

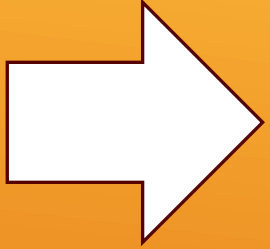
CURSOR SQL

Es una estructura que se carga a memoria ram con el resultado de una consulta, permitiéndonos recorrer fila por fila de esa estructura, dejándonos leer la información y eventualmente modificar dicho conjunto de datos.



PARALELISMO CON QUIASMA

- _(a)Y acontecerá que quien no tome sobre si el NOMBRE de Cristo
- __(b) Tendrá que SER LLAMADO por algún otro nombre
- ___(c) Por tanto, se hallara a la IZQUIERDA DE DIOS
- ____(d) Y quisiera que RECORDASEIS que este es el NOMBRE
- _____(e) Que dije que os daría, el cual NUNCA SERIA BORRADO
- _____(f) Sino por TRANSGRESION
- _____(f') Por tanto, tened cuidado de no TRANSGREDIR
- _____(e') Para que el nombre NO SEA BORRADO de vuestros corazones
- _____(d') Yo os digo: quisiera que os ACORDASEIS de conservar siempre escrito ese NOMBRE
- ___(c') En vuestros corazones para que no os halléis a la IZQUIERDA DE DIOS
- __(b') Sino que le oigáis y conozcáis la voz por la cual SEREIS LLAMADOS
- _(a')Y también el NOMBRE por el cual el os llamara.



CICLO DE VIDA DE CURSOR(BASICO)

PARAMETROS CURSOR

**CICLO DE VIDA DE
CURSOR(AVANSADO)**

1.-CREAR CURSOR

Declaramos el cursor y especificamos con que consulta quiero que haga la estructura con la que navegara en la ram

Declare – instrucción usada para crear variables y cursores

Cursor – Indicador de tipo de instrucción cursor

For – Asignacion de query para el cursor

```
Declare Cursor_1 Cursor
for SELECT Correo,NomCompleto from UsuariosCotizan
```

100 %

Resultados Mensajes

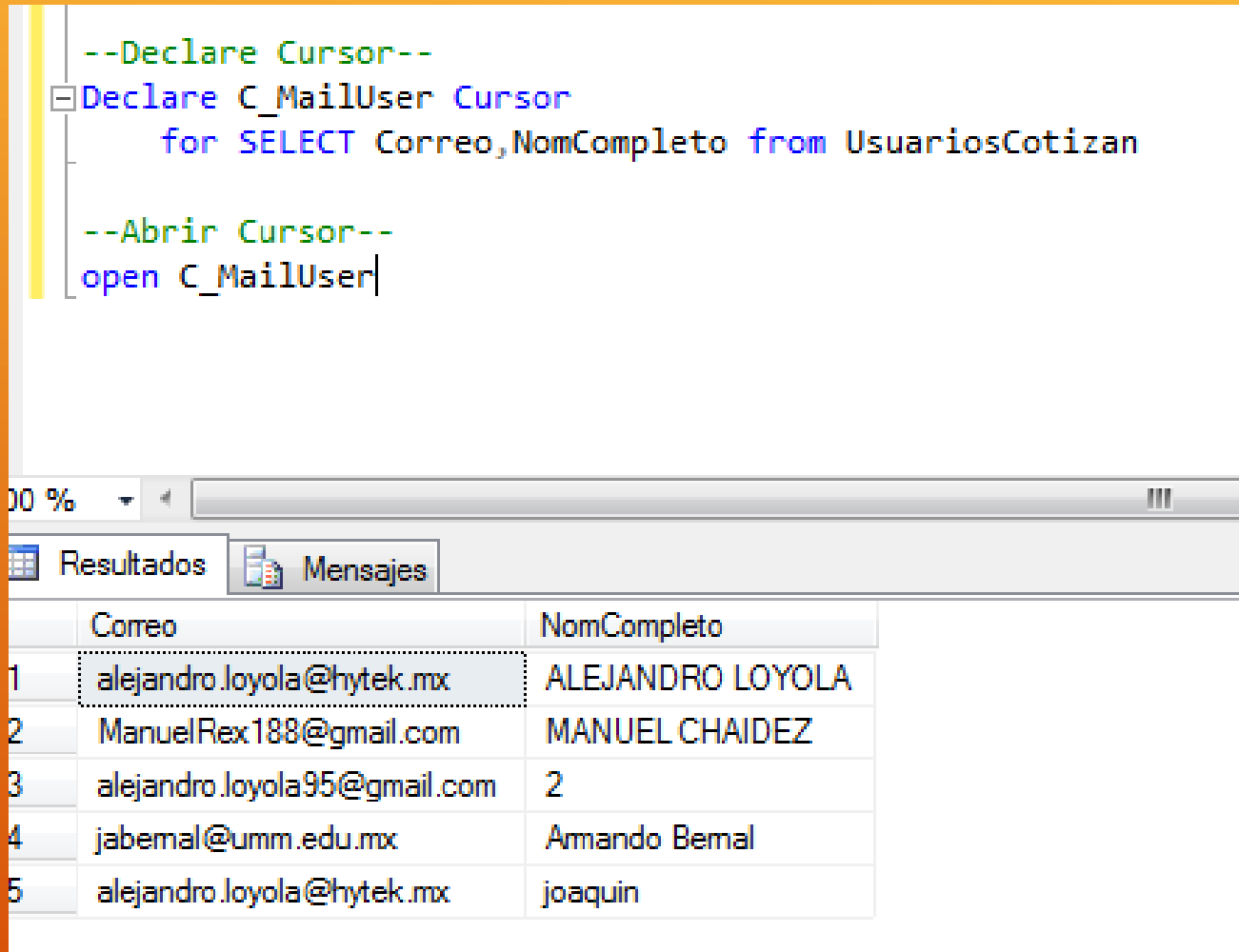
	Correo	NomCompleto
1	alejandro.loyola@hytek.mx	ALEJANDRO LOYOLA
2	ManuelRex188@gmail.com	MANUEL CHAIDEZ
3	alejandro.loyola95@gmail.com	2
4	jabemal@umm.edu.mx	Amando Bernal
5	alejandro.loyola@hytek.mx	joaquin

2.-OPEN CURSOR

► Se abre el cursor con Open

```
--Declare Cursor--
Declare C_MailUser Cursor
    for SELECT Correo,NomCompleto from UsuariosCotizan

--Abrir Cursor--
open C_MailUser
```

