

# Programación básica con Python

Taller de programación básica con Python para  
análisis y visualización de datos

**TECHSCHOOL**

# ~ Estructura

1. Presentación personal
2. ¿Qué lograremos hoy?
3. Conceptos iniciales
4. Introducción a Python
5. Análisis de datos con Pandas y visualización con Altair
6. ¡Manos al código!

# Presentación personal

**TECHSCHOOL**

# ~ Francisca B. Medina Concha

Docente de Narración Gráfica  
Diseñadora UX en Cencosud  
Desarrolladora web freelance

frani.be

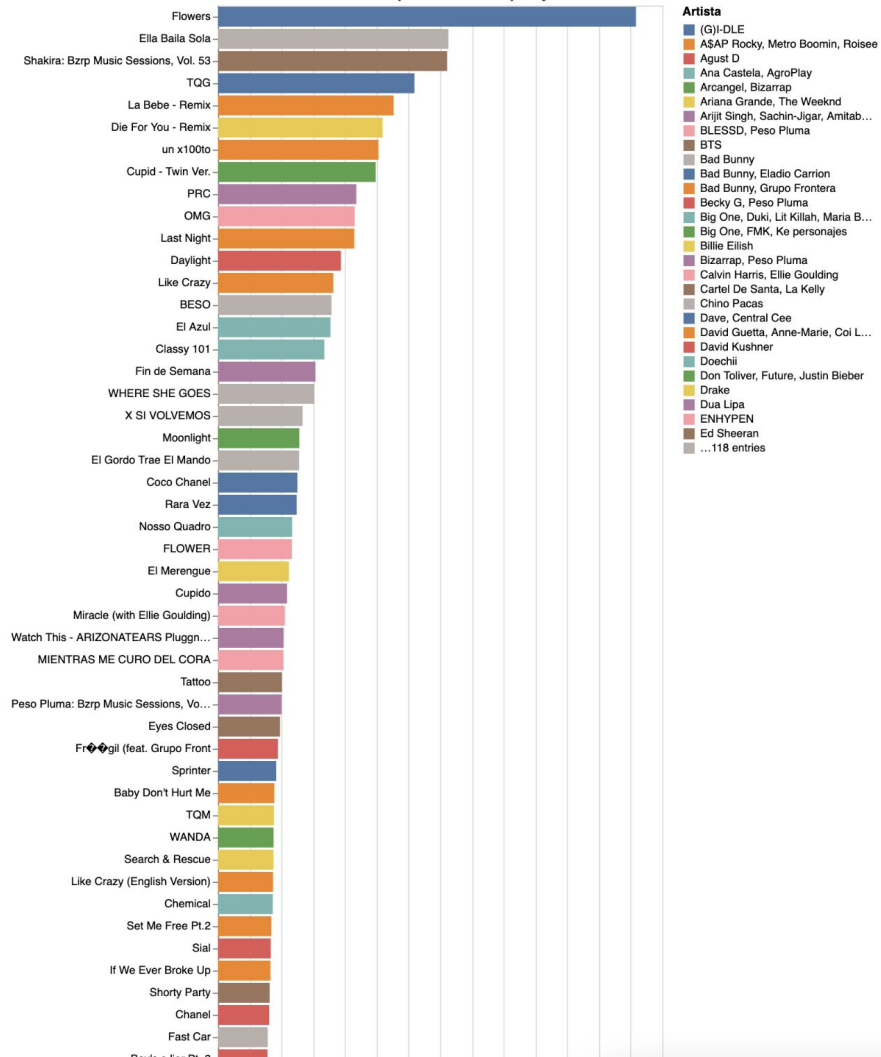


The background of the slide features a complex, light blue geometric pattern. It consists of numerous hexagons of varying sizes, some of which are interconnected by thin lines, creating a network-like or molecular structure. The overall effect is a subtle, technical, and modern aesthetic.

¿Qué lograremos hoy?

**TECHSCHOOL**

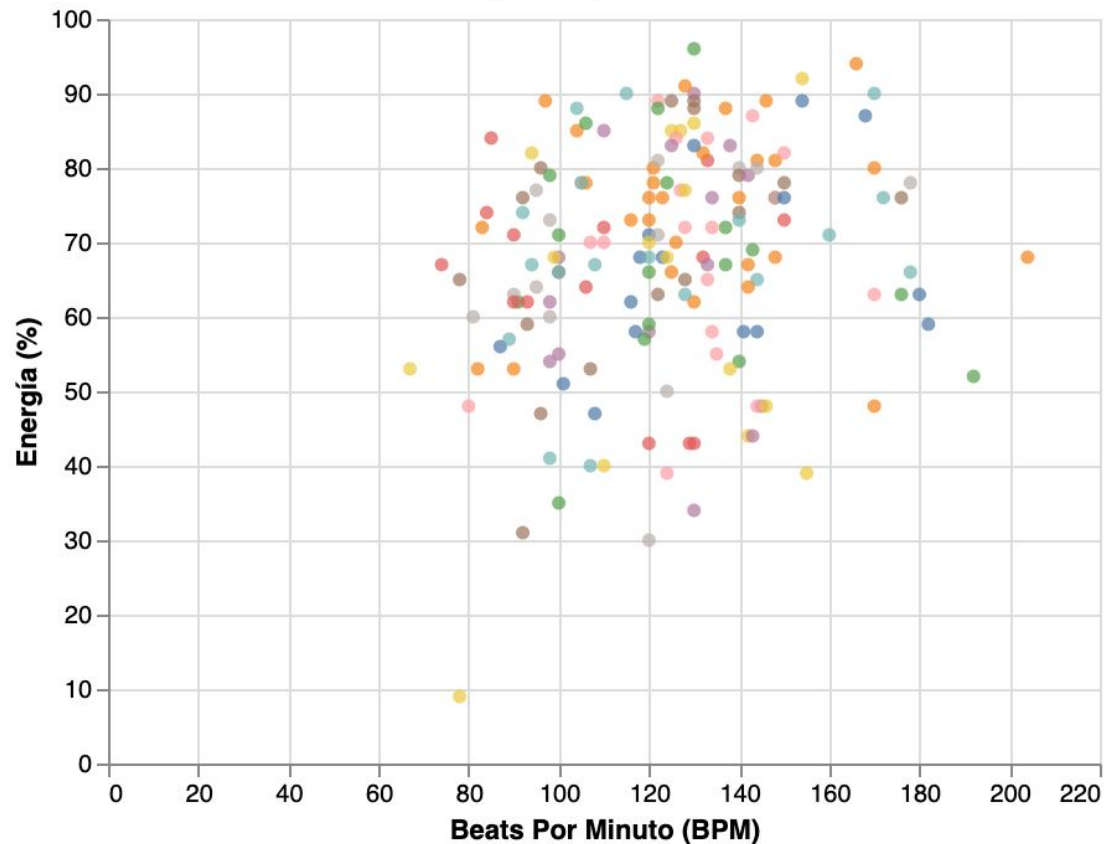
# Canciones más reproducidas en Spotify en 2023



