

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD  
DEL CUSCO

FACULTAD DE INGENIERÍA ELÉCTRICA,  
ELECTRÓNICA, INFORMÁTICA Y MECÁNICA  
INGENIERÍA INFORMÁTICA Y DE SISTEMAS



---

## Guía de Laboratorio 6 - Uso de Cursores

---

*Alumno:*

Ian Logan Will Quispe Ventura

211359

*Docente:*

Karelia Medina Miranda

*Curso:*

Fundamentos y Diseño de Base de Datos

Cusco - Perú  
2023 - II

## Ejercicio Número 2

Determinar la relación de los tres prestatarios rurales más representativos de cada comunidad. Los más representativos son los que tienen mayor número de préstamos.

```
begin
declare @CodComunidad varchar(12);
declare @CodPrestatario varchar(12);
declare @NombrePrestatario varchar(40);
declare @NumPrestamos int;

-- Crear una tabla temporal para almacenar los
    resultados
create table #Resultados (
    CodComunidad varchar(12),
    CodPrestatario varchar(12),
    NombrePrestatario varchar(40),
    NumPrestamos int);

-- Declarar cursor para comunidades
declare cursor_Comunidad cursor
for
select CodComunidad from Comunidad;
open cursor_Comunidad;

-- Recorrer comunidades
fetch next from cursor_Comunidad into @CodComunidad;
while @@FETCH_STATUS = 0
begin
    -- Cursor que contará los préstamos de cada
        prestatario en cada comunidad
    declare cursor_Prestatario cursor
    for
    select P.CodPrestatario, Pr.Nombres, count(*) as
        NumPrestamos
    from Prestamo P
    join Prestatario Pr on P.CodPrestatario = Pr.
        CodPrestatario
    where Pr.CodComunidad = @CodComunidad
    group by P.CodPrestatario, Pr.Nombres;
```

```

open cursor_Prestatario;
-- Recorremos cada uno de los prestatarios para
  seleccionar los tres con mayor número de pré
  stamos
declare @Counter int = 0;
fetch next from cursor_Prestatario into
  @CodPrestatario, @NombrePrestatario,
  @NumPrestamos;
while @@FETCH_STATUS = 0 and @Counter < 3
begin
  -- Guardamos los resultados obtenidos en la
    tabla temporal
  insert into #Resultados (CodComunidad,
    CodPrestatario, NombrePrestatario,
    NumPrestamos)
  values (@CodComunidad, @CodPrestatario,
    @NombrePrestatario, @NumPrestamos);

  set @Counter = @Counter + 1;
  -- Siguiente prestatario
  fetch next from cursor_Prestatario into
    @CodPrestatario, @NombrePrestatario,
    @NumPrestamos;
end;
close cursor_Prestatario;
deallocate cursor_Prestatario;
-- Siguiente comunidad
fetch next from cursor_Comunidad into
  @CodComunidad;
end;

close cursor_Comunidad;
deallocate cursor_Comunidad;
select * from #Resultados;
-- Eliminar la tabla temporal
drop table #Resultados;
end;

```

## Resultados

	CodComunidad	CodPrestatario	NombrePrestatario	NumPrestamos
1	C001	P02	Delgado Caceres Ana Maria	3
2	C001	P11	Choquehuanca Paredes Adan	3
3	C001	P26	Castro Humpire Jose Luis	4
4	C002	P01	Lopez Caceres Jose Antonio	14
5	C002	P22	Caceres Torres Dario	1
6	C003	P03	Candia Calderon Maria	3
7	C003	P037	Alvarez Mejia Juan Carlos	1
8	C003	P23	Guzman Perez Eddy	1
9	C004	P27	Colque Rivera Felipe	4
10	C005	P039	Gonzales Rosas Veronica	2
11	C007	P07	Huaman Pariapaza Yvan	3
12	C007	P08	Noa Huamani Francisco	3
13	C007	P09	Segovia Quispe Nancy	4
14	C008	P10	Rios Campos Adams	4
15	C008	P12	Quispe Mamani Jaime	3
16	C008	P25	Meza Huamani Edgar	4
17	C010	P16	Rado Rojas Julia Olimpia	3
18	C010	P17	Gonzales Estrada Ramon	4
19	C010	P18	Gallegos Holgado Rufina	3
20	C011	P19	Ccahuana Castillo Hector Luis	4
21	C011	P20	Vargas Arce Lisbeth	3
22	C011	P21	Florez Montalvo Silvia	3
23	C014	P28	Arzubialde Rosas Julio Cesar	4
24	C014	P29	Morales Chacon Virginia	5
25	C014	P30	Calderon Manrique Cristina	4

