Repasando Vistas, Funciones y Procedimientos Almacenados. <https://mariadb.com/kb/en/library/documentation/>

Crea una carpeta **Actividad3\_UT4\_VosIniciales** donde irás introduciendo los elementos creados:

1. Este documento con las sentencias copiadas y pegadas
2. y la BD **bdprestamosVosIniciales** final exportada

Con las SELECT una vez probadas, se puede utilizar phpMyAdmin para crear el código PHP, utilizando el enlace: 

Acciones:

Crea una BD **bdprestamosVosiniciales** e importa la base de datos **bdprestamos.sql** que se encuentra en la carpeta **Recursos Actividad3\_UT4**.

Incluye en un script las sentencias SELECT que permitan:

1. Mostrar el DNI, nombre, apellidos, y teléfono de los socios de Donostia, Pamplona y Logroño (con la clausula IN).

SELECT dni, nombre, apellidos, Telefono

from socios

where Localidad in ('Pamplona', 'Donostia', 'Logroño')

1. Los isbn y nº de carnet de los prestamos realizados en el mes de abril del 2012 ordenados por fecha (BETWEEN).

select isbn, IdCarnet

from prestamos

where Fecha BETWEEN "2012-04-1" and "2012-04-30"

ORDER by Fecha

1. El nº de carnets que se contrataron en cada año.

SELECT COUNT(IdCarnet) as "Numero carnet", year(FechaCont) as "Año"

FROM carnets

GROUP BY year(FechaCont)

1. El isbn, nº de carnet, el DNI y la Fecha de los 10 primeros prestamos(fecha más antigua) utiliza alias para las tablas.

SELECT isbn as "ISBN", IdCarnet as "Nº Carnet", (SELECT DNI from carnets c where c.IdCarnet = p.IdCarnet) as "DNI", Fecha

FROM prestamos p

order by fecha ASC

limit 10

1. El nº de carnets (utiliza el alias Número)por cada Tipo, que contrataron su carnet en la oficina Biblosa, ordenados por orden alfabético de Tipo de carnet.



SELECT count(IdCarnet) as "Nº Carnets", Tipo

FROM `carnets`

WHERE Oficina = "Biblosa"

GROUP BY Tipo

ORDER BY tipo

1. Modifica los tipos de carnet ‘Moroso’ a ‘Expulsado’ .

Primero realiza una SELECT para comprobar los registros a modificar.

(Para recuperar la información inicial se intercambia los valores de la modificación, de Explusado a Moroso)

SELECT \*

FROM `carnets`

WHERE tipo = "Moroso"

UPDATE `carnets` SET Tipo="Expulsado"

WHERE tipo = "Moroso"

1. Modifica/los tipos de carnet ‘Normal’ a ‘Corriente’ de los socios de Pamplona.
2. Primero realiza una SELECT para comprobar que realmente son los registros que queremos modificar:

SELECT \* FROM carnets c INNER JOIN socios s ON c.DNI=s.DNI WHERE tipo="Normal" AND localidad="Pamplona"

1. Después la UPDATE

UPDATE `carnets` c INNER JOIN socios s ON c.DNI=s.DNI

SET `Tipo`= "Corriente"

WHERE tipo = "Normal" AND localidad = "Pamplona"

1. Realiza el borrado de los prestamos del socio cuyo DNI es 76325102. (Recuerda que si se quiere recuperar la información se deberá realizar las INSERT correspondientes)

SELECT\*

FROM prestamos

where IdCarnet = (SELECT IdCarnet from carnets where dni = "76325102")

DELETE FROM prestamos

where IdCarnet = (SELECT IdCarnet from carnets where dni = "76325102")

1. Escribe 1 sentencia CRUD a elegir (Insert y/o Select y/o Update y/o Delete) Enunciado y solución.

Sacar el nombre y DNI de los socios que no han realizado ningún préstamo

--Comprobamos los carnets que no han realizado ningún préstamo

SELECT \*

FROM carnets c LEFT JOIN prestamos p ON c.IdCarnet = p.IdCarnet

WHERE p.IdFicha IS NULL

--Sacamos el dni y el nombre de dichos carnets

SELECT DNI, nombre

FROM socios

WHERE DNI IN (SELECT c.DNI

FROM carnets c LEFT JOIN prestamos p ON c.IdCarnet = p.IdCarnet

WHERE p.IdFicha IS NULL)

1. Realiza una función llamada **mesFechaVosIniciales**, que muestre el mes en castellano y el año de una determinada fecha y prueba la función. Busca en internet la forma de crear funciones. Ejemplos de enlaces:

<http://sql-info.de/mysql/examples/create-function-examples.html>

<https://www.nosolocodigo.com/como-crear-funciones-en-mysql>

Revisa la documentación facilitada. Realiza pruebas progresivas en dificultad.

