

DATA MANAGEMENT

SQL Segmento 3

Índice

1. SUBQUERIES

2. UNIONS

a. Union

b. Union All

3. JOINS

a. Left Join

b. Inner Join

c. Full Join

d. Right Join

4. PRÁCTICA

5. NO CODE



Data Management

Subqueries



Subqueries

Introducción a Subqueries

Una *subquery* es una estructura **SELECT FROM** independiente cuyo output se almacena temporalmente en memoria para poder operar como si fuera una tabla disponible en la base de datos.

Para implementar una *subquery*, utilizamos el **WITH** statement con la siguiente sintaxis:

WITH

temp_table_1 AS (SELECT... FROM...),

temp_table_2 AS (SELECT... FROM...)

SELECT... FROM temp_table_1

Subqueries

Ejemplo

SQL Query

```
1 WITH my_subquery AS(SELECT Sector, SUM(Revenue) AS total_revenue
2 FROM fortune
3 GROUP BY 1)
4 SELECT AVG(total_revenue) AS avg_sector_revenue FROM my_subquery
5
```

Output

avg_sector_revenue

876000

Data Management

Unions



Unions

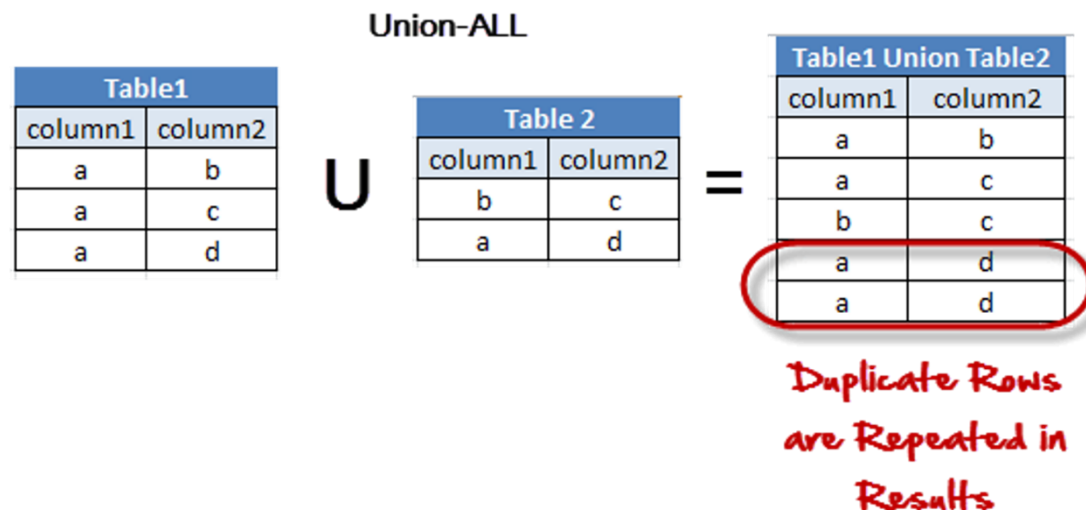
UNION (UNION ALL)

Una UNION combina dos tablas de forma **vertical** (uniendo filas). UNION ALL incluye todas las filas, sin importar si son duplicados.

```
SELECT column1, column2 FROM table1
```

```
UNION ALL
```

```
SELECT column1, column2 FROM table2
```



