Question 1

???????????????????????????

Question 2

SELECT \* FROM invoice inv right join item it on inv.item =it.item\_codigo where item is null

Question 3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Modifier | Class | Package | Subclass | World |
| Public | X |  | X | X |
| Protected | X |  | X | X |
| Default | X |  | x | X |
| Private |  |  |  | X |

Question 4

El proposito es de capturar los error sin permitir que la aplicacion collapse,

Try{

Integer numero = null;

System.out.println(“>>>”+numero.toString());

}cath(Exception e){

Log.error(3);

}finally{

System.out.println(“terminar”);

}

Question 5

Java uilitiza la serializacion para trtasmitir los objetos en borma binara par que que tiene la JVM trabajar.

Question 6

El uso modificador *static*es compartir el valor de una variable entre las intancias de una misma clase, si declaramos una todos los objetos que declaremos de esa mima clase compartiran el valor variable *static*

Question 7

1. un conjunto limitado de conexiones a una base de datos
2. las conexiones se encuentran en un estado pendiente pero igual se cargan en memoria es importante verificar la memoria del servidor de app para no sobrecargarlo.

Question 8

* Data Access Object.- Los Objetos de Acceso a Datos pueden usarse en [Java](http://es.wikipedia.org/wiki/Lenguaje_de_programaci%C3%B3n_Java) para aislar a una aplicación de la tecnología de persistencia Java subyacente la cual podría ser [JDBC](http://es.wikipedia.org/wiki/JDBC), [JDO](http://es.wikipedia.org/w/index.php?title=JDO&action=edit&redlink=1), [Enterprise JavaBeans](http://es.wikipedia.org/wiki/Enterprise_JavaBeans), [TopLink](http://es.wikipedia.org/w/index.php?title=TopLink&action=edit&redlink=1), [Hibernate](http://es.wikipedia.org/wiki/Hibernate_%28Java%29), [iBATIS](http://es.wikipedia.org/wiki/IBATIS), o cualquier otra tecnología de persistencia.

Usando Objetos de Acceso de Datos significa que la tecnología subyacente puede ser actualizada o cambiada sin cambiar otras partes de la aplicación.

* Composite View.- se componen de varias subvistas atómicas. Cada componente de la plantilla se podría incluir dinámicamente dentro del total y la distribución de la página se maneja independientemente del contenido.
* FrontController.- Para aplicaciones Web se recomienda utilizar este patrón que obliga a que todas las peticiones hechas a nuestra aplicación pasen por un servlet Controlador.
* Sesión facade.- patrón que nos permite dar modelo de acceso a la parte central de la aplicación hecha en EJBs.

Question 9

public static void main(String [] args) {

new Thread1().start(); -- error no existe el myMethod1

new Thread2().start();

RequestLogIteration(5000);

TemporalSroreIteration(100000);

}

Question 10

La interfaz "Local" permite al código cliente manipular métodos de clase del EJB que no están asociados a ninguna instancia particular.

La interfaz “remota” especifica los métodos de instancia públicos encargados de realizar las operaciones

Question 11

???????????????????????????????????????????????

Question 12

A0001 Billy 700.00

A0002 Faktum 500.00

A0003 Lack 11600.00

A0004 Besta 5200.00

A0005 sova 100.00