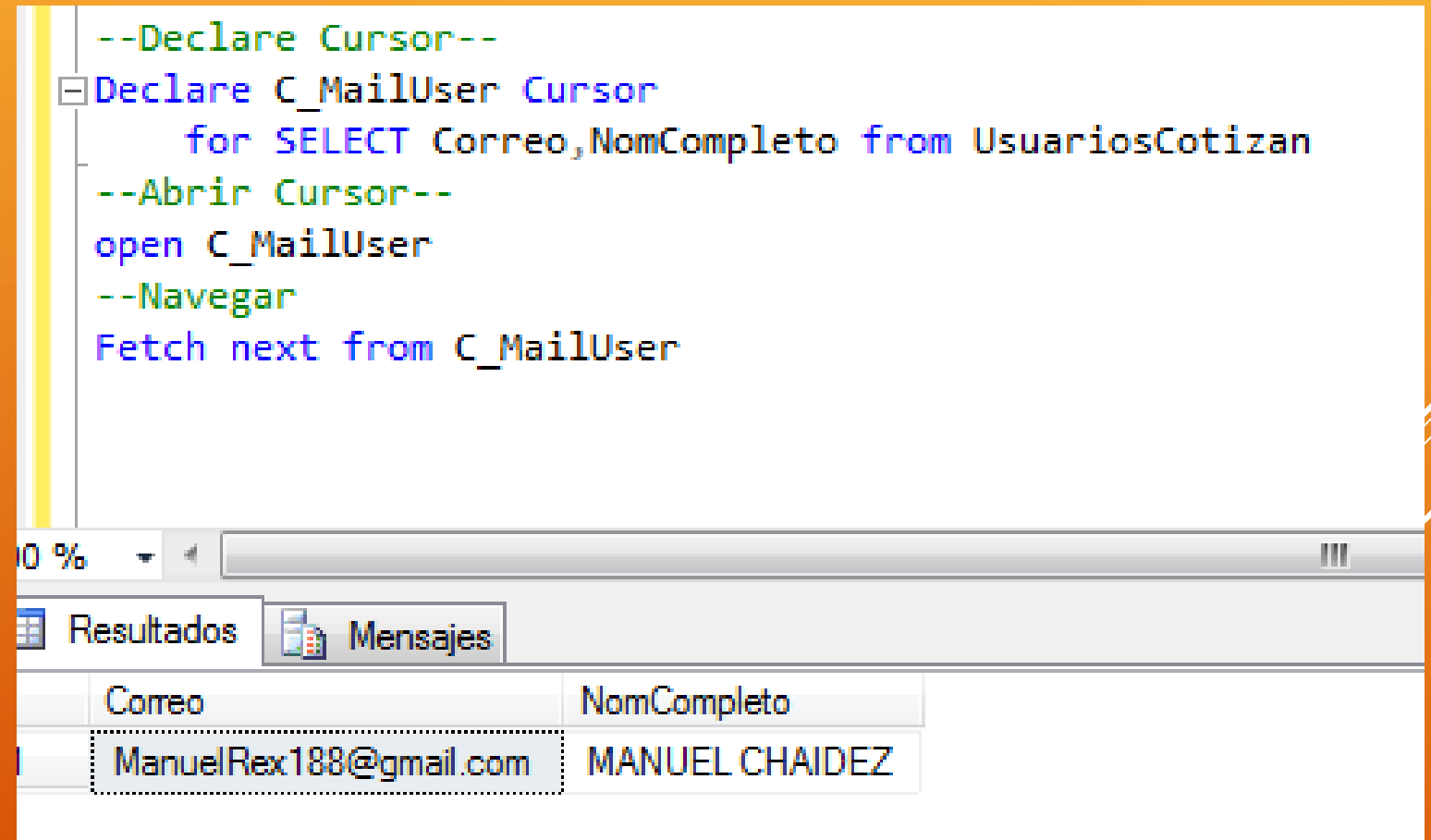


	Correo	NomCompleto
1	alejandro.loyola@hytek.mx	ALEJANDRO LOYOLA
2	ManuelRex188@gmail.com	MANUEL CHAIDEZ
3	alejandro.loyola95@gmail.com	2
4	jabemal@umm.edu.mx	Amando Bernal
5	alejandro.loyola@hytek.mx	joaquin

3.-NAVEGAR

- Para navegar en la estructura guardada en ram usamos la instrucción **Fetch next**

```
--Declare Cursor--  
Declare C_MailUser Cursor  
    for SELECT Correo,NomCompleto from UsuariosCotizan  
--Abrir Cursor--  
open C_MailUser  
--Navegar  
Fetch next from C_MailUser
```



The screenshot shows a database interface with a query editor at the top and a results pane at the bottom. The query editor contains SQL code for declaring and opening a cursor, and fetching the next row. The results pane shows a table with two columns: 'Correo' and 'NomCompleto'. The first row of data is highlighted with a dashed border.

Correo	NomCompleto
ManuelRex188@gmail.com	MANUEL CHAIDEZ

4.-CLOSE CURSOR

Cerramos el cursor
con la instrucción
close

```
--Declare Cursor--  
Declare C_MailUser Cursor  
    for SELECT Correo,NomCompleto from UsuariosCotizan  
--Abrir Cursor--  
open C_MailUser  
--Navegar  
Fetch next from C_MailUser  
--Close  
close C_MailUser
```

100 %

Mensajes

Comandos completados correctamente.

