

## 2ª GUIA – Cuestionario

Nombre: Calva Hernández José Manuel

Fecha: 06/11/17

Grupo: 2CM3

### Excepciones

- Que llamadas a métodos se ponen dentro de un bloque try?

Aquellas que pueden generar una excepción

- El código dentro del bloque try como se escribe?

Como si se tratase de un mundo ideal ya que los errores casi nunca deberían de aparecer

- Si una excepción es lanzada en cualquier parte del bloque try que instrucciones se ejecutan

Las instrucciones contenidas dentro del bloque catch que haga referencia a esa excepción

- Para lanzar una excepción que operador se usa?

Se utiliza el operador throw

- Cual es el argumento para el operador throw?

El nombre de la excepción que va a lanzar

- Que instrucción aborta el método y lanza la excepción a lo largo de la pila de llamadas?

La instrucción throw

VII.-Que pasa sin ningún método en la pila de llamadas (ni siquiera main) maneja la excepción.

La máquina virtual java finaliza su ejecución con un mensaje de error.

Cual es el esquema que usa Java para el manejo de errores?

```
try{  
    //código que podría lanzar una excepción  
}catch (tipoExcepción a){  
    //procesar el tipo de excepción  
} catch (otroTipoExcepción a){  
    //procesar el tipo de excepción  
}finally{  
    //código que se ejecuta después del try, se haya producido o no una excepción  
}
```

Cual es el esquema que usa Java para sincronizar el acceso a datos compartidos?

```
Synchronized (objeto){  
    // En este método solo puede acceder un hilo a la vez  
}
```

b1) Cual es el modelo Cliente-Servidor?

Es donde a través de dos programas, un cliente y un servidor, pueden intercambiar información, para esto, el cliente debe de conocer la ip del servidor y simplemente mandar una solicitud para establecer conexión con este.

Cual es el modelo de componentes en Java?

Component es una clase abstracta, que solo se pueden crear objetos pertenecientes a sus subclases y no al ella misma. Las subclases que representan los elementos comunes GUI son: Button,Checkbox, Choice, Label, List, Scrollbar, TextArea, y TextField, y cada uno de ellos tiene un comportamiento predefinido

Cual es el modelo de eventos en Java?

Propone la existencia de 3 tipos de objetos:

- Events: que pueden suceder y para los cuales es de interés registrar información de estado que detalle para cada tipo de evento los aspectos particulares del caso. Ejemplos de eventos de interés podrían ser: un evento de mouse o uno de teclado.

- Sources: eventos que los disparan, en general frente a una acción del usuario. Fuentes de eventos puede ser: la ventana de diálogo donde se visualiza la imagen
- Listeners: subscriben a las fuentes de eventos de su interés y son comunicados oportunamente por estas con el evento adecuado al caso.

En Java cual es el equivalente a RPC (Remote procedure Call)

Su equivalente es Java RMI (Remote Method Invocation)

## Hilos

f) Que es un quantum?

Un quantum de tiempo es un pequeño intervalo de tiempo que se asigna a un proceso para que ejecute sus instrucciones.

g) Que es un cambio de contexto?

Es el paso de la ejecución de un hilo al siguiente

h) Que es una condición de competencia?

Es la condición que permite una relativa prioridad de un hilo sobre otro.

i) Que es una región crítica?

Porción de un programa en la que se accede a un recurso compartido que no debe ser accedido por más de un hilo en ejecución

j) Que es exclusión mutua?

Es la propiedad de permitir sólo a un subproceso adquirir los recursos a la vez.

k) Que es espera ocupada?

Estado donde un proceso repetidamente verifica la condición, como esperar la entrada por teclado o que el ingreso de una sección crítica este habilitado.

l) Que es un multitarea?

Se define como la propiedad de que varios procesos compartan recursos comunes.

m) Que es un planificador?

