

# Base de datos desde cero

Clase 9 - 09/11/2022

## ¿Qué podemos hacer con lo aprendido hasta ahora?

Hasta ahora hemos aprendido lo necesario para mostrar columnas (Select) y filas de una fuente de datos. Como fuente de datos (From) utilizamos una Tabla, más de una tabla unidas por Inner Join, Left Join, Right Join y sub consultas. A dicho origen de datos aprendimos a aplicarle filtros (Where) a los efectos de obtener la información realmente importante para nosotros, de esta manera la consulta se hace más eficiente.

También aprendimos a agrupar (Group By) las filas tomando como criterio de agrupamiento un campo o más, y a su vez utilizar alguna de las funciones agregadas para obtener información sobre dichos grupos (Count, Max, Avg, etc).

Todo esto lo podemos ordenar por uno o más campos.

Lo expuesto lo realizamos en una sola consulta, hasta ahora no tuvimos la necesidad de generar una consulta tras otra para resolver los problemas planteados.

## Sentencia If en MySql

If dentro de las consultas Sql

If es una estructura de control de flujo muy utilizada por los lenguajes de programación, MySql armó su propio If para las consultas Sql.

Ejemplo:

```
select alu.nombreyapellido,  
       if(alu.idlocalidad is null, 'Sin asignar', loc.localidad) localidad  
from Alumno alu left join Localidad loc  
    on alu.idlocalidad = loc.idlocalidad  
order by alu.nombreyapellido;
```

## Sentencia Case When Else

El Case When Else, nos permite en base a distintas evaluaciones realizadas, tomar un camino u otro. Esta sentencia existe en la mayoría de los motores de Base de Datos.

Ejemplo:

```
select alu.nombreapellido,  
       case When alu.idlocalidad is null then 'Sin asignar'  
       else  
           loc.localidad end  localidad  
from Alumno alu left join Localidad loc  
    on alu.idlocalidad = loc.idlocalidad  
order by alu.nombreapellido;
```

## Sentencia Case When Else

Más comparaciones, ejemplos:

```
select alu.nombreyapellido,  
       case  
         When alu.idlocalidad is null then 'Sin asignar'  
         When alu.idlocalidad = 3 then 'Capital'  
         When alu.idlocalidad = 1 then 'Capital del Monte'  
         else  
           loc.localidad end localidad  
from Alumno alu left join Localidad loc  
  on alu.idlocalidad = loc.idlocalidad  
order by alu.nombreyapellido;
```

## Sentencia Case When Else

Complicando un poco más:

```
select
  sum(case
    when alu.IdLocalidad = 3 then 1 else 0 end) Capital,
  sum(case
    when alu.idLocalidad > 0 and alu.idlocalidad <> 3 then 1 else 0 end) Interior
from Alumno alu left join Localidad loc
  on alu.idlocalidad = loc.idlocalidad
order by alu.nombreyapellido;
```

# Introducción Básica a la Programación

¿Cuál es el concepto de programación?

La **programación** informática es el arte del proceso por el cual se **limpia**, codifica, traza y protege el código fuente de programas computacionales, en otras palabras, es indicarle a la computadora lo **que** tiene **que** hacer.

Fuente: <https://www.netec.com/que-es-programacion>

¿Qué es la programación Wikipedia?

En informática la **programación** es un proceso por el cual se escribe (en un lenguaje de **programación**), se prueba, se depura y se mantiene el código fuente de un programa informático.

Fuente: <https://es.wikipedia.org/wiki/Portal:Programaci%C3%B3n>




# Introducción Básica a la Programación

## ¿Qué es programar en equipos tecnológicos?

Creo que podemos simplificar la definición diciendo que: es un conjunto de pasos ordenados, que lo escribimos en un determinado lenguaje de programación, el cual de alguna forma es entendido por el equipo tecnológico, para ejecutarlos. Dichos pasos tienen como objetivo resolver algún tipo de problema.

## ¿Qué sería una lista de pasos?

- 
- Agregar a un bol 200 grs de harina
  - Agregar dos huevos
  - Mezclar hasta formar una masa suave
  - Cubrir la masa con un repasador y dejar descansar 20 minutos.
  - Estirar la masa
  - Cortar en rectángulos de 15 cms por 25 cms.

**Objetivo:** Masa para lasaña.

**Secuencial**



# Introducción Básica a la Programación

## Variables:

- Es un espacio de memoria que se reserva para que durante la ejecución de nuestro programa podamos utilizarla para guardar determinados valores.
- Dichos valores son utilizados durante la ejecución del programa para realizar cálculos, para almacenar temporalmente algunos valores, etc.
- Una variable normalmente acepta un tipo de datos.

Por ejemplo: Definición

Declare:

```
variable_A Integer;
variable_B Integer;
variable_C Integer;
variable_C Varchar(50);
```

Ejemplo de uso de variables:

```
variable_A = 5;
variable_B = 5;
variable_C = variable_A +
variable_B;
```

Ejemplo en MySQL:

```
declare A Integer;
set variable_A = 5;
```

# Introducción Básica a la Programación

## Contadores y Acumuladores

### Contadores:

Nos permiten contar cuántas veces pasó algo, o bien cuantas veces se ejecutó un determinado número de procesos.

Ejemplo:

Set  $A = A + 1$ ; – Se cuenta de a 1.