5.-DEALLOCATE

Aunque cerramos el cursor este se mantiene en la memoria por lo cual tenemos que usar la instrucción deallocate

```
--Declare Cursor--  
Declare C_MailUser Cursor  
    for SELECT Correo,NomCompleto from UsuariosCotizan  
--Abrir Cursor--  
open C_MailUser  
--Navegar  
Fetch next from C_MailUser  
--Close  
close C_MailUser  
--Borrar de memoria Ram  
Deallocate C_MailUser
```

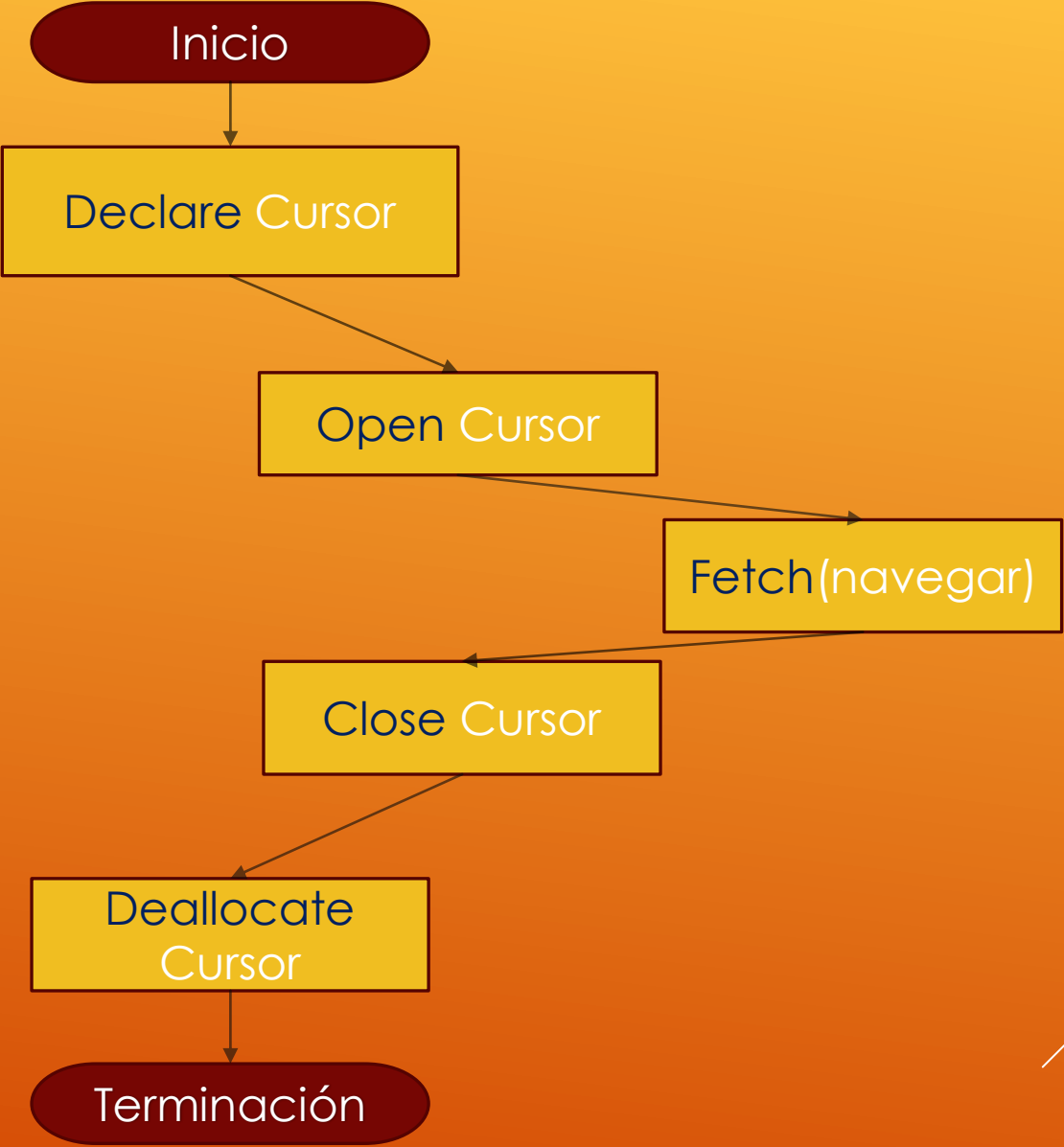
100 %

Mensajes

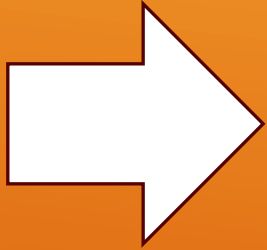
Comandos completados correctamente.