Unions

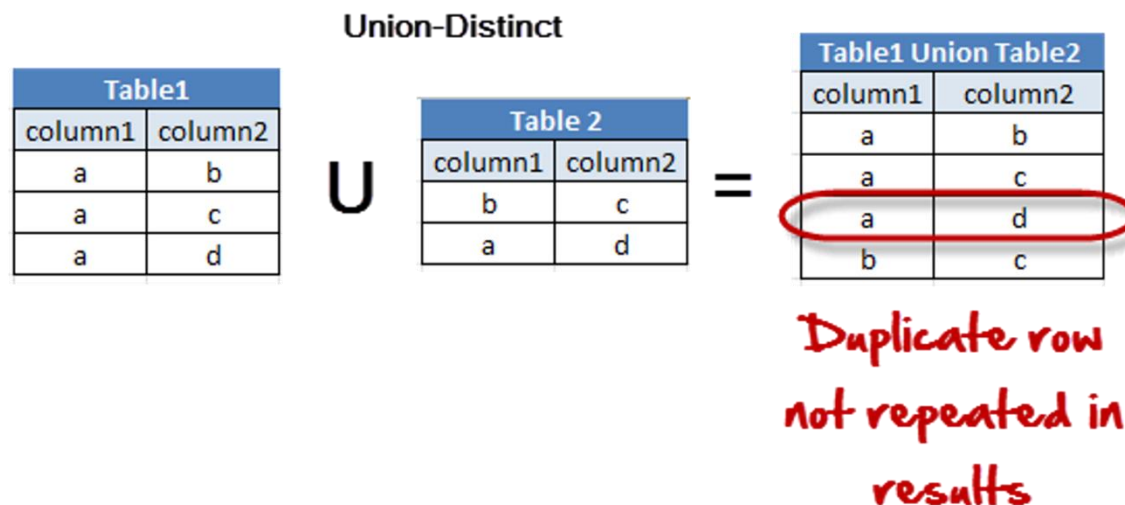
UNION (DISTINCT)

Una UNION combina dos tablas de forma **vertical** (uniendo filas). UNION (DISTINCT) excluye duplicados del resultado final.

```
SELECT column1, column2 FROM table1
```

```
UNION
```

```
SELECT column1, column2 FROM table2
```



Unions

PRACTICA

Preparar dos tablas en Excel con formato .csv, añadir datos en 2-3 filas y comprobar los dos tipos de UNIONS