Set  $A = A - 1$ ; – Se puede contar para atrás.

Set  $A = A + 3$ ; – Se puede contar, no solo de a uno, podemos contar por cualquier número CONSTANTE-

Set  $A = A - 3$ ;

Observar que el valor de A se incrementa o decrementa un número constante. A esto le llamamos contador.

# Introducción Básica a la Programación

## Contadores y Acumuladores

### Acumuladores:

Nos permiten acumular en una variable un valor variable. Como ejemplo podemos decir que nos serviría para determinar cuanto es el total de una factura, acumularíamos todos los importes del detalle de la factura, uno a uno

Ejemplo: (recorremos el total de la factura)

Set TotFac = TotFac + vTotalDetalle; -vTotalDetalle es el importe del detalle..

También podemos usarlo para ir decrementando un valor de forma variable.

Set TotSaldo = TotSaldo - vTotalDetalle; vTotalDetalle es el importe del detalle..

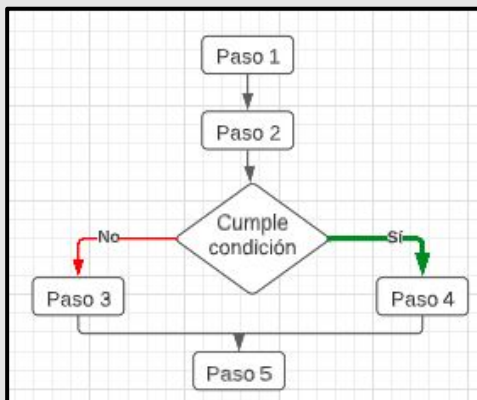
Observar que el valor de TotFac se incrementa y el de Saldo se decrementa un número variable. A esto lo llamamos acumulador.

# Introducción Básica a la Programación

## Estructura de control Selectiva

Agregando algunas características a la programación, podemos decir que no siempre es un paso seguido del otro, si no que tenemos también estructuras de control que nos permiten ir por un camino u otro.

### If then else



### en MySql

```

BEGIN
  declare vMensaje varchar(50);
  if pDni > 500 then
    set vMensaje = 'Tu dni es alto';
  else
    set vMensaje = 'Tu dni es bajo';
  end if;
  return vMensaje;
END
  
```

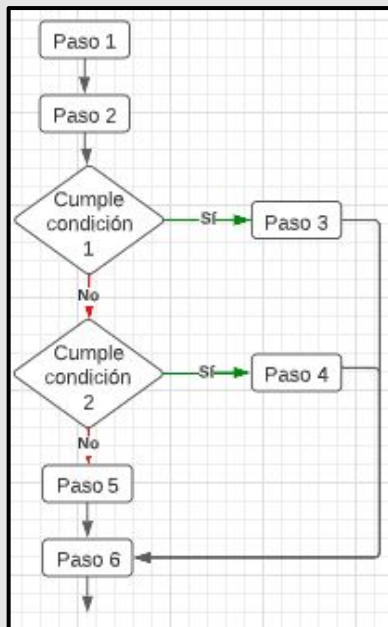
### en Oracle

```

vMensaje varchar(50);
begin
  if pDni > 500 then
    vMensaje := 'Tu dni es alto';
  else
    vMensaje := 'Tu dni es bajo';
  end if;
  return vMensaje;
end;
  
```

# Introducción Básica a la Programación

## Estructura de control Selectiva múltiples



en MySql

```

declare vMensaje varchar(50);
if pDni > 500 then
    set vMensaje = 'Tu dni es alto';
elseif pDni > 200 then
    set vMensaje = 'Tu dni es medio';
else
    set vMensaje = 'Tu dni es bajo';
end if;
return vMensaje;
  
```

en Oracle

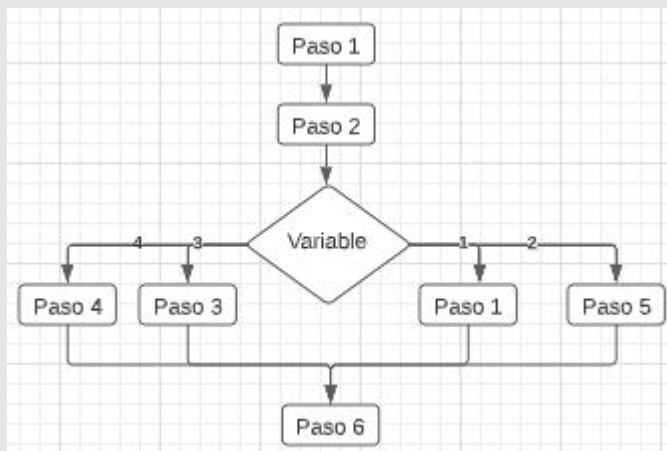
```

if pDni > 500 then
    vMensaje := 'Tu dni es alto';
elsif pDni > 200 then
    vMensaje := 'Tu dni es medio';
else
    vMensaje := 'Tu dni es bajo';
end if;
return vMensaje;
  
```

# Introducción Básica a la Programación

## Estructura de control Selectiva múltiples (case, switch, etc)

case / switch



en Oracle

```

Case vMes
  when 1 then vMensaje := 'Enero';
  when 2 then vMensaje := 'Febrero';
  else vMensaje := 'Mes no válido';
end case;
return vMensaje;
  
```

# ¡Muchas gracias!

    /poloticmisiones