

ACTIVIDAD 1 OBJETIVOS: Caracterizar los mecanismos de comunicación.

Responda brevemente a las siguientes preguntas (con un máximo de **50 palabras** por respuesta):

- a) ¿Qué diferencias existen entre los mecanismos que utilizan primitivas básicas de comunicación y los que utilizan construcciones del lenguaje de programación?
- b) ¿Cómo puede ser la estructura de los mensajes en un mecanismo de comunicación cualquiera?
- c) En los mecanismos de comunicación el contenido de los mensajes puede ser en bytes o bien en texto. ¿Qué ventajas presentan los mensajes de texto frente a los mensajes en bytes?
- d) ¿Qué diferencias existen entre el direccionamiento directo y el indirecto?
- e) ¿Qué tipos de sincronización existen en los mecanismos de comunicación? ¿Podrías dar ejemplos de cada uno?
- f) ¿Qué diferencias existen entre comunicación persistente y no persistente? ¿Podrías dar ejemplos de mecanismos de comunicación para cada tipo de persistencia?

ACTIVIDAD 2 OBJETIVOS: Clarificar conceptos sobre mecanismos de comunicación.

Explique brevemente los siguientes conceptos (con un máximo de **50 palabras** cada uno):

- a) Sockets
- b) Servicios web
- c) Colas de mensajes
- d) RPC
- e) XML
- f) JSON
- g) Serialización

ACTIVIDAD 3 OBJETIVOS: Caracterizar el mecanismo de Invocación a Objeto Remoto (ROI).

Ordene, según el orden en que tienen lugar, los siguientes pasos de una ROI. ¿Falta algún paso para que todo el mecanismo ROI funcione? Si es así, descríbalos.

- A. El método llamado finaliza y se desbloquea el esqueleto
- B. El proxy desempaqueta los resultados y los devuelve al proceso cliente
- C. El proceso cliente invoca el método del proxy local relacionado con el objeto remoto

- D. El esqueleto empaqueta los resultados y llama al ORB, el cual hace llegar el mensaje al proxy
- E. El proxy empaqueta los argumentos y, utilizando la referencia al objeto, llama al ORB
- F. El ORB gestiona la invocación, haciendo que el mensaje llegue al esqueleto.

ACTIVIDAD 4 OBJETIVOS: Caracterizar el mecanismo de Invocación a Objeto Remoto (ROI).

Sobre el paso de argumentos en el mecanismo de comunicación ROI, indique si las siguientes afirmaciones son Verdaderas (V) o falsas (F), justificando su respuesta:

	En un paso de parámetros por referencia, ésta puede pertenecer a uno nodo que no sea ni el invocador ni el invocado. Justificación:
	Los argumentos se pueden pasar por valor, no sólo mediante referencias a objetos. Justificación:
	Los argumentos que se pasan por referencia se serializan antes de transmitirlos al nodo destino. Justificación:
	En un paso de parámetros por referencia, ésta puede pertenecer al nodo invocador. Justificación:

ACTIVIDAD 5 OBJETIVOS: Caracterizar el mecanismo de Invocación a Objeto Remoto (ROI).

En el mecanismo ROI, la creación de objetos puede realizarse mediante dos procedimientos distintos. ¿Cuáles son? Indique, para los siguientes pasos, a qué procedimiento se corresponde y en qué orden tienen lugar (Nota: un mismo paso puede pertenecer a los dos procedimientos).

- A. El servidor obtiene una referencia al objeto.
- B. Un proceso (cliente) solicita a un servidor (factoría) que cree un determinado objeto.
- C. El proceso servidor usa la referencia del objeto para registrarlo en un servidor de nombres, proporcionando una cadena de texto como nombre del objeto.
- D. El servidor (factoría) devuelve al cliente una copia de la referencia del objeto que ha solicitado crear.

- E. Un proceso que conozca el nombre utilizado para registrar el objeto contacta con el servidor de nombres y obtiene una referencia al objeto.
- F. Un proceso crea un objeto y lo registra en el ORB.
- G. El servidor crea el objeto (que le han solicitado crear) y lo registra en el ORB.

ACTIVIDAD 6 OBJETIVOS: Caracterizar el mecanismo de comunicación Java RMI.

Indique si las siguientes afirmaciones son verdaderas (V) o falsas (F). Justifique su respuesta.

	Los objetos remotos deben residir en la misma JVM. Justificación:
	Java construye automáticamente los esqueletos y los proxies a partir de la especificación de la interfaz del objeto remoto . Justificación:
	Todos los objetos que se pasan como argumentos en Java RMI deben ser remotos, no permitiéndose por tanto pasar objetos locales. Justificación:
	El servidor de nombres de Java RMI almacena, para cada objeto registrado, su nombre y esqueleto. Justificación:
	El mecanismo de comunicación de Java RMI no tiene nada que ver con el mecanismo de comunicación ROI (invocación a objeto remoto). Justificación:

ACTIVIDAD 7 OBJETIVOS: Caracterizar el mecanismo de comunicación Java RMI.

Dada la siguiente aplicación:

```
public interface Hola {
    String saluda();
}
class ImpleHola implements Hola {
    ImpleHola() {...} // constructor
    public String saluda() {return "Hola a todos";}
}
...
Hola h = new ImpleHola();
System.out.println(h.saluda());
```

Se desea implementarla como una aplicación Java RMI. Indique brevemente todos los pasos que se deben realizar, tanto para el desarrollo del cliente como para el desarrollo del servidor.

ACTIVIDAD 8 OBJETIVOS: Caracterizar el mