

## FORMATEO DE RESULTADOS DE CONSULTAS DE BASES DE DATOS.

Hay varias formas de mostrar los resultados de una consulta.

**Opción 1: Usar `join` y `map` para mostrar cada fila como texto:**

```
for row in registros:  
    print("|".join(map(str, row)))
```

`map` convierte cada elemento de `row` en un `str` y los une usando `join` con el separador que le indiquemos (`|` en este caso).

```
12|Crispulo|Delgado|55|c.delgado@usal.es|Salamanca|None  
7|David|Ram  rez|52|david.ramirez@example.com|Salamanca|None
```

Esta opci  n es bastante r  pida y legible.

**Opci  n 2: Formatear con f-strings:**

Sabiendo la estructura de la tabla (es decir, los nombres de las columnas, por ejemplo: `id`, `nombre`, `apellido`, `edad`, `email`), se puede hacer algo m  s descriptivo pero m  s trabajoso usando las f-strings:

```
for row in registros:  
    id_cliente, nombre, apellido, edad, email, ciudad, fecha = row  
    print(f"ID: {id_cliente}, Nombre: {nombre} {apellido}, Edad: {edad}, Email: {email}, Ciudad: {ciudad}, Fecha de registro: {fecha} ")
```

El resultado es:

```
12|Crispulo|Delgado|55|c.delgado@usal.es|Salamanca|None  
7|David|Ram  rez|52|david.ramirez@example.com|Salamanca|None
```

**Opci  n 3: Mostrar en forma de tabla con `tabulate`**

Previamente tenemos que importar el m  dulo en el archivo (es posible que requiera instalaci  n con `pip`):

```
from tabulate import tabulate
```

`Tabulate` crea una tabla con los resultados que le proporcionemos, con cabecera (si lo indicamos as  ) y con la posibilidad de usar distintos tipos de tabla.

Ejemplo:

```
print(tabulate(registros, headers=["Nombre", "Apellido", "Email"], tablefmt="grid"))
```

El resultado es:

```
+-----+-----+-----+  
| Nombre | Apellido | Email |  
+-----+-----+-----+  
| Crispulo | Delgado | c.delgado@usal.es |  
+-----+-----+-----+  
| David | Ram  rez | david.ramirez@example.com |  
+-----+-----+-----+
```

Le da un aspecto m  s profesional.

Otras opciones de `tablefmt` (formato de tabla):

## Formatos principales de `tabulate`

- `"plain"` → sin bordes, solo texto separado por espacios.
- `"simple"` → líneas mínimas, estilo muy limpio.
- `"grid"` → con líneas horizontales y verticales (el que ya viste).
- `"fancy_grid"` → igual que `"grid"` pero con bordes más estilizados.
- `"pipe"` → estilo Markdown con barras verticales (`|`).
- `"orgtbl"` → estilo de tablas de Emacs/Org-mode.
- `"github"` → estilo Markdown usado en GitHub.
- `"psql"` → similar a la salida de PostgreSQL.
- `"rst"` → tablas en formato reStructuredText.
- `"mediawiki"` → formato para wikis tipo MediaWiki.
- `"html"` → genera una tabla en HTML.
- `"latex"` → genera una tabla en LaTeX.
- `"latex_booktabs"` → tabla LaTeX con estilo de libro (más elegante).
- `"tsv"` → valores separados por tabuladores.