- Artista
- (G)I-DLE
  - ASAP Rocky, Metro Boomin, Roisee
  - Agust D
  - Ana Castela, AgroPlay
  - Arcangel, Bizarrap
  - Ariana Grande, The Weeknd
  - Arijit Singh, Sachin-Jigar, Amitab...
  - BLESSD, Peso Pluma
  - BTS
  - Bad Bunny
  - Bad Bunny, Eladio Carrion
  - Bad Bunny, Grupo Frontera
  - Becky G, Peso Pluma
  - Big One, Duki, Lit Killah, Maria B...
  - Big One, FMK, Ke personajes
  - Billie Eilish
  - Bizarrap, Peso Pluma
  - Calvin Harris, Ellie Goulding
  - Cartel De Santa, La Kelly
  - Chino Pacas
  - Dave, Central Cee
  - David Guetta, Anne-Marie, Col L...
  - David Kushner
  - Doechi
  - Don Toliver, Future, Justin Bieber
  - Drake
  - Dua Lipa
  - ENHYPEN
  - Ed Sheeran
  - ...118 entries



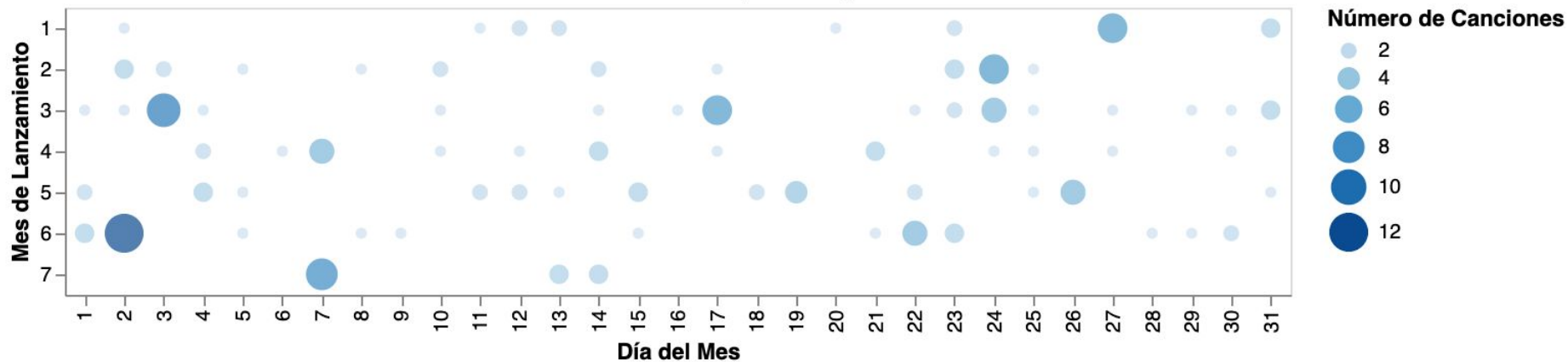
# Relación entre BPM y Energía de las Canciones en 2023



## Artista

- (G)I-DLE
- A\$AP Rocky, Metro Boomin, Roisee
- Agust D
- Ana Castela, AgroPlay
- Arcangel, Bizarrap
- Ariana Grande, The Weeknd
- Arijit Singh, Sachin-Jigar, Amitab...
- BLESSD, Peso Pluma
- BTS
- Bad Bunny
- Bad Bunny, Eladio Carrion
- Bad Bunny, Grupo Frontera
- Becky G, Peso Pluma
- Big One, Duki, Lit Killah, Maria B...
- Big One, FMK, Ke personajes
- Billie Eilish
- Bizarrap, Peso Pluma
- Calvin Harris, Ellie Goulding
- Cartel De Santa, La Kelly
- Chino Pacas
- Dave, Central Cee
- David Guetta, Anne-Marie, Coi L...
- David Kushner
- Doechii
- Don Toliver, Future, Justin Bieber
- Drake
- Dua Lipa
- ENHYPEN
- Ed Sheeran
- ...118 entries

Número de Canciones lanzadas por Día y Mes en 2023





# Conceptos iniciales

**TECHSCHOOL**

~ ¿Qué es la programación?