DIAGRAMA

Parámetros



CICLO DE VIDA DE CURSOR(BASICO)



PARAMETROS CURSOR

CICLO DE VIDA DE CURSOR(AVANSADO)

PARÁMETROS

READ ONLY

Previene Updates hechos por este cursor. Por lo tanto este cursor no puede ser usado en una instruccion WHERE CURRENT OF , UPDATE o DELETE.

LOCAL

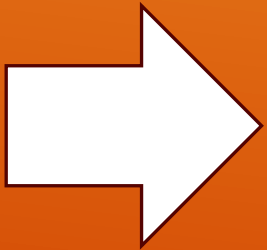
Especifica que el cursor solo puede ser usado en el Store procedure o trigger en el que fue creado, cuando la el procedure termina o el trigger implicitamente el cursor sera borrado de la memoria deallocated

FAST_FORWARD

Especifica un FORWARD_ONLY Y UN READ_ONLY con optimizacion habilitada, este no puede ser especificada si se usa SCROLL o FOR_UPDATE

CICLO DE VIDA DE CURSOR(BASICO)

PARAMETROS CURSOR



CICLO DE VIDA DE CURSOR(AVANZADO)

WHILE

Para empezar a hacer recorridos en nuestra estructura en memoria usamos controles de flujo como el WHILE

```
--Declare Cursor--  
DECLARE C_MailUser Cursor LOCAL FAST_FORWARD READ_ONLY  
    for select NonCompleto from UsuariosCotizan  
  
--Abrir Cursor--  
open C_MailUser  
--Navegar  
Fetch next from C_MailUser  
while (@@FETCH_STATUS= 0)  
BEGIN  
  
    --INSTRUCCION --  
  
        Fetch next from C_MailUser  
END  
--Close  
  
close C_MailUser  
--Borrar de memoria Ram  
Deallocate C_MailUser
```

@@FETCH_STATUS

- Retorna un dato tipo integer el cual representa el estatus de la ultima instruccion FETCH al cursor que actualmente abierto

Return value	Description
0	Instruccion FETCH fue exitosa
-1	La instruccion FETCH ha fallado o la fila ha sobrepasado limite de la estructura
-2	No se ha encontrado la Fila
-9	El cursor no esta haciendo la operacion FETCH

REGLAS IMPORTANTES

- Antes de cada while se debe hacer primero un Fetch para recibir la respuesta de si fue exitoso o no
- Utilizar dentro de la condición de While @@FETCH_STATUS para comprobar si hubo éxito o fallo
- Escribir WHILE() BEGIN END
- Escribir Después de la instrucción Fetch next

```
--Declare Cursor--
DECLARE C_MailUser Cursor LOCAL FAST_FORWARD READ_ONLY
    for select NonCompleto from UsuariosCotizan

--Abrir Cursor--
open C_MailUser
--Navegar
Fetch next from C_MailUser
while ( @@FETCH_STATUS= 0)
BEGIN

    --INSTRUCCION --

        Fetch next from C_MailUser
END
--Close

close C_MailUser
--Borrar de memoria Ram
Deallocate C_MailUser
```

```

-----Declarar Variables
DECLARE @Nombre varchar(200)
DECLARE @IDUser nvarchar(10)

--Declare Cursor--
DECLARE C_MailUser Cursor LOCAL FAST_FORWARD READ_ONLY
for select NomCompleto, IDUsuarioCotiza from UsuariosCotizan

--Abrir Cursor--
open C_MailUser

--Navegar
Fetch next from C_MailUser into @Nombre, @IDUser
while (@@FETCH_STATUS= 0)
BEGIN

    --INSTRUCCION --
    print @Nombre + ' ' + @IDUser

    Fetch next from C_MailUser into @Nombre, @IDUser

END

--Close
close C_MailUser

--Borrar de memoria Ram
Deallocate C_MailUser

```

21 %

Mensajes

ALEJANDRO LOYOLA 7
 MANUEL CHAIDEZ 9
 2 1011
 Armando Bernal 1012
 joaquin 1013


```
DECLARE @CustomerId INT
```

Declare Variable for holding records

```
DECLARE Cursor1 CURSOR READ_ONLY  
FOR
```

Declare the Cursor

```
SELECT CustomerId  
FROM Customers
```

The Select Query

```
OPEN Cursor1
```

Open the Cursor

```
FETCH NEXT FROM Cursor1 INTO  
@CustomerId
```

Fetch the First record into the variable

```
WHILE @@FETCH_STATUS = 0  
BEGIN
```

Loop until records being fetched

```
PRINT @CustomerId
```

Process the record

```
FETCH NEXT FROM Cursor1 INTO  
@CustomerId
```

Fetch the next record

```
END
```

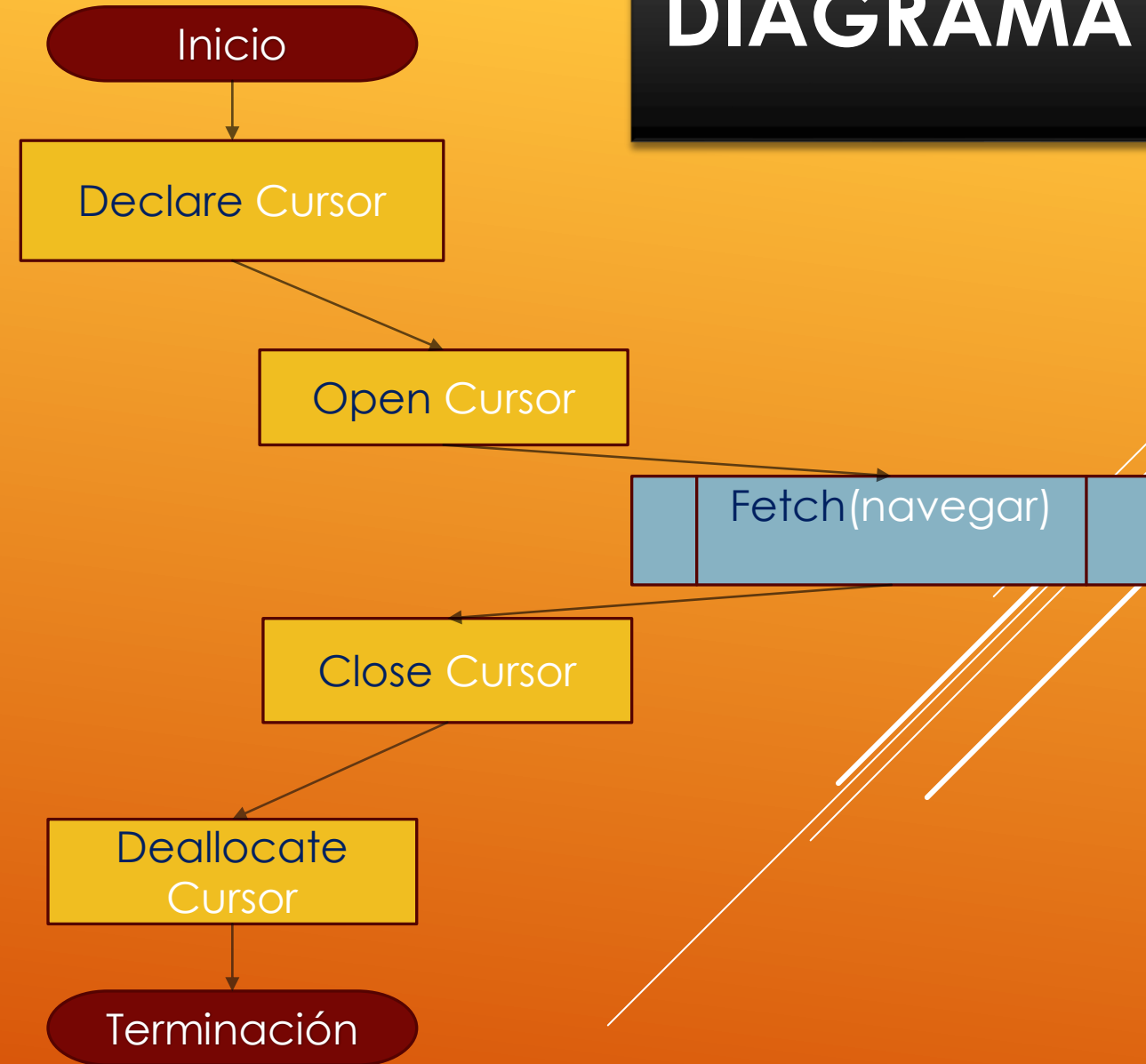
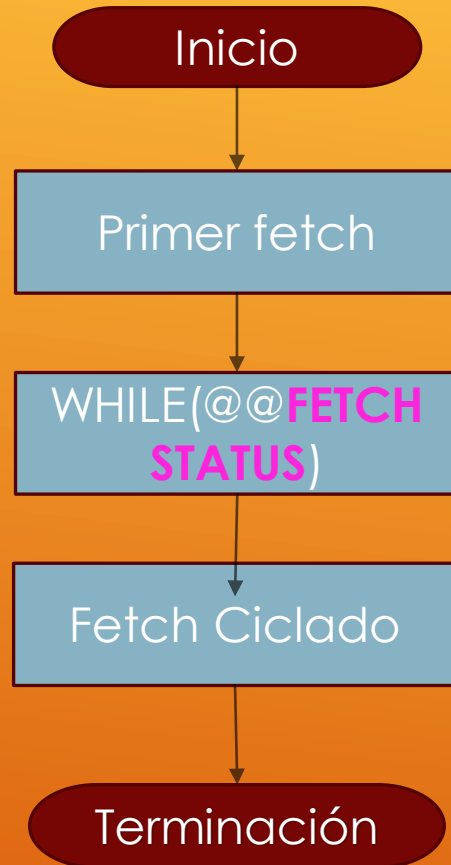
Close the Cursor

```
CLOSE Cursor1
```