### Problema Número 3

Determinar los movimientos de un prestatario. Debe mostrar cada préstamo con sus respectivas cancelaciones. Los importes del préstamo van en la columna del Debe y los importes de las cancelaciones van al haber. Considerar que un prestatario puede tener varios préstamos.

```
begin
declare @CodPrestatario TCodPrestatario;
set @CodPrestatario = 'P01';

-- Crear tabla para almacenar los movimientos de los
  prestatarios
create table #Movimientos (
    DocMovimiento varchar(12),
    FechaMovimiento datetime,
    Debe numeric(15,2),
    Haber numeric(15,2),
    SaldoPrestado numeric(15,2),
    SaldoTotal numeric(15,2)
);

-- Declarar variables
declare @SaldoPrestado numeric(15,2) = 0;
declare @SaldoTotal numeric(15,2) = 0;

-- Obtener préstamos y cancelaciones del prestatario
insert into #Movimientos
select P.DocPrestamo as DocMovimiento,
       P.FechaPrestamo as FechaMovimiento,
       P.Importe as Debe, 0 as Haber,
       P.Importe as SaldoPrestado,
       @SaldoTotal + P.Importe as SaldoTotal

from Prestamo P
where P.CodPrestatario = @CodPrestatario
order by P.FechaPrestamo;
```

```

insert into #Movimientos
select A.DocCancelacion as DocMovimiento,
       A.FechaCancelacion as FechaMovimiento,
       0 as Debe, A.Importe as Haber,
       -A.Importe as SaldoPrestado,
       @SaldoTotal - A.Importe as SaldoTotal

from Amortizacion A
where A.DocPrestamo in (select P.DocPrestamo
                        from Prestamo P where P.CodPrestatario =
                        @CodPrestatario)
order by A.FechaCancelacion;

select * from #Movimientos;
drop table #Movimientos;
end;

```



## Resultados

	DocMovimiento	FechaMovimiento	Debe	Haber	SaldoPrestado	SaldoTotal
1	PR10001	2004-01-02 00:00:00.000	1500.00	0.00	1500.00	1500.00
2	PR10003	2004-01-24 00:00:00.000	3000.00	0.00	3000.00	3000.00
3	PR10006	2009-01-07 00:00:00.000	1000.00	0.00	1000.00	1000.00
4	PR10007	2010-03-05 00:00:00.000	7000.00	0.00	7000.00	7000.00
5	PR10008	2013-08-08 00:00:00.000	7000.00	0.00	7000.00	7000.00
6	PR10009	2008-02-07 00:00:00.000	7000.00	0.00	7000.00	7000.00
7	PR10010	2009-04-01 00:00:00.000	7000.00	0.00	7000.00	7000.00
8	PR10011	2007-09-07 00:00:00.000	7000.00	0.00	7000.00	7000.00
9	PR10012	2008-06-05 00:00:00.000	7000.00	0.00	7000.00	7000.00
10	PR10013	2012-02-07 00:00:00.000	7000.00	0.00	7000.00	7000.00
11	PR10014	2014-01-09 00:00:00.000	7000.00	0.00	7000.00	7000.00
12	PR10080	2008-07-15 00:00:00.000	6000.00	0.00	6000.00	6000.00
13	PR10116	2005-03-23 00:00:00.000	5000.00	0.00	5000.00	5000.00
14	PR10122	2007-08-28 00:00:00.000	3000.00	0.00	3000.00	3000.00
15	C10000	2004-03-03 00:00:00.000	0.00	500....	-500.00	-500.00
16	C10001	2004-05-04 00:00:00.000	0.00	600....	-600.00	-600.00
17	C10136	2008-12-15 00:00:00.000	0.00	200...	-2000.00	-2000.00
18	C10137	2009-12-27 00:00:00.000	0.00	200...	-2000.00	-2000.00
19	C10138	2010-07-15 00:00:00.000	0.00	200...	-2000.00	-2000.00
20	C10202	2005-04-20 00:00:00.000	0.00	500...	-5000.00	-5000.00
21	C10214	2008-04-02 00:00:00.000	0.00	100...	-1000.00	-1000.00
22	C10215	2009-04-02 00:00:00.000	0.00	100...	-1000.00	-1000.00