**TECHSCHOOL**

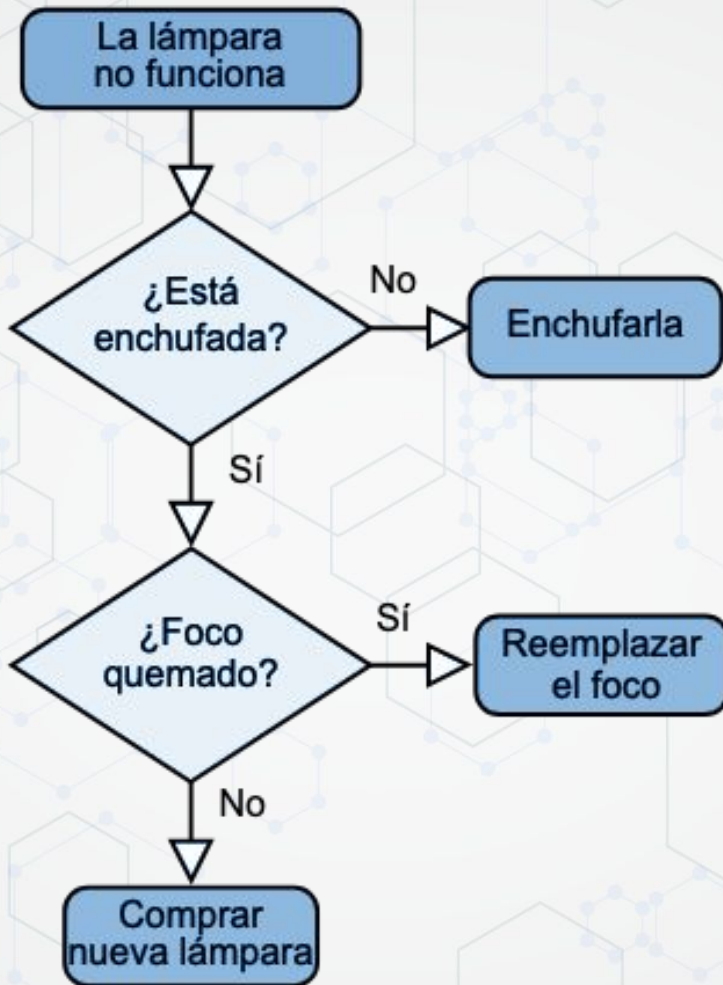
# ~ ¿Qué es la programación?

Indicar al dispositivo computacional lo que tiene que hacer,  
mediante **algoritmos**

# ~ Concepto de algoritmo

Conjunto ordenado y finito de operaciones que permite hallar la solución de un problema.

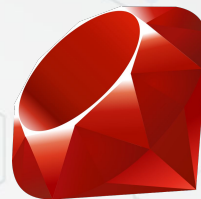
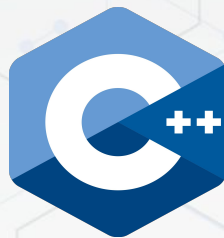
Se describe en tres partes básicas: entrada (escritura), proceso (ejecución) y salida (resultado).



# ~ Lenguaje de programación

Es una **herramienta traductora** de un **lenguaje fuente** basado en algoritmos, a un **lenguaje máquina** comprensible por el computador, para la realización de un determinado proceso.





Y muchos, MUCHOS más...

**TECHSCHOOL**



## ~ Librería

Una librería es una colección preescrita de código que los desarrolladores pueden utilizar para realizar tareas comunes o repetitivas. En lugar de escribir código desde cero para realizar una función específica, un desarrollador puede usar una función predefinida desde una librería.

## ~ Análisis de datos

El análisis de datos se refiere al proceso de inspeccionar, limpiar, transformar y modelar datos con el objetivo de descubrir información útil, llegar a conclusiones y apoyar la toma de decisiones. Utiliza técnicas cuantitativas y cualitativas para entender patrones, relaciones y estructuras dentro de los datos.

# ~ ¿Qué son los datos?

En la vida cotidiana usamos la palabra "**datos**" de manera general como sinónimo de "**información**", pero debemos comenzar a entenderlo como un **concepto más específico** para este curso.

# ~ ¿Qué son los datos?

Se refiere a hechos, eventos, transacciones, entre otros, que han sido registrados.

Es la unidad más pequeña de información \*fáctica.

Puede ser empleado para el cálculo, el razonamiento, la discusión, la inferencia, la deducción, y su presentación o visualización.

**Fáctico:** Alude a aquello vinculado a los hechos. El término suele usarse para calificar a lo que depende de los hechos o se fundamenta en ellos

# ~Diferencia entre datos e información

## Datos

Entrada sin procesar de la cual se produce la información.

## Información

Datos procesados y comunicados para ser interpretados por el receptor.

Un dato por sí mismo no constituye información, es el procesamiento de los datos lo que nos proporciona información.

# ~Diferencia entre datos e información





# ~ Herramientas para el análisis de datos

**Excel:** Una herramienta básica pero poderosa para la manipulación y visualización de datos.

**Python (especialmente con bibliotecas como Pandas):** Lenguaje de programación versátil para análisis de datos y aprendizaje automático.

**R:** Lenguaje de programación diseñado específicamente para análisis estadístico.

**Tableau:** Herramienta de visualización de datos para transformar datos crudos en gráficos interactivos.

**SQL:** Lenguaje de consulta estructurado para gestionar y recuperar datos de bases de datos relacionales.

**Power BI:** Herramienta de Microsoft para visualización y análisis de datos.

**TECHSCHOOL**



## ~ Visualización de datos

La visualización de datos es el acto de representar información en forma gráfica. Permite a los usuarios ver y comprender tendencias, patrones y correlaciones que podrían no ser evidentes en los datos brutos. La visualización eficiente de datos puede ayudar a simplificar complejas relaciones de datos y facilitar su interpretación.

# ~ Importancia

Comprensión rápida

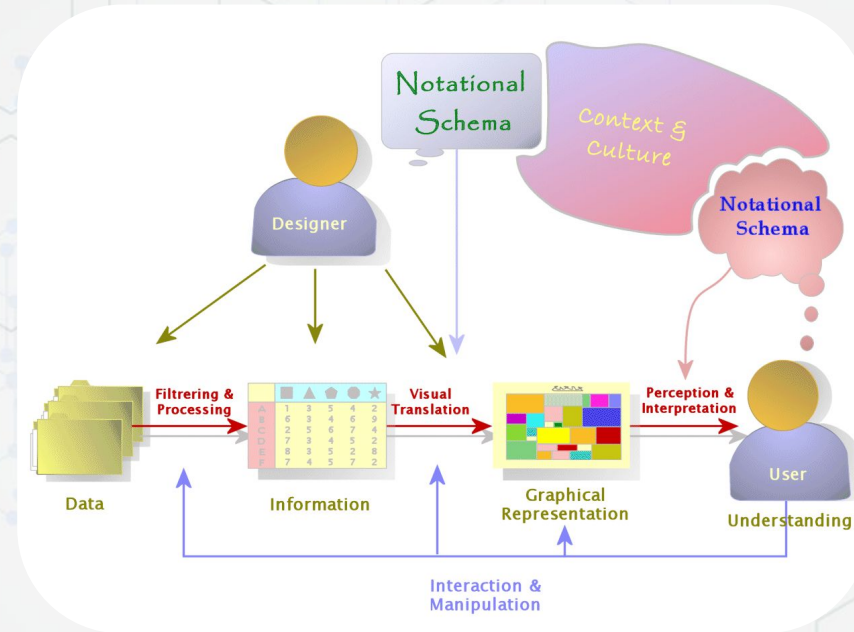
Descubrimiento de patrones

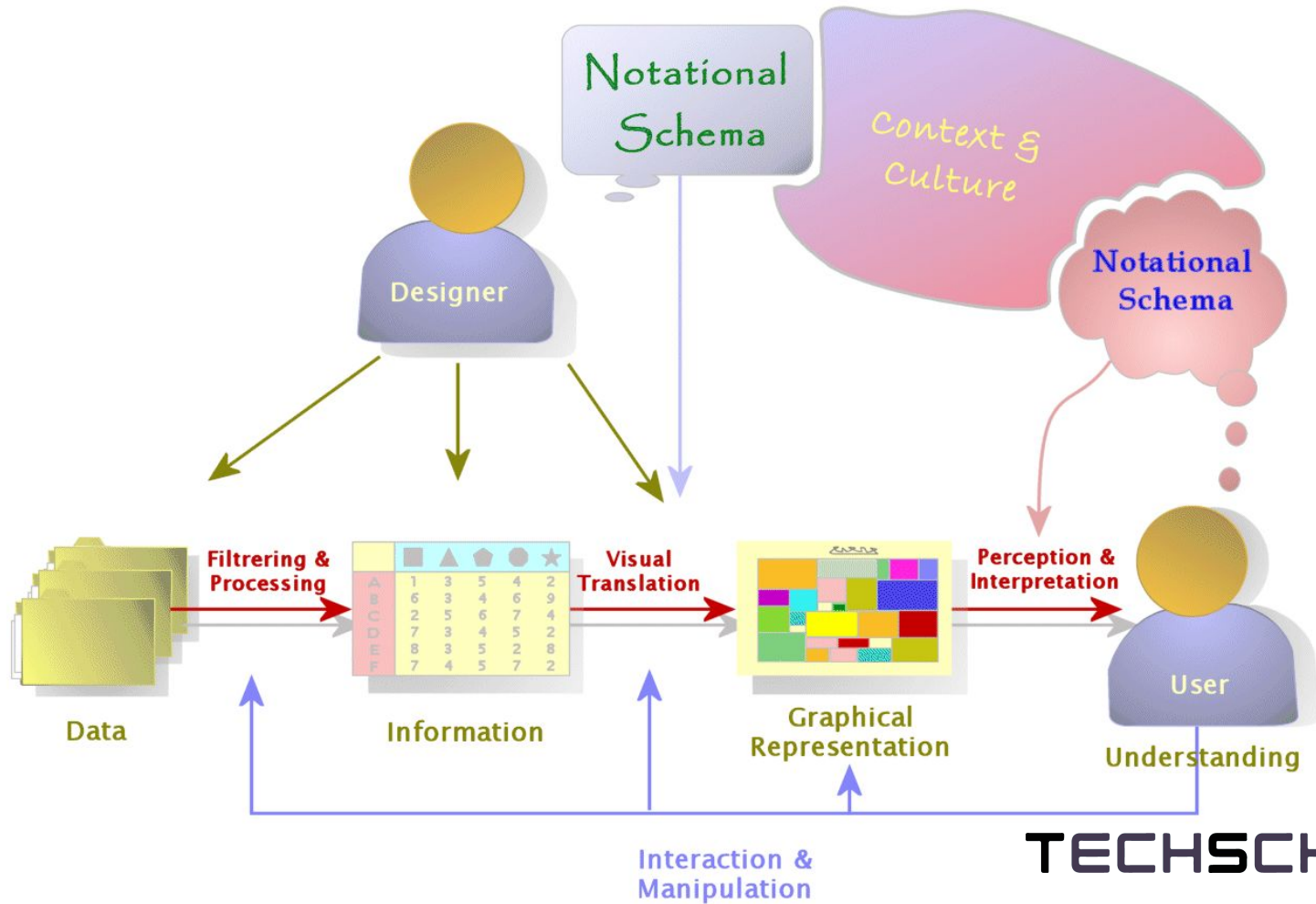
Facilita la toma de decisiones

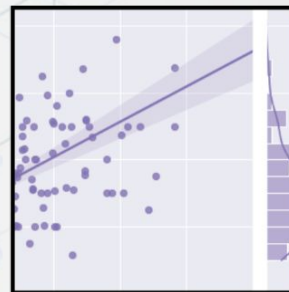
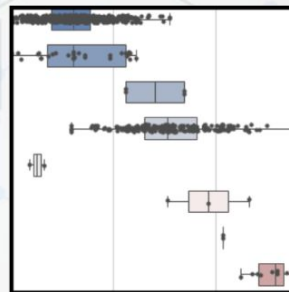
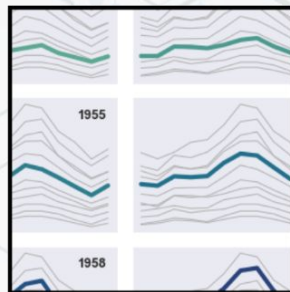
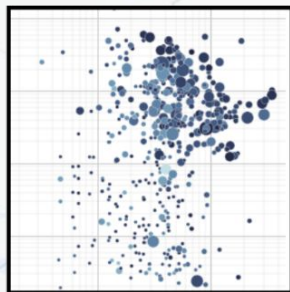
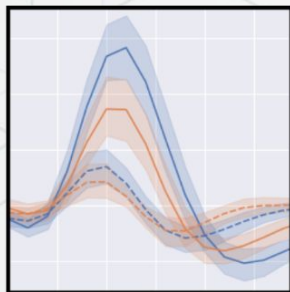
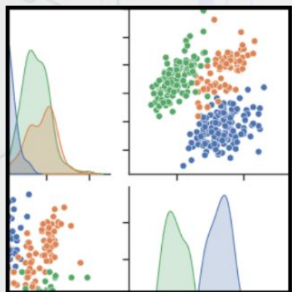
Comunicación efectiva

**TECHSCHOOL**

# ~ Diagrama InfoVis







**TECHSCHOOL**

The background of the slide features a light gray pattern of interconnected hexagons and lines, creating a network-like or molecular structure. The lines are thin and light blue, while the hexagons are slightly darker and more prominent.

~ Una efectiva visualización de datos es...

Simple

Con uso adecuado de colores

Consistente

**TECHSCHOOL**



## ~Tabla de datos original

	A	B
1	nombreColor	valorHexadec
2	rojo	#f00
3	verde	#0f0
4	azul	#00f
5	cyan	#0ff
6	magenta	#f0f
7	amarillo	#ff0
8	negro	#000



TECHSCHOOL



~ JSON

```
json
{
  "arrayColores": [
    {
      "nombreColor": "rojo",
      "valorHexadec": "#f00"
    },
    {
      "nombreColor": "verde",
      "valorHexadec": "#0f0"
    },
    {
      "nombreColor": "azul",
      "valorHexadec": "#00f"
    },
    {
      "nombreColor": "cyan",
      "valorHexadec": "#0ff"
    },
    {
      "nombreColor": "magenta",
      "valorHexadec": "#f0f"
    },
    {
      "nombreColor": "amarillo",
      "valorHexadec": "#ff0"
    },
    {
      "nombreColor": "negro",
      "valorHexadec": "#000"
    }
  ]
}
```

TECHSCHOOL

~ CSV

CSV

```
nombreColor,valorHexadec  
rojo,#f00  
verde,#0f0  
azul,#00f  
cyan,#0ff  
magenta,#f0f  
amarillo,#ff0  
negro,#000
```

TECHSCHOOL

# ~ XML

xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<arrayColores>
  <nombreColor>rojo</nombreColor>
  <valorHexadec>#f00</valorHexadec>
</arrayColores>
<arrayColores>
  <nombreColor>verde</nombreColor>
  <valorHexadec>#0f0</valorHexadec>
</arrayColores>
<arrayColores>
  <nombreColor>azul</nombreColor>
  <valorHexadec>#00f</valorHexadec>
</arrayColores>
<arrayColores>
  <nombreColor>cyan</nombreColor>
  <valorHexadec>#0ff</valorHexadec>
</arrayColores>
<arrayColores>
  <nombreColor>magenta</nombreColor>
  <valorHexadec>#f0f</valorHexadec>
</arrayColores>
<arrayColores>
  <nombreColor>amarillo</nombreColor>
  <valorHexadec>#ff0</valorHexadec>
</arrayColores>
<arrayColores>
  <nombreColor>negro</nombreColor>
  <valorHexadec>#000</valorHexadec>
</arrayColores>
```

TECHSCHOOL

# Introducción a Python

**TECHSCHOOL**



python<sup>TM</sup>

**TECHSCHOOL**

The background of the slide features a light gray pattern of interconnected hexagons and lines, creating a network-like or molecular structure.

# ~ Python

Inicialmente diseñado como una forma de escribir scripts que "automatizan las cosas aburridas"

**Líder en ciencia de datos, machine learning y gestión de infraestructura**

**TECHSCHOOL**



# ~ Características de Python

Lenguaje interpretado

Lenguaje de fácil lectura y escritura

Tipado dinámico

Orientado a los objetos

Amplio repertorio de librerías y frameworks



# ~ Sintaxis de Python

Python es conocido por su legibilidad y simplicidad, lo que hace que aprender y escribir código en este lenguaje sea más sencillo en comparación con muchos otros.

Python utiliza la **indentación** (espacios o tabulaciones) para definir bloques de código. La cantidad de espacio debe ser consistente dentro del mismo bloque.

# ~ Elementos básicos

Comentarios

Print

Variables

Tipos de datos

Operaciones matemáticas

Condicionales

Bucles

Funciones

Listas, tuplas,  
diccionarios

Importación de módulos

## ~ Comentarios

Los comentarios de una sola línea se inician con #.

Para comentarios multilínea, se pueden usar tres comillas dobles o simples.

```
# Esto es un comentario
```

```
"""
```

```
Esto es un comentario  
multilínea
```

```
"""
```

## ~ Print

Un print es una función que nos permite mostrar un mensaje en la pantalla. En este caso, el mensaje es 'Hola mundo'.

```
print('Hola mundo')
```

## ~ Variables

Una variable es un espacio en la memoria de la computadora donde podemos guardar un valor. En este caso, el valor es 'Hola mundo', y el nombre de la variable es mensaje.

```
mensaje = 'Hola mundo'  
print(mensaje)
```

# ~ Tipos de datos

**int:** números enteros

**float:** números decimales

**str:** texto

**bool:** booleano (True o False)

**list:** lista

**tuple:** tupla

**dict:** diccionario



# ~ Operaciones matemáticas

Los operadores son símbolos que nos permiten hacer operaciones matemáticas.

```
print(1 + 2)
print(1 - 2)
print(1 * 2)
print(1 / 2)
print(1 // 2)
print(1 % 2)
print(1 ** 2)
```

# ~ Condicionales

Los condicionales son instrucciones que nos permiten tomar decisiones.

```
a = 1
b = 2
if a == b:
    print('a es igual que b')
else:
    print('a es diferente de b')
```

# ~ Condicionales

Los condicionales son instrucciones que nos permiten tomar decisiones.

```
a = 1
b = 2
if a > b:
    print('a es mayor que b')
elif a < b:
    print('a es menor que b')
else:
    print('a es igual a b')
```

## ~ Bucles

Es una estructura de control que permite repetir un conjunto específico de instrucciones varias veces, según se cumpla una condición o hasta que una condición determinada se cumpla.

## ~ Bucle for

Este bucle se utiliza para iterar sobre una secuencia o para ejecutar un bloque de código un número determinado de veces.

```
for i in range(5):  
    print(i)
```

## ~ Bucle while

Permite ejecutar un conjunto de instrucciones mientras una condición sea verdadera.

```
contador = 0
while contador < 5:
    print(contador)
    contador += 1
```



## ~ Funciones

Una función es un bloque de código organizado y reutilizable que se utiliza para realizar una tarea determinada.

- Permiten la reutilización de código

- Facilitan la modularidad

- Hacen que el código sea más legible y mantenible

## ~ Funciones

Una función se define utilizando la palabra clave `def`. Se invoca la función por su nombre seguido de paréntesis, y pasando los argumentos necesarios.

```
def suma(a, b):  
    return a + b  
  
total = suma(5, 3)
```

# ~Listas, tuplas, diccionarios

**Listas:** Colecciones ordenadas y mutables.

**Tuplas:** Colecciones ordenadas e inmutables.

**Diccionarios:**  
Colecciones no ordenadas, mutables e indexadas. Compuestos por pares clave-valor.

```
lista = [1, 2, 3, 4]
```

```
tupla = (1, 2, 3, 4)
```

```
diccionario = {  
    "clave1": "valor1",  
    "clave2": "valor2"  
}
```

## ~ Importación de módulos

Un módulo es un archivo que contiene definiciones, como funciones, clases y variables, así como código ejecutable. La importación de módulos permite reutilizar código escrito por otros o por ti mismo en diferentes programas y scripts.

La capacidad de importar módulos es esencial para mantener el código organizado, modular y reutilizable.

## ~ Importación de módulos

```
import math  
from datetime import date
```

# Análisis de datos con Pandas y visualización con Altair

**TECHSCHOOL**



# ~ Librería Pandas



Pandas es una librería que proporciona estructuras de datos y herramientas de análisis de datos de alto rendimiento y fáciles de usar.

[pandas.pydata.org/](https://pandas.pydata.org/)

Lectura y escritura de datos en múltiples formatos (CSV, Excel, SQL, etc)  
Filtrado, selección y manipulación de datos.  
Gestión de datos faltantes.  
Fusionar y unir datos.  
Transformación de datos.  
Funcionalidades de análisis.

**TECHSCHOOL**

# ~ Librería Altair



Seaborn es una librería de visualización de datos. Está diseñada para crear visualizaciones atractivas y es útil para visualizar datos estadísticos.

[altair-viz.github.io/](https://altair-viz.github.io/)

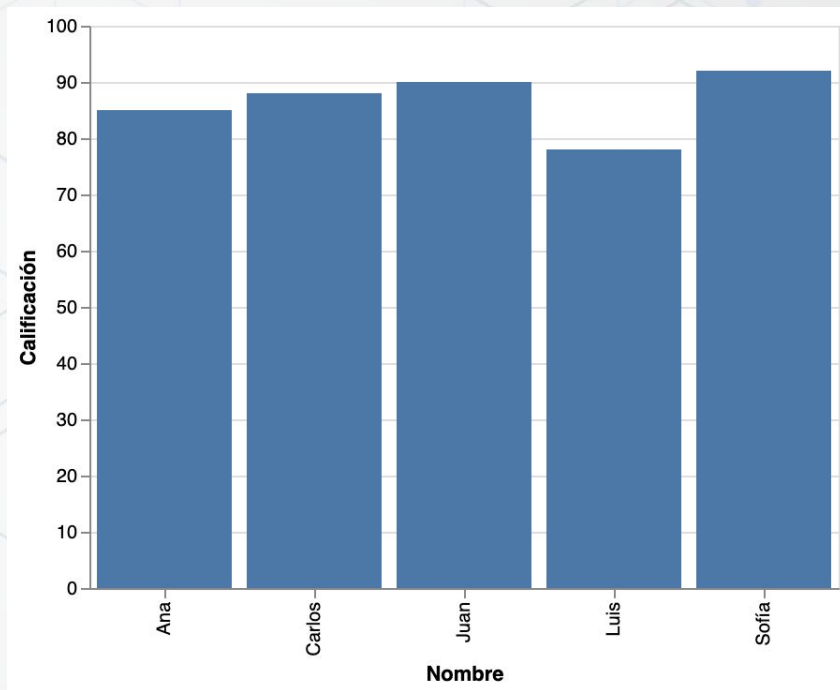
Está integrado con Pandas, lo que facilita la visualización directa de DataFrames. Incluye una amplia variedad de estilos y paletas de colores. De código simple y legible. Las visualizaciones se pueden exportar a HTML.

## ~ Ejemplo gráfico de barras

```
import pandas as pd
import altair as alt

data = {
    'Nombre': ['Ana', 'Juan', 'Luis',
               'Sofía', 'Carlos'],
    'Calificación': [85, 90, 78, 92, 88]
}
df = pd.DataFrame(data)

alt.Chart(df).mark_bar().encode(
    x='Nombre:O',
    y='Calificación:Q'
).properties(width=400)
```

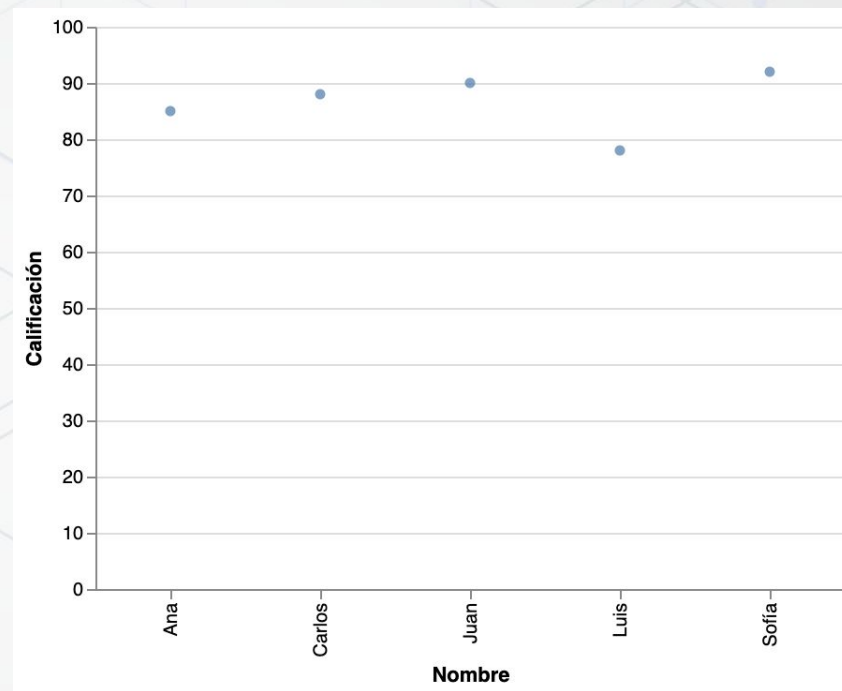


## ~ Ejemplo gráfico de puntos

```
import pandas as pd
import altair as alt

data = {
    'Nombre': ['Ana', 'Juan', 'Luis',
               'Sofía', 'Carlos'],
    'Calificación': [85, 90, 78, 92, 88]
}
df = pd.DataFrame(data)

alt.Chart(df).mark_circle().encode(
    x='Nombre:O',
    y='Calificación:Q'
).properties(width=400)
```

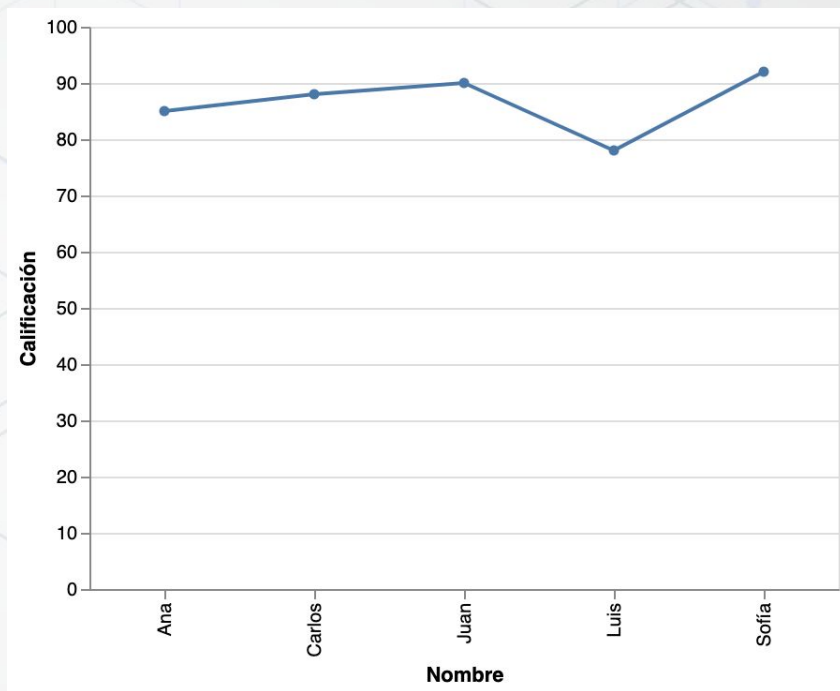


## ~ Ejemplo gráfico de línea

```
import pandas as pd
import altair as alt

data = {
    'Nombre': ['Ana', 'Juan', 'Luis',
              'Sofía', 'Carlos'],
    'Calificación': [85, 90, 78, 92, 88]
}
df = pd.DataFrame(data)

alt.Chart(df).mark_line(point=True).encode(
    x='Nombre:O',
    y='Calificación:Q'
).properties(width=400)
```

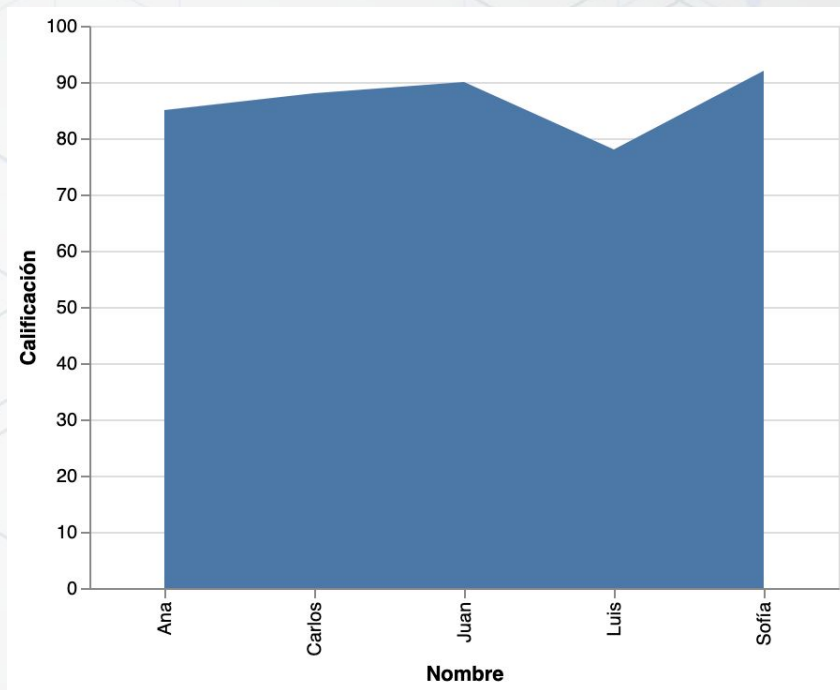


## ~ Ejemplo gráfico de área

```
import pandas as pd
import altair as alt

data = {
    'Nombre': ['Ana', 'Juan', 'Luis',
               'Sofía', 'Carlos'],
    'Calificación': [85, 90, 78, 92, 88]
}
df = pd.DataFrame(data)

alt.Chart(df).mark_area().encode(
    x='Nombre:O',
    y='Calificación:Q'
).properties(width=400)
```



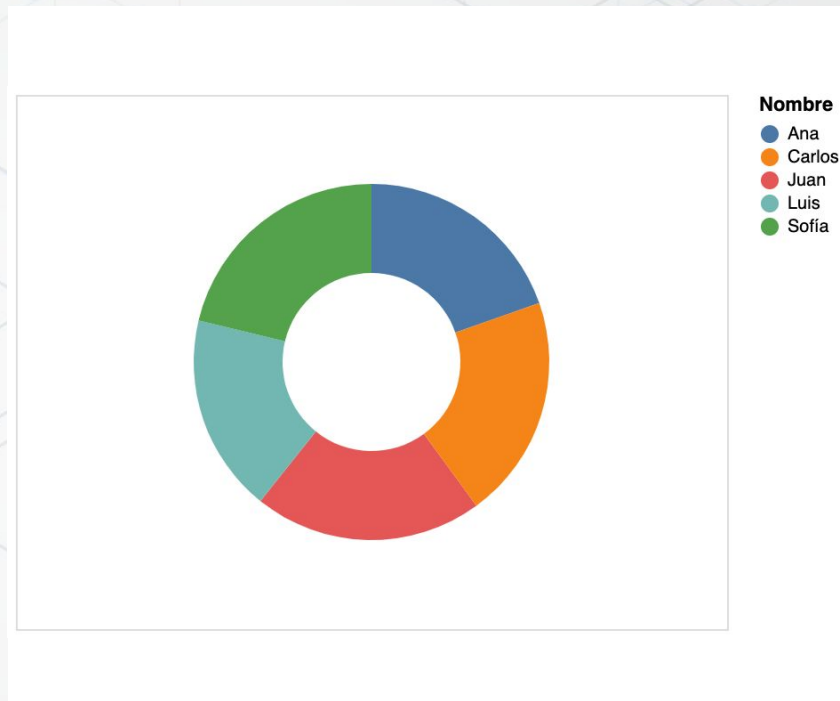


## ~ Ejemplo gráfico de dona

```
import pandas as pd
import altair as alt

data = {
    'Nombre': ['Ana', 'Juan', 'Luis',
               'Sofía', 'Carlos'],
    'Calificación': [85, 90, 78, 92, 88]
}
df = pd.DataFrame(data)

alt.Chart(df).mark_arc(innerRadius=50,
                        outerRadius=100).encode(
    theta='Calificación:Q',
    color='Nombre:N'
).properties(width=400)
```

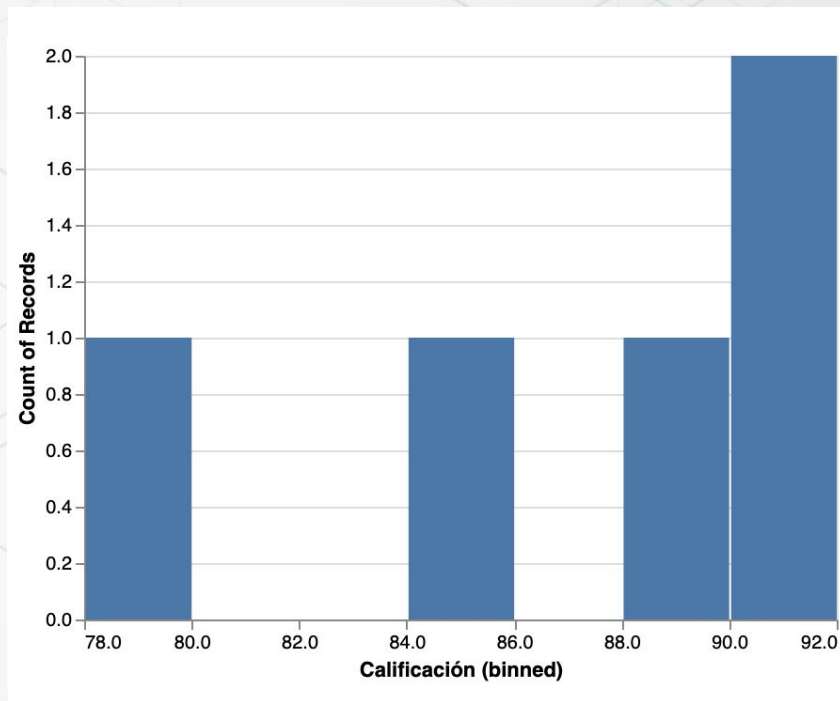


# ~ Ejemplo histograma

```
import pandas as pd
import altair as alt

data = {
    'Nombre': ['Ana', 'Juan', 'Luis',
              'Sofía', 'Carlos'],
    'Calificación': [85, 90, 78, 92, 88]
}
df = pd.DataFrame(data)

alt.Chart(df).mark_bar().encode(
    x=alt.X("Calificación:Q", bin=True),
    y='count()'
)
```



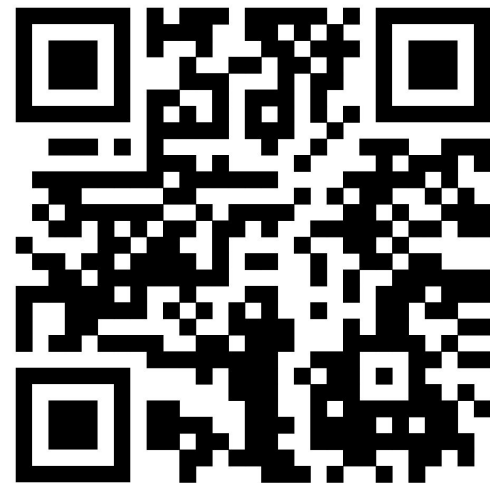
The background of the slide features a complex, repeating pattern of light blue hexagons and interconnected lines, creating a network-like or molecular structure. The lines are thin and connect various points, some of which are marked with small blue dots. The overall effect is a technical and modern aesthetic.

**¡Manos al código!**

**TECHSCHOOL**

## ~ Pasos iniciales

1. Abrir [Google Colab](#)
2. Click "New Notebook"
3. Activar Drive
4. Actualizar
5. Subir CSV a la carpeta Colab Notebooks



[github.com/JSConfCL/techschool/tree/main/ProgramacionPython](https://github.com/JSConfCL/techschool/tree/main/ProgramacionPython)

**TECHSCHOOL**