- UNION ALL
- UNION

Data Management

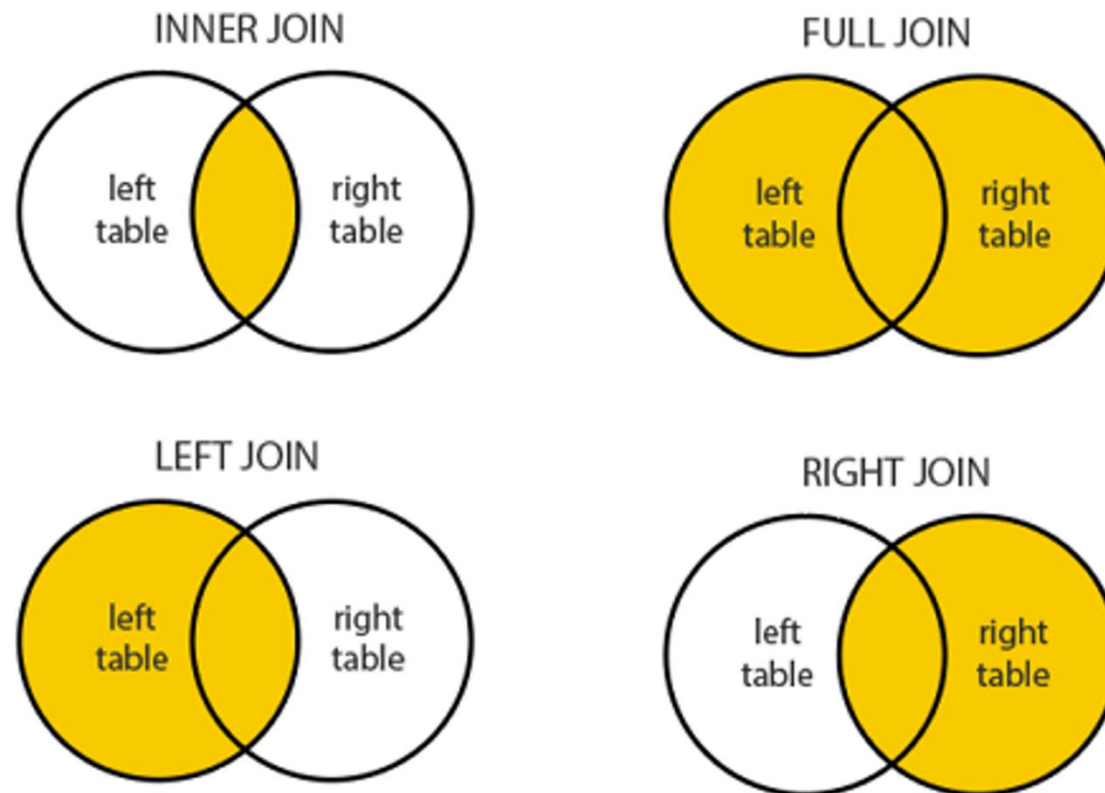
Joins



Joins

Introducción a JOINS

Una *join* es combinar dos tablas de datos de forma **horizontal** (juntando columnas). Existen distintas formas de realizar *joins*, estas difieren en el tratamiento de las observaciones que no coinciden.



Joins

INNER JOIN

Una *join* de tipo INNER mantiene solo las observaciones que coinciden en ambas tablas (LEFT y RIGHT).

```
SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate  
FROM Orders
```

```
INNER JOIN Customers ON Orders.CustomerID=Customers.CustomerID;
```

```
SELECT T1.OrderID, T2.CustomerName, T1.OrderDate  
FROM Orders T1
```

```
INNER JOIN Customers T2 ON T1.CustomerID=T2.CustomerID;
```

Joins

LEFT JOIN

Una *join* de tipo LEFT mantiene todas las observaciones de la tabla principal (LEFT) e integra las observaciones que coinciden de la tabla secundaria (RIGHT).

```
SELECT Customers.CustomerName, Orders.OrderID  
  
FROM Customers  
  
LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID  
  
ORDER BY Customers.CustomerName;
```

Joins

RIGHT JOIN

Una *join* de tipo RIGHT es el opuesto a una LEFT join. Mantiene todas las observaciones de la tabla secundaria (RIGHT) e integra aquellas observaciones de la tabla principal (LEFT) cuyos valores coincidan.

```
SELECT Customers.CustomerName, Orders.OrderID  
  
FROM Customers  
  
RIGHT JOIN Orders ON Customers.CustomerID = Orders.CustomerID  
  
ORDER BY Customers.CustomerName;
```

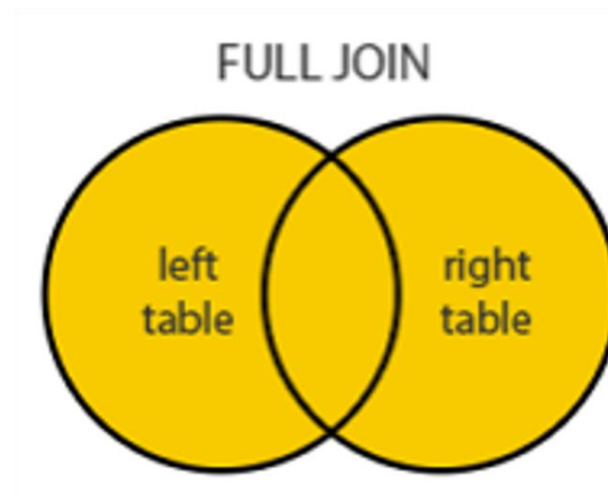
Joins

FULL JOIN

Una *join* de tipo FULL mantiene todas las observaciones de ambas tablas (LEFT y RIGHT), sin importar si coinciden o no.

```
SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate  
FROM Orders
```

```
FULL JOIN Customers ON Orders.CustomerID=Customers.CustomerID;
```