Componente muy importante en los sistemas ya que su función consiste en repartir el tiempo disponible de un microprocesador entre todos los procesos que están disponibles para su ejecución.

n) En Thread hilo=new Thread(this); this es un objeto de una clase que \_\_\_\_\_?

Implementa el método desde la cual fue llamado

o) En (new Button()).addActionListener(this); this es un objeto de una clase que \_\_\_\_\_?

Qué es un método ...

p) de clase?

Es la lógica de una clase, es decir, contienen el código que manipula el estado del objeto.

q) de instancia

Es aquel que se invoca siempre sobre un mismo objeto de una clase.

r) abstracto

Es un método declarado pero no implementado, es decir, es un método del que sólo se escribe su nombre, parámetros y tipo devuelto pero no su código.

s) sincronizado

Es un tipo de método que permite la exclusión mutua de secciones críticas de código.

t) remoto

Objeto llamado por una clase que reside en otra.

1.- ¿Dentro de que método esta el cuerpo de un hilo?

Run

2.-¿Cuales son los estados por los que puede pasar un hilo durante su ciclo de vida?

Recién nacido, listo, ejecutable, espera, dormido, suspendido, muerto, bloqueado.

3.- ¿Como pasa un hilo del recién nacido estado al estado listo?

Llamando el método start()

4.- ¿Cómo pasa un hilo del estado listo al estado ejecutable?

Al otorgarle el recurso del procesador para que pueda ejecutarse.

5.- ¿Cómo pasa un hilo del estado ejecutable al estado en espera?

Al llamar el método wait()

6.- ¿Cómo pasa un hilo del estado ejecutable al estado dormido?

Al invocar el método sleep()

7.- ¿Cómo pasa un hilo del estado ejecutable al estado suspendido?

Al invocar el método suspend()

8.- ¿Cómo pasa un hilo del estado ejecutable al estado dormido?

Al invocar el método sleep()

9.- ¿Cómo pasa un hilo del estado ejecutable al estado muerto?

Al invocar el método stop()

10.- ¿Cómo pasa un hilo del estado en espera al estado listo?

Al ejecutarse un notify() o un notifyAll()

11.- ¿Cómo pasa un hilo del estado dormido al estado listo?

Al acabarse el tiempo de siesta

12.- ¿Cómo pasa un hilo del estado suspendido al estado listo?

Al llamarse mediante resume()

13.- ¿Cómo pasa un hilo del estado bloqueado al estado listo?

Al completar la solicitud de E/S

14.- Para que sirve el método start()?

Permite a un hilo recién nacido ser inicializado con lo cual cambia a estado “listo”

15.- Para que sirve el método stop()?

Detiene un hilo poniéndolo en estado “muerto”

16.- ¿Para qué sirve el método suspend()?

Pasa un hilo de ejecutable a suspendido. Un hilo no consume recursos en este estado.

17.- ¿Para qué sirve el método resume()?

Permite cambiar el estado de un hilo de suspendido a ejecutable

18.- ¿Para qué sirve el método wait()?

Cambia el estado de un hilo de ejecutable a en espera, pasando a esperar en un pool de hilos.

19.- ¿Para qué sirve el método notify()?

Permite cambiar el estado de un hilo de en espera a ejecutable

20.- ¿Para qué sirve el método notifyall()?

Cambia a ejecutable todos los hilos que se encontraban en espera en ese momento de la ejecución

21.- Si un hilo se lleva del estado ejecutable al estado bloqueado llamando el método suspend. ¿Cómo puede llevarse de nuevo al estado ejecutable?

Con el método resume()

22.- ¿Qué método debe llamar un hilo para pasar al estado bloqueado cierto número de milisegundos y transcurridos estos volver al estado ejecutable?

sleep()

23.- ¿Cuáles son las dos causas por las que un hilo llega al estado muerto?

Por la finalización del método run() o por la llamada del método stop()

24.- ¿Qué método se usa para finalizar la vida de un hilo?

stop()

25.- ¿Qué método coloca el primer hilo de la cola de espera de un objeto en el estado ejecutable?

notify()

27.- ¿Bajo qué circunstancias se implementa en una clase la interfaz Runnable?