Deallocate the Cursor

```
DEALLOCATE Cursor1
```

DIAGRAMA



FETCH INTO

Permite a los datos de las columnas de un Fetch ser puestas en variables locales, el Numero de variables debe igual al numero De columnas del cursor

-----Declarar Variables

DECLARE @Nombre varchar(200)

DECLARE @IDUser nvarchar(10)

--Declare Cursor--

DECLARE C_MailUser Cursor LOCAL FAST_FORWARD READ_ONLY
for select NomCompleto, IDUsuarioCotiza from UsuariosCotizan

--Abrir Cursor--

open C_MailUser

--Navegar

Fetch next from C_MailUser into @Nombre, @IDUser

while (@@FETCH_STATUS= 0)

BEGIN

--INSTRUCCION --

print @Nombre +' '+ @IDUser

Fetch next from C_MailUser into @Nombre, @IDUser

END

--Close

close C_MailUser

--Borrar de memoria Ram

Deallocate C_MailUser

```

-----Declarar Variables
DECLARE @Nombre varchar(200)
DECLARE @IDUser nvarchar(10)

--Declare Cursor--
DECLARE C_MailUser Cursor LOCAL FAST_FORWARD READ_ONLY
for select NomCompleto, IDUsuarioCotiza from UsuariosCotizan

--Abrir Cursor--
open C_MailUser

--Navegar
Fetch next from C_MailUser into @Nombre, @IDUser
while (@@FETCH_STATUS= 0)
BEGIN

    --INSTRUCCION --
    print @Nombre + ' ' + @IDUser

    Fetch next from C_MailUser into @Nombre, @IDUser

END

--Close
close C_MailUser

--Borrar de memoria Ram
Deallocate C_MailUser

```

21 %

Mensajes

ALEJANDRO LOYOLA 7
 MANUEL CHAIDEZ 9
 2 1011
 Armando Bernal 1012
 joaquin 1013