Ejemplos de INNER JOIN

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

ORDERS

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

CUSTOMERS

```
SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate
```

```
FROM Orders
```

```
INNER JOIN Customers ON Orders.CustomerID=Customers.CustomerID;
```


Ejemplos de INNER JOIN

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

ORDERS

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

CUSTOMERS

Ejemplos de LEFT JOIN

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

ORDERS

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

CUSTOMERS

```
SELECT Customers.CustomerName, Orders.OrderID
```

```
FROM Customers
```

```
LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID
```

```
ORDER BY Customers.CustomerName;
```

Ejemplos de LEFT JOIN

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

ORDERS

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

CUSTOMERS

Ejemplos de RIGHT JOIN

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

ORDERS

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

CUSTOMERS

```
SELECT Customers.CustomerName, Orders.OrderID
```

```
FROM Customers
```

```
RIGHT JOIN Orders ON Customers.CustomerID = Orders.CustomerID
```

```
ORDER BY Customers.CustomerName;
```

Ejemplos de RIGHT JOIN

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

ORDERS

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

CUSTOMERS

Ejemplos de FULL JOIN

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

ORDERS

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

CUSTOMERS

```
SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate
```

```
FROM Orders
```

```
FULL JOIN Customers ON Orders.CustomerID=Customers.CustomerID;
```

Ejemplos de FULL JOIN

OrderID	CustomerID	OrderDate
10308	2	1996-09-18
10309	37	1996-09-19
10310	77	1996-09-20

ORDERS

CustomerID	CustomerName	ContactName	Country
1	Alfreds Futterkiste	Maria Anders	Germany
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Mexico
3	Antonio Moreno Taquería	Antonio Moreno	Mexico

CUSTOMERS

Data Management

Práctica



Práctica

Ex.1: Descripción

Los directores de películas de James Bond han trabajado en promedio en dos películas cada uno. ¿Cómo obtendrías esta información?

Tabla: james_bond

Resultado esperado

Row	DIR_AVG_MOVIES
1	2.0

Práctica

Ex.1.1: Descripción

En la tabla de google analytic indica para el mes de septiembre, que canal ha sido el más utilizado en cada día (hay 10 registrados en la base de datos)

Tabla: Google_analytic

Resultado esperado

date	mes	canal	max_sesiones
2019-09-21	09	Social	13
2019-09-22	09	Referral	9
2019-09-23	09	Referral	13
2019-09-24	09	Social	14
2019-09-25	09	Social	20
2019-09-26	09	Social	15
2019-09-27	09	Social	10

Práctica

Ex.2: Descripción

La tabla gobierno_paro reporta data mensual de paro para *municipios* de España en 2018. Calcular la media de paro de cada *municipio* en 2018.

Tabla: gobierno_paro

Resultado esperado

Row	comunidad_autonoma	avg_unemployment
1	Andalucia	809843.0
2	Cataluna	392992.0
3	Comunitat Valenciana	375974.0
4	Madrid, Comunidad de	360194.0
5	Canarias	210249.0

Práctica

Ex.3: Descripción

Extraer consolas no discontinuadas y su fecha de lanzamiento (*first retail availability*). Agregar ventas globales de videojuegos publicados desde el año 2000. Ordenar según ventas totales de videojuegos (descendiente).

- *Table 1: videogames_games*
- *Table 2: videogames_consoles*

Resultado esperado

Row	console_name	first_retail_availability	global_games_sales
1	Playstation 3	2006-11-11	945.0
2	Nintendo 3DS	2011-02-26	240.0
3	Platstation 4	2013-11-15	239.0
4	Xbox One	2013-11-22	129.0
5	Nintendo WiiU	2012-11-18	79.0
6	Play Station Vita	2011-12-17	58.0

Challenge??



BUSINESS CASE

Analizando Productos de Amazon Consumer Electronics

Eres un Analista de Datos en Amazon.com (AMZN). El equipo de Consumer Electronics (CE) te ha solicitado que los ayudes a entender cómo es el posicionamiento de AMZN vs. otros retailers.

