

## EJERCICIOS REPASO

### PROCEDIMIENTO

#### *BD Pacientes*

Para aquellos pacientes que tienen más de 3 visitas, muestra por pantalla sus visitas, incrementando el importe un 20% de aquellas que sean por 'Gripe'. Si son por 'Úlcera', se deberá incrementar un 40%.

Al procedimiento que se llamará en el script principal se le pasará el código de Paciente y no devolverá nada, simplemente imprimirá por pantalla cada visita de cada paciente.

Desde el script principal será desde donde se imprima por pantalla la siguiente información:

**Código Paciente   -   Nombre Paciente   -   Nº Visitas**

```
--Procedimiento
CREATE PROCEDURE procvisitas @codpaciente smallint
AS
DECLARE @CONT INT, @MAX INT
DECLARE @ENFER NVARCHAR(25), @IMPOR INT
SELECT @CONT = MIN([Nº VISITA]) FROM VISITAS WHERE [CODIGO PACIENTE] = @codpaciente
SELECT @MAX = MAX([Nº VISITA]) FROM VISITAS WHERE [CODIGO PACIENTE] = @codpaciente
PRINT space(2) + '- Visitas:'
WHILE (@CONT <= @MAX)
BEGIN
    IF (EXISTS(SELECT * FROM VISITAS WHERE
        [Nº VISITA] = @CONT AND [CODIGO PACIENTE]=@codpaciente))
    BEGIN
        SELECT @ENFER=ENFERMEDAD, @IMPOR=IMPORTE FROM VISITAS WHERE [Nº VISITA] =
            @CONT
        IF (@ENFER='Gripe')
            BEGIN
                SET @impor = @impor * 1.2
            END
        ELSE IF (@ENFER='Úlcera')
            BEGIN
                SET @impor = @impor * 1.4
            END
        PRINT space(6) + ' * ' + CAST(@CONT AS NVARCHAR(5)) + ' ' + @ENFER +
            ' ' + CAST(@IMPOR AS NVARCHAR(10))
    END
    SET @CONT += 1
END

--Script Principal
DECLARE @CONT INT, @MAX INT
DECLARE @NOMBRE NVARCHAR(30), @NVISITAS INT
SELECT @CONT=MIN(CODIGO), @MAX=MAX(CODIGO) FROM PACIENTES
PRINT 'Cod. Paciente - Nombre - NºVisitas'
PRINT '-----'
WHILE (@CONT<=@MAX)
BEGIN
    IF (EXISTS(SELECT v.[CODIGO PACIENTE] FROM PACIENTES p inner join VISITAS v ON
        v.[CODIGO PACIENTE] = p.CODIGO WHERE [CODIGO PACIENTE] = @CONT GROUP BY v.[CODIGO
        PACIENTE] HAVING COUNT(*) > 3))
    BEGIN
        SELECT @NOMBRE = NOMBRE FROM PACIENTES
        WHERE CODIGO = @CONT
        PRINT CAST(@CONT AS NVARCHAR(6)) + space(8)+ @NOMBRE
        EXEC procvisitas @cont
    END
END
```

```

        SET @CONT += 1
    END

```

## FUNCIÓN

### BD Videoclub

Muestra por cada socio la película con peor calificación y con el formato que se requiera en cada momento.

La función recibirá el código de socio y de formato de stock, y a partir de aquí tendrá que devolver el nombre de la película como parámetro de retorno.

Desde el script principal se mostrará todo por pantalla tal cual

**Código Socio - Nombre Socio - Cliente VIP**

7 - JUAN - SI

-----  
Titanic

--Script Principal

```

DECLARE @cont int, @max int

```

```

DECLARE @nombre nvarchar(30), @isvip bit, @nombrePeli nvarchar(40)

```

```

SELECT @cont = min(CODIGO), @max = max(CODIGO) from SOCIOS

```

```

PRINT 'C.Socio - Nombre Socio - Cliente VIP'

```

```

WHILE(@cont <= @max)

```

```

    BEGIN

```

```

        if(exists(SELECT * FROM SOCIOS WHERE CODIGO = @cont))

```

```

            BEGIN

```

```

                SELECT @nombre = NOMBRE, @isvip = CLIENTEVIP FROM SOCIOS WHERE
                CODIGO = @cont

```

```

                SET @nombrePeli = dbo.minimaCalifica(@cont, 'VHS')

```

```

                if (@isvip = 1)

```

```

                    BEGIN

```

```

                        PRINT cast(@cont as nvarchar(6)) + space(9) + @nombre
                        + space(9) + 'Sí'

```

```

                        PRINT '-----'

```

```

                        PRINT space(6) + @nombrePeli

```

```

                    END

```

```

                else

```

```

                    BEGIN

```

```

                        PRINT cast(@cont as nvarchar(6)) + space(9) + @nombre
                        + space(9) + 'No'

```

```

                    END

```

```

            END

```

```

        SET @cont += 1

```

```

    END

```

--FUNCIÓN

```

CREATE FUNCTION minimaCalifica (@codigoSocio int, @codigostock nvarchar(3))

```

```

RETURNS nvarchar(40)

```

```

AS

```

```

    BEGIN

```

```

        declare @nompelicula nvarchar(40), @califica int

```

```
select top 1 @califica=calificacion, @nompelicula=titulo from alquiler a inner join  
stock st on st.CODIGO = a.CODIGO_STOCK inner join PELICULAS p  
on p.CODIGO = st.CODIGO_PELICULA WHERE CODIGO_SOCIO = @codigoSocio  
AND FORMATO = @codigostock order by CALIFICACION asc  
RETURN @nompelicula
```

END