Cuando la clase ya ha heredado de otros atributos

28.- ¿Qué método(s) tiene que implementar una clase que implemente la interfaz Runnable?

run()

29.- ¿En Java qué es un monitor?

Son objetos que pueden ser utilizados por más de un hilo de ejecución. Son ejecutados con exclusión mutua.

30.-Por que es necesario sincronizar el acceso a datos compartidos?

Para garantizar la integridad de los datos.

## **Red**

Que es un protocolo de comunicaciones?

Es un sistema de reglas que permiten que dos o más entidades de un sistema de comunicación se comuniquen entre ellas para transmitir información por medio de cualquier tipo de variación de una magnitud física.

32.1 En comunicaciones que es un protocolo?

Es un sistema de reglas que permiten que dos o más entidades de un sistema de comunicación se comuniquen entre ellas para transmitir información por medio de cualquier tipo de variación de una magnitud física.

32.2 Por que se usan pilas de protocolos?

Porque es una forma de mantener ordenada la colección de protocolos entrantes para poder seguir la secuencia de los mismos

32.3 -Que es TCP/IP?

Transmission Control Protocol/Internet Protocol, es el protocolo más usado en internet

-Que es una dirección IP?

Es un número que identifica, de manera lógica y jerárquica, a una Interfaz en red

-Que es un puerto?

Es una interfaz para comunicarse con un programa a través de una red.

-Que es números de puertos están reservados?

Los puertos inferiores al 1024

## **Sockets**

33.-Que es un socket?

Un socket es un punto de comunicación por el cual el proceso puede emitir o recibir información

35.-Cual es el nombre de la clase de un socket cliente?

Socket

36.-Cuales son los 2 argumentos mas usuales del constructor de un socket cliente?

El primero es la url del servidor donde se va a conectar, o en su defecto especificar que es una conexión local, y el segundo es el número de puerto al cual se va a conectar

37.-Cual es el Ciclo de Vida de un socket cliente?

1.- El nuevo socket es creado usando un constructor socket().

2.- El socket intenta conectarse a un host remoto.

3.- Una vez que la conexión es establecida los hosts local y remoto obtienen flujos de E/S para el socket y usan esos flujos para enviarse datos uno a otro. Esta conexión es full-duplex. Lo que significan los datos depende del protocolo.

4.- Cuando la transmisión de datos está completa, uno o ambos cierra la conexión.

38.-Cual es el nombre de la clase de un socket servidor?

ServerSocket

39.-Cuales es el argumento mas usual del constructor de un socket servidor?

El número de puerto sobre el cual va a operar

40.-Cual es el ciclo de vida de un socket servidor?

1.- Un ServerSocket nuevo es creado en un puerto particular usando el constructor ServerSocket()

2.- El ServerSocket() escucha los intentos de conexión entrantes con ese puerto usando su método accept(). Accept() se bloquea hasta que un cliente intenta realizar una conexión, en dicho punto accept() regresa un socket que conecta el cliente al servidor.

3.- Dependiendo del tipo de servidor el método getInputStream() o el método getOutputStream() del socket será llamado. Para obtener flujos de E/S que comunican con el cliente.

4.- El servidor y cliente interactúan de acuerdo a un protocolo establecido hasta que es tiempo de cerrar la conexión.

5.- El servidor, el cliente o ambos, cierran la conexión.

6.- El servidor regresa al paso 2 y espera la siguiente conexión.

41.-Para que sirve el método accept?

Para escuchar los intentos de conexión de los clientes.

42.-Que tipo de valor regresa el método accept y para que sirve dicho valor?

Regresa un socket que nos permitirá conocer los flujos de E/S que está realizando el cliente

43.-Que métodos se usan para obtener flujos de E/S de un socket?

Se utilizan métodos try-catch, ya que estos pueden generar excepciones si se intentan conectar en un mismo puerto

## **RMI**

44.-¿Para que sirve RMI?