Detalles

Data disponible publicamente (*scrapped*) en 2017/2018 para productos en Amazon.com

- Table 1: *product_data*; contiene atributos de productos
- Table 2: *amazon*; incluye precios de productos para Amazon
- Table 3: *competitors*; contiene precios de competidores

Supuestos,

- la tabla *product_data* contiene **todos** los productos de CE en Amazon.com
- la tabla *competitors* contiene **todos** los competidores

Recomendaciones,

- utilizar *number_of_reviews* como variable proxy para ventas
- utilizar tantas queries como sean necesarias

Enviar a albertoarranz77@gmail.com

- SQL relevante utilizado para cada ejercicio

Data Management

No-code



No-code

¿Que es el Nocode?

Las plataformas de desarrollo sin código (nocode) permiten a los programadores y no programadores crear software de aplicación a través de interfaces gráficas de usuario y configuración en lugar de la programación informática tradicional.

No-code

¿Que es el Nocode?

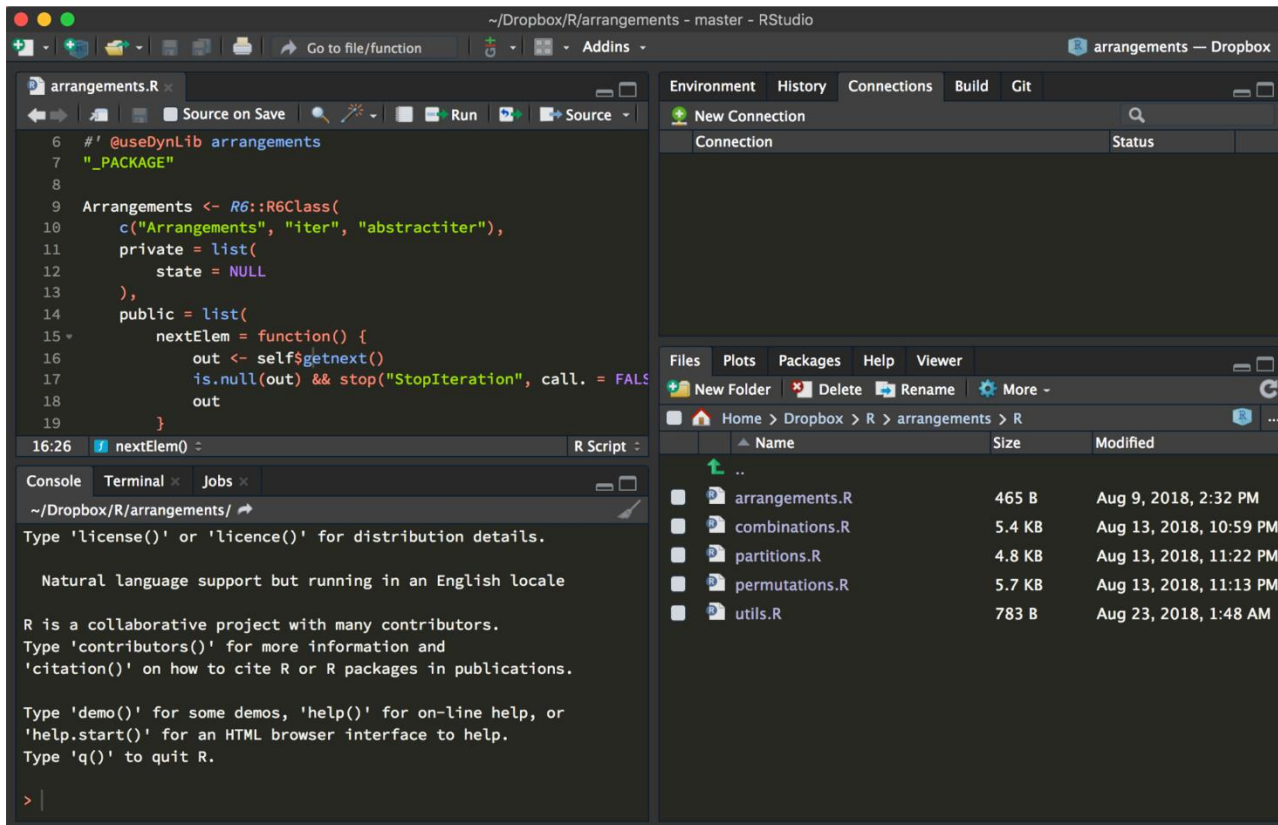
El movimiento no code es una filosofía digital cuyo objetivo es permitir a cualquier persona acceder a la **creación de apps** y softwares **sin saber programación**.

Es lo que se conoce como **programación sin código o programación visual**. Es decir, en prescindir de lenguajes de codificación para que la creación sea un proceso más visual y sencillo.

Para poder crear una aplicación sin programar se utilizan **diferentes herramientas de desarrollo** que ofrecen un **entorno visual**.

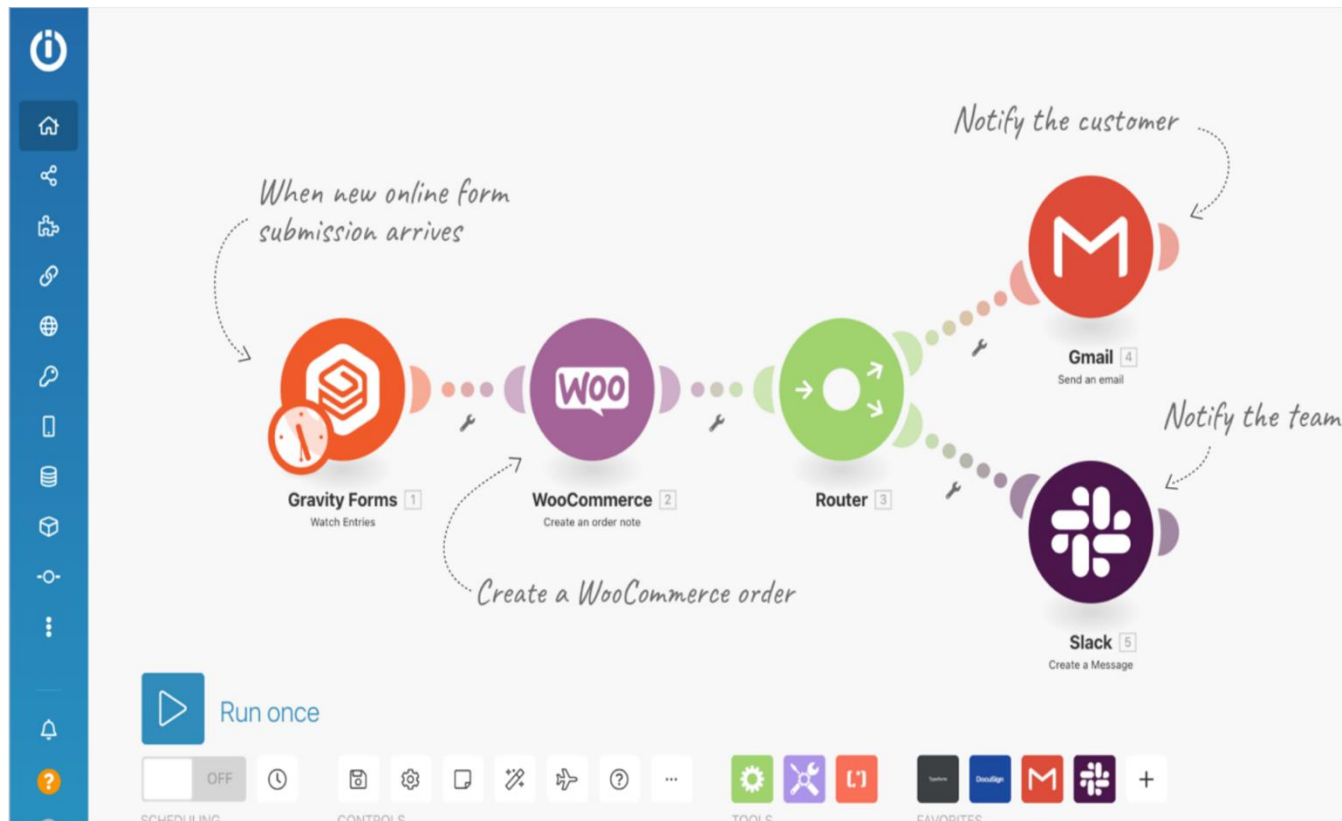
Así los usuarios solo tienen que implementar métodos sencillos para crear su app o web, como por ejemplo agregar componentes con la técnica de arrastrar y soltar (también conocida como **drag&drop**).

