1. Definición de vista. (1pto)

Una vista no es mas que una consulta almacenada en el servidor y a la que podemos invocar;

1. ¿Cuál es la sintaxis para la creación de una vista? Explícala lo más detalladamente posible. (1pto)

CREATE

[OR REPLACE]

[ALGORITHM = {UNDEFINED | MERGE | TEMPTABLE}]

[DEFINER = { user | CURRENT\_USER }]

[SQL SECURITY { DEFINER | INVOKER }]

VIEW view\_name [(column\_list)]

AS select\_statement

[WITH [CASCADED | LOCAL] CHECK OPTION]

1. ¿Una vista es como una tabla? Razona tu respuesta. (0,5 ptos)

No, una tabla almacena información, pero una vista es una consulta que nos devuelve la información de una o varias tablas en forma de tabla temporal;

1. ¿Se puede usar las vistas dentro de los procedimientos almacenados? (0,5 ptos)

Si, ya que las vistas al fin y al cabo son consultas ya predefinidas que nos devuelven datos;

1. Explica, lo más detallado posible, utilizando un ejemplo las ventajas de utilización de las vistas en cuanto a: (2 ptos)

* Seguridad y confidencialidad:
* Facilidad de uso para consultas “complejas”

Dado que al invocar a una vista esta nos devuelve los datos de una consulta y no accedemos directamente a la tabla sino al resultado es mucho mas seguro operar sobre el resultado que operar directamente sobre las tablas.

Por otra parte como una vista no es mas que una consulta podemos crear vistas simples y llamarlas dentro de consultas mas complejas haciendo todo mas sencillo y entendible, o incluso crear vistas complejas a las cuales es mas fácil referencia y pueden hacer uso de ellas usuarios con menos conocimientos;

1. ¿Qué privilegio de requiere para poder crear vistas? ¿Qué operaciones se pueden realizar, además de la creación, sobre las vistas? (1 pto)

Para poder crear una vista deberemos tener los mismos privilegios que para crear una tabla o un procedimiento almacenado. Operaciones CRUD, Create, Read, Update and Delete;

1. Realiza cuatro vistas sobre la base de datos de tu proyecto. Explica qué hace cada una de ellas. Haz una captura de pantalla que muestre tanto el código MySql como el resultado obtenido. (4 ptos)

CREATE VIEW peliculasAndDirectores AS SELECT p.titulo, d.nombre FROM pelicula p INNER JOIN director d ON p.director = d.id;

CREATE VIEW peliculasAndDuracion AS SELECT concat(titulo, ' (', duración, ')') FROM pelicula;

CREATE VIEW resetDuracion AS UPDATE pelicula SET puntuación = 0;

CREATE VIEW borrarPeliculasGeneroComedia AS DELETE FROM pelicula WHERE genero = 'Comedia'