Permite que objetos Java en diferentes hosts se comuniquen unos con otros.

45.- ¿Con qué se programa a más alto nivel con sockets o con RMI?

Sockets

46.- ¿Con RMI la programación está centrada en la invocación de métodos remotos, en el caso de los sockets en que se centra la programación?

En el uso de IP's y de host, los cuales crean un flujo de datos para poder comunicar al cliente y al servidor.

47.- Escriba los nombres de las capas del protocolo RMI del lado del cliente.

1. Programa Cliente
2. Stub
3. Nivel de referencia remota
4. Nivel de transporte

48.-Escriba los nombres de las capas del protocolo RMI del lado del servidor.

1. Programa Cliente
2. Esqueleto
3. Nivel de referencia remota
4. Nivel de transporte

49.- ¿Para qué sirve la capa de referencia remota?

Es utilizada por la capa Stub y skeleton, es la responsable del funcionamiento independientemente de la capa de transporte que se esté utilizando. Esta capa proporciona un objeto RemoteRef que representa la conexión con el objeto remoto. El objeto stub utiliza el método invoke() de RemoteRef para enviar las llamadas a los métodos del objeto remoto.

50.- ¿Para qué sirve la capa Stub?

Es responsable de:

- Iniciar una llamada al objeto remoto
- Ordenar los argumentos del flujo
- Informar a la capa de referencia remota que la llamada debe ser hecha
- Retornar el valor o excepción de un flujo
- Informar a la capa de referencia remota que la llamada está completa.

51.- ¿Para qué sirve la capa del esqueleto?

Es responsable de:

- Decodificar los datos del flujo de datos
- Hacer la llamada de la implementación del objeto remoto actual
- Retornar el estado de la llamada o excepción dentro de un flujo.

56.- ¿Para qué sirven los métodos bind y rebind de la clase Naming?

`bind(url, object)` Liga un nombre a un objeto remoto.

`rebind(url, object)` Similar al `bind()` pero reemplaza la asociación hecha.

57.- ¿Para qué sirve el método `lookup` de la clase `Naming`?

`lookup(url)` Retorna un objeto remoto (un stub) asociado con el url.

58.- ¿Cuál es el nombre del programa que se usa para generar los stubs y los esqueletos para los objetos remotos?

Para generar estos "Stubs" y "Skeletons" Java ofrece una herramienta llamada `rmic`

59.- ¿Cuál es el nombre del programa que se usa para registrar un objeto remoto?

`rmiregistry`

## HTTP

60.- ¿Qué es HTTP?

Hyper Text Transports Protocol, que es el protocolo de comunicación entre servidores y navegadores de web

61.- ¿Para qué sirven las formas de html?

Es un lenguaje que nos permitirá diseñar principalmente páginas web

62.- ¿Qué es CGI?

Common Gateway Interface, se usa para generar páginas web dinámicamente, principalmente usado para procesar entradas de html realizadas por el usuario

63.- ¿Cuándo se usa el método POST en una petición http?

Cuando se busca enviar la cadena de búsqueda por medio de un flujo de salida

64.- ¿Cuándo se usa el método GET en una petición http?

Cuando se enviará la cadena de consulta agregándola al url y así enviándolo al servidor

64.1.- En CGI ¿qué es una cadena de consulta?

Son parejas de nombre-valor separadas por "&" y donde el nombre se separa del valor mediante "="

65.- ¿Qué es un Servlet?

Es el equivalente a un applet, pero utilizado del lado del servidor al cuál hacen extensible

## JavaBeans

71. ¿Cuáles son las tres partes de un JavaBean?

Propiedades, Métodos y Eventos

72. ¿Qué es una propiedad de un JavaBean?

Son atributos de un Bean que son referenciadas por nombre

73. ¿Qué es una propiedad indexada?

Una propiedad indexada representa un array de valores.

74. ¿Qué es una propiedad encadenada?