No-code



**EL NO CODE ES UNA
“ABSTRACCION”**

No-code



SUSTITUIMOS LINEAS DE CODIGO POR UN EDITOR VISUAL

No-code

¿Qué ventajas ofrece el nocode?

Permite que el desarrollo de software sea accesible a un público que no contaba con los conocimientos para hacerlo.

Pero, ¿programar sin código de verdad funciona en la realidad?

Si, el crecimiento de las herramientas no-code así lo demuestran.

De acuerdo con datos de Forrester, el mercado de plataformas de desarrollo sin código crecerá a **21.2** mil millones de dólares en 2022.

Para entender bien su crecimiento, debes considerar que en 2017 apenas representaban **3.8** mil millones.

No-code

Beneficios que existen el hacer uso de aplicaciones no-code

Agilidad:

Empresas, emprendedores y usuarios particulares están apostando por las no-code tools (herramientas sin código) debido a que han entendido que con ellas es más rápido crear aplicaciones web y móviles.

Al utilizar módulos prediseñados y elementos visuales, la creación de aplicaciones es más rápida, lo que permite una mayor agilidad y reducir el tiempo que se dedica a estas tareas.

Curva de aprendizaje:

Gracias a la facilidad de uso permitimos a más gente de nuestra organización el uso de herramientas de no code.

No-code

Beneficios que existen el hacer uso de aplicaciones no-code

Menor coste:

Otra de las ventajas de poder crear aplicaciones web sin programar es que resulta más rentable.

Contratar desarrolladores para iniciar un proyecto de programación puede ser costoso. Pero con las soluciones no-code la inversión es significativamente menor.

No tendrás que gastar demasiado contratando un equipo de desarrolladores y además, al ser más rápido el proceso de desarrollo, también se ahorran gastos.

No-code

Beneficios que existen el hacer uso de aplicaciones no-code

Modificaciones fáciles y rápidas:

Cambiar una característica o funcionalidad de un software o app programando es complejo. Este procedimiento lleva tiempo y mucho esfuerzo, sobre todo cuando el lenguaje de programación es poco conocido.

Esto no ocurre haciendo uso de herramientas no-code, pues en el desarrollo sin código si necesitas cambiar algo, puedes hacerlo de forma sencilla y su implementación es mucho más rápida.

No-code

Desventajas del no-code

PERSONALIZACIÓN

Al basar el desarrollo en herramientas predefinidas

“CAUTIVIDAD”

Una vez elegimos una plataforma de no code estamos obligados a utilizar su stack, haciendo difícil la migración a otro sistema

SEGURIDAD

Las herramientas no code viven en la nube y tienen sus propios protocolos de seguridad, ajenos a los propios de nuestra compañía

No-code

Herramientas Populares de No-Code

webflow

zapier



 **Airtable**

glide

 **make**
formerly Integromat



red.es



UNIÓN EUROPEA

"El FSE invierte en tu futuro"

Fondo Social Europeo

