167
MARINA	6Cwfluxlc	[desconoc]	5993879
Djo	5p9HO3XC	[psyched]	1993601
Tyla	35o3O3ls	[afrobeat]	2514428
Sara Land	7eLLuAMT	[hard roc]	543097
Three Sacri	0oKSD6uP	[retro so]	782141
d4vd	5y8KLUFn	[desconoc]	5911294
Artemas	0PCCGZ0w	[desconoc]	1233930
Mike Snov	41cKWYVA	[alternati]	636751
Three 6 M	26s8LSoll	[memphi]	1419271
Chris Lorei	7tm9Tuc7i	[bassline]	133414
SAINT JHN	0H39MDGf	[desconoc]	1125214



Diseño de Gráficas.

En un ciclo while true, en el cual siempre va a ejecutarse, definimos una variable llamada menú para mostrar en pantalla que grafica se pueden realizar y se pueden mostrar

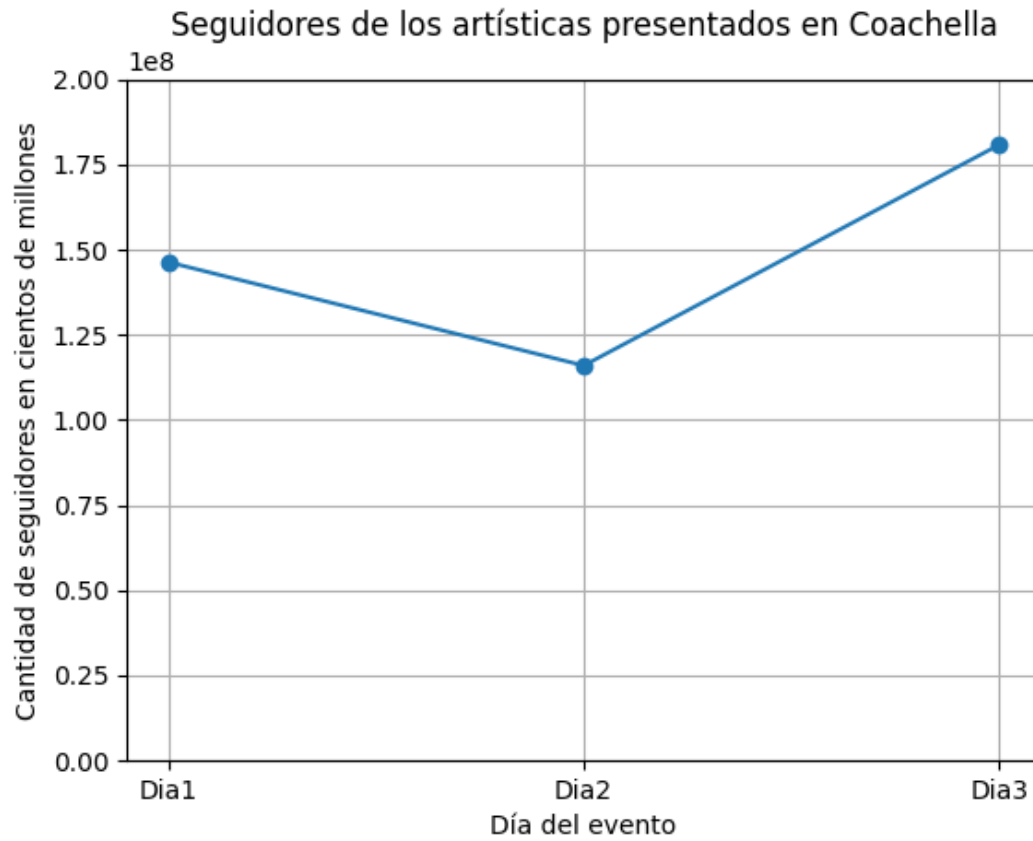
```
while True:
    menu = ""¿Qué gráfica quieres ver?
    1. Gráfica lineal de seguidores.
    2. Gráfica de pastel para seguidores.
    3. Histograma de popularidad.
    4. Gráfica de barras horizontales para géneros
    5. Salir""
    print(menu)
```

Con una variable llamada grafica guardamos el valor de la opción de la gráfica a elegir.

```
grafica=int(input("ingrese la grafia que desee ver: "))
```

Si se ingresa la opción 1, se va genera la gráfica lineal de seguidores, en donde realiza el análisis de saber aproximadamente cuantos seguidores (asistentes) asistieron en cada día del festival, y obtener la conclusión de en qué día hubo más asistentes.

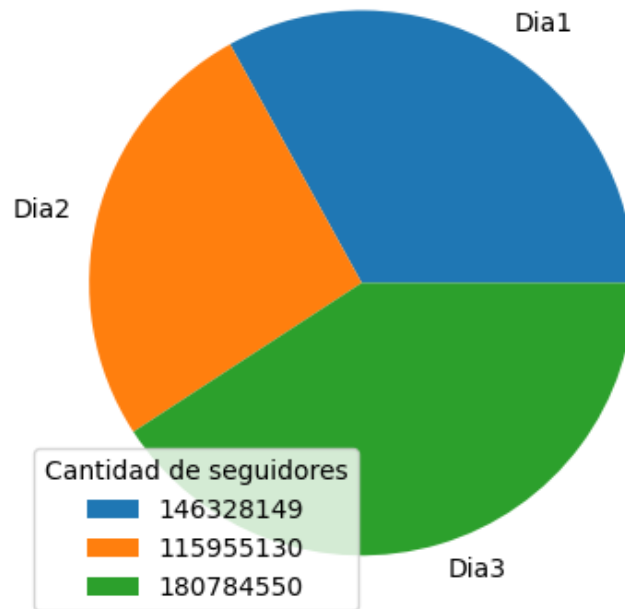
```
if grafica==1:
    plt.plot(seguidores_lineal_x,seguidores_lineal_y, marker='o')
    plt.title("Seguidores de los artísticas presentados en Coachella")
    plt.xlabel("Día del evento")
    plt.ylabel("Cantidad de seguidores en cientos de millones")
    plt.axis(ymin=0,ymax=200000000)
    plt.grid()
    plt.show()
```



Si la opción es la 2, se genera la grafica de pastel de seguidores, es otra manera de ver la primera gráfica, ya que es con el mismo objetivo.

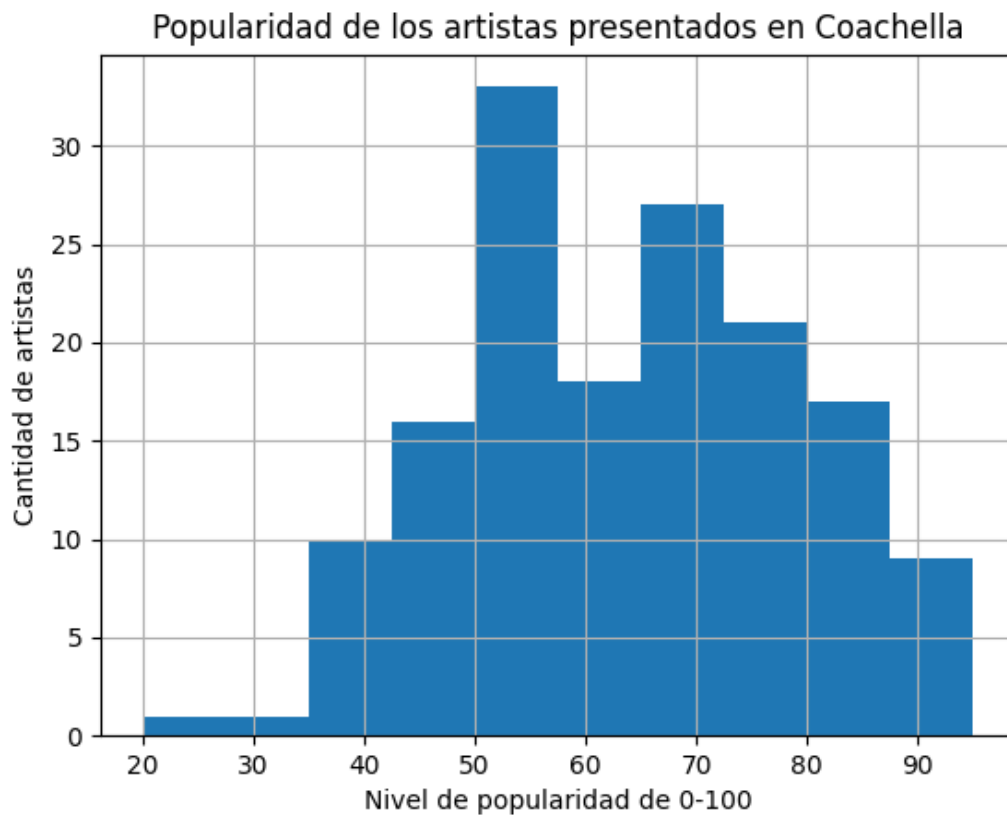
```
plt.show()  
elif grafica==2:  
    plt.pie(seguidores_pie_y, labels=seguidores_pie_label)  
    plt.title("Seguidores de los artísticas presentados en Coachella")  
    plt.legend(seguidores_pie_y, title="Cantidad de seguidores")  
    plt.show()
```