Completa la función:

**DROP FUNCTION IF EXISTS fechaMesVosIniciales;**

**DELIMITER //**

**CREATE FUNCTION fechaMesVosIniciales (………..…) RETURNS varchar(20)**

**BEGIN**

**DECLARE salida ………………….;**

**SET salida = CASE month(vfecha)**

**when 1 then 'Enero'**

**when 2 then 'Febrero'**

**…………………………………………………**

**else "No es correcto" end ;**

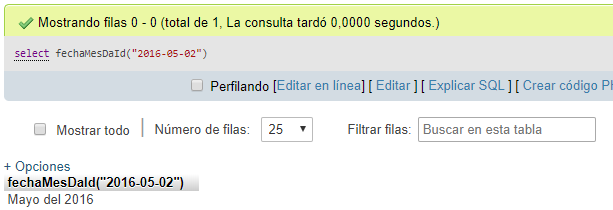
**RETURN concat(………………………………………………………);**

**END;**

Se puede cambiar la función siempre que obtenga la salida indicada

Ejemplo: **2016/05/02** deberá dar como resultado **Mayo del 2016**

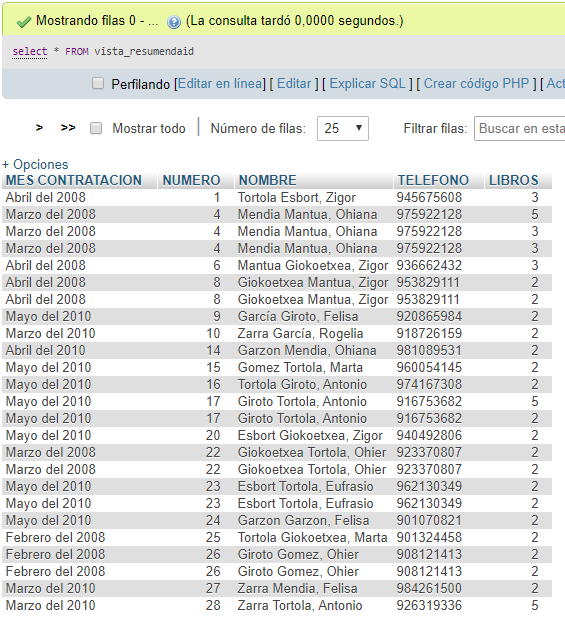
**Captura de la prueba de la función:**

****

1. Crear una vista llamada **vista\_ResumenVosiniciales**, que muestre el Mes (Nombre del mes de FechaCont, para ello utiliza la función anterior), Número (Identificador del carnet), Nombre (Apellidos y nombre separados por una coma), Teléfono, Libros (El número de Libros alquilados por socio y por mes). Prueba la vista.

[CREATE](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/create-view.html) [VIEW](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/create-view.html) vista\_ResumenDaId AS [SELECT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/select.html) fechaMesDaId(fechaCont) AS "MES CONTRATACION", c.idCarnet AS "NUMERO", CONCAT(Apellidos, ", ",nombre) AS "NOMBRE" ,telefono AS "TELEFONO", [COUNT](http://localhost/phpmyadmin/url.php?url=https://dev.mysql.com/doc/refman/5.5/en/group-by-functions.html#function_count)(idFicha) AS "LIBROS" FROM carnets c JOIN prestamos p ON c.IdCarnet = p.IdCarnet JOIN socios s ON s.DNI = c.DNI GROUP BY p.IdCarnet, MONTH(p.Fecha)

**Captura de la prueba de la vista:**



1. Un Procedimiento Almacenado (PA o sp): **prestamosFechaVosiniciales**, que muestre los préstamos entre dos fechas. Prueba el PA .

DELIMITER $$

DROP procedure IF EXISTS prestamosFechaDaId $$

CREATE procedure prestamosFechaDaId(IN fechaInicio date, IN fechaFin date)

BEGIN

SELECT \*

FROM prestamos

WHERE fecha BETWEEN fechaInicio AND fechaFin;

END $$

DELIMITER ;

**Pega la captura de la prueba del procedimiento almacenado:**



* + - * Una variante del anterior. Un PA: n**umPrestamosVosiniciales**, que muestre el nº de préstamos entre entre dos fechas. Se puede hacer con parámetro de salida o sin él.
* **Sin parámetro de salida:**

**DELIMITER $$**

**DROP procedure IF EXISTS numPrestamosDaId $$**

**CREATE procedure numPrestamosDaId(IN fechaInicio date, IN fechaFin date)**

**BEGIN**

**SELECT count(\*) as "Total prestamos"**

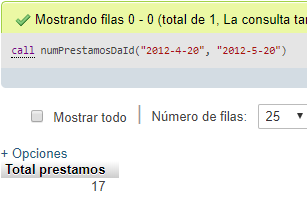
**FROM prestamos**

**WHERE fecha BETWEEN fechaInicio AND fechaFin;**

**END $$**

**DELIMITER ;**

**Pega la captura de la prueba del procedimiento almacenado:**



* **Con parámetro de salida para el nº de préstamos:**

**DELIMITER $$**

**DROP procedure IF EXISTS numPrestamosDaId\_Salida $$**

**CREATE procedure numPrestamosDaId\_Salida(IN fechaInicio date, IN fechaFin date, OUT total INT)**

**BEGIN**

**SELECT count(\*) as "Total prestamos"**

**FROM prestamos**

**WHERE fecha BETWEEN fechaInicio AND fechaFin**

**INTO total;**

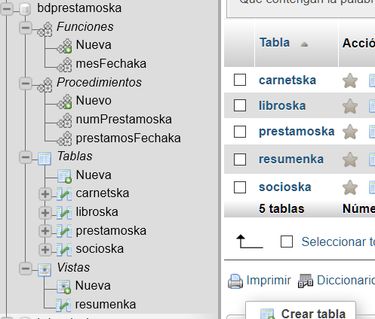
**END $$**

**DELIMITER ;**

**Pega la captura de la prueba del procedimiento almacenado:**



Copia y pega en el documento que vas a entregar una captura similar a:



Exporta la **bdprestamosVosIniciales.sql**. Comprime la carpeta donde se encuentran los elementos y súbela a moodle.