Los objetos de una clase que tiene una propiedad ligada notifican a otros objetos (listeners) interesados, cuando el valor de dicha propiedad cambia, permitiendo a estos objetos realizar alguna acción. Cuando la propiedad cambia, se crea un objeto (event) que contiene información acerca de la propiedad (su nombre, el valor previo y el nuevo valor), y lo pasa a los otros objetos (listeners) interesados en el cambio.

75. ¿Qué es una propiedad restringida?

Una propiedad restringida es similar a una propiedad encadenada salvo que los objetos (listeners) a los que se les notifica el cambio del valor de la propiedad tienen la opción de vetar (veto) cualquier cambio en el valor de dicha propiedad.

76. ¿Qué es un método getter?

Es aquel método de una clase que nos proporcionará el valor o el conjunto de datos que contenga ésta sin necesidad de invocarlos directamente, manteniendo así la propiedad de encapsulamiento de la clase.

77. ¿Qué es un método setter?

Es aquel método de una clase que nos permitirá modificar el valor o el conjunto de datos que contenga ésta sin necesidad de invocarlos directamente, manteniendo así la propiedad de encapsulamiento de la clase.

78.- ¿Por medio de qué objetos se comunican los JavaBeans?

Los eventos son el medio de comunicación entre los Beans

79.-Que es el mecanismo de reflexión

Es el proceso que nos permite analizar la clase de un Bean para determinar sus métodos

80.-Que es el mecanismo de introspección

Es el proceso de exponer las propiedades, los métodos y eventos que soporta un componente JavaBean

## **JDBC**

Que es un SGBD?

Es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información de una base de datos, además, proporciona herramientas para añadir, borrar, modificar y analizar los datos

Que es una tabla?

Relaciones en el sentido matemático del término

Que es un registro?

Filas o tuplas, es la instanciación de una tupla

Que es un campo?

Es la unidad mínima de información en la base de datos

Que es un metadato?

Datos sobre los datos, representa la estructura y relaciones del mismo.

Que es un diccionario de datos?

Recoge todos los nombres de los objetos (tablas, vistas, alias...) que contiene una base de datos

Que significan las siglas CRUD

Create, Read, Update, Delete

Que significan las siglas SQL

Structure Query Language

Para que sirve SQL

Es el lenguaje estándar para los SGBD Relacionales

En SQL que es resultado de una consulta

Son los registros que cumplen las determinadas restricciones de la consulta

81.-¿Que es ODBC?

Oracle Data Base Connectivity

82.-¿Qué es JDBC?

Java Data Base Connectivity, una api para ejecutar frases SQL, posibilita al programador el acceso a la base de datos local y remota mediante una interfaz

83.-¿Cuáles son las partes de URL en JDBC?

Protocolo:subprotocolo://nombre:dominio/

84.-¿Cómo se realiza la conexión con JDBC a una Base de Datos?

Se insertará el comando DriverManager.getConnection(url+dbName,"root","root"), donde la url es el sitio a donde se busca conectar, y seguido va el nombre de login plus el password a la base de datos

85.-¿Cómo se envía una instrucción de SQL a la Base de Datos?

La instrucción se enviará por medio de una línea de texto, comúnmente guardada en un string, y por medio del siguiente comando

```
Statement stmt=conex.createStatement();
```

```
stmt.executeQuery(sql);
```

86.-¿Cómo se procesan los resultados de una instrucción de SQL?

```
ResultSet result = stmt.executeQuery(sql);
```

```
displayResults(result);
```

Donde result nos traerá el resultado de la consulta, y displayResult lo mostrará en pantalla.

87.-¿Para que sirve la clase DriverManager?

Para poder acceder a la base de datos

88.-¿Para que sirve la clase Statement?

Para administrar la base de datos por medio de consultas o actualizaciones a la misma

89.-¿Para que sirve la clase ResultSet?

Nos permitirá conocer el resultado de las diferentes acciones que realicemos sobre la base de datos