Seguidores de los artísticas presentados en Coachella



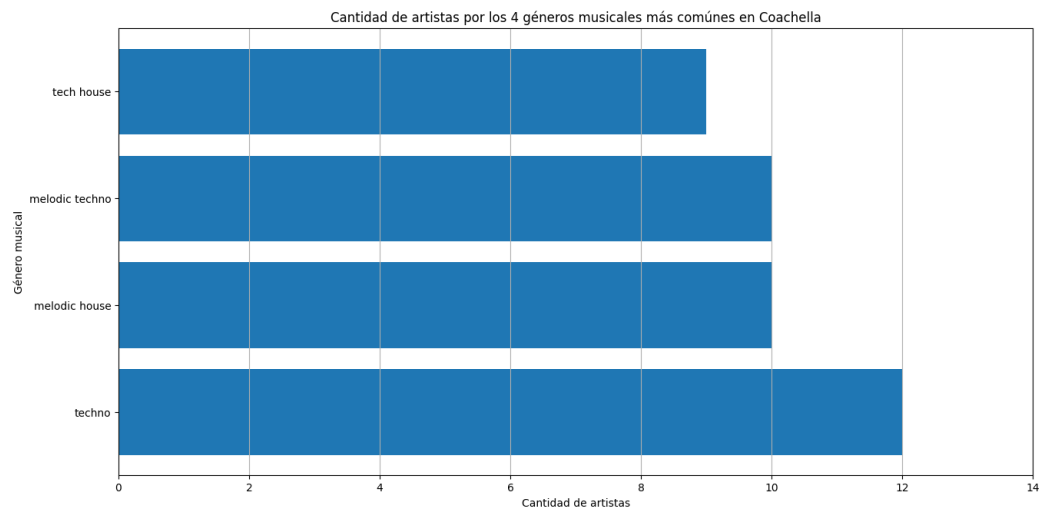
La opción 3, para la comparación de popularidad de los artistas, es decir, poder analizar el cuanto artista cuenta con la misma popularidad.

```
elif grafica==3:
    plt.hist(popularidad_hist)
    plt.title("Popularidad de los artistas presentados en Coachella")
    plt.xlabel("Nivel de popularidad de 0-100")
    plt.ylabel("Cantidad de artistas")
    plt.grid()
    plt.show()
```



La última gráfica, es para saber que cantidad de artistas que hay por los 4 géneros musicales más comunes en el festival.

```
elif grafica==4:
    plt.barh(generos_barh_x, generos_barh_y)
    plt.title("Cantidad de artistas por los 4 géneros musicales más comunes")
    plt.xlabel("Cantidad de artistas")
    plt.ylabel("Género musical")
    plt.axis(xmin=0, xmax=14)
    plt.grid(axis="x")
    plt.show()
```



la última opción es para salir del programa, además de validar las opciones que, de entrada, en caso de que se ingrese una opción inválida.

```
elif grafica==5:  
    break  
else:  
    grafica=int(input("Este valor no es aceptable, ingresa otro: "))
```

Validación de Datos para Visualización.

Para poder realizar las gráficas primero tenemos que preparar los datos

en la primera línea se crea una lista con elementos que son la suma de los seguidores, esta lista es para el eje y , para el eje x en la segunda línea, se crea una lista con los días para que sean las etiquetas para la gráfica de líneas

```
seguidores_lineal_y=[suma_dia1,suma_dia2,suma_dia3]
seguidores_lineal_x=["Dia1","Dia2","Dia3"]
```

Después usamos los mismos datos para ahora para la gráfica de pastel

```
seguidores_pie_y=[suma_dia1,suma_dia2,suma_dia3]
seguidores_pie_label=["Dia1","Dia2","Dia3"]
```

Se crea una lista nueva llamada hist para el histograma de popularidad

```
hist=[int(c) for c in lista_popularidad]
popularidad_hist = hist
```

Para la gráfica de barras se usarán de forma horizontal, esta para la gráfica de los géneros

```
generos_barh_y=[]
generos_barh_x=[]
```

Se recorre una lista de las 5 categorías más comunes en frec_coachella, con los géneros musicales más frecuentes en el festival.

```
for genero in frec_coachella.most_common(5):
```

En el condicional si el género es desconocido, no se incluye en la gráfica, y si no es desconocido se agrega el nombre del género a la lista generos_barh_x y la frecuencia a la de generos_barh_y.

```
if genero[0]=="desconocido":
    pass
else:
    generos_barh_x.append(genero[0])
    generos_barh_y.append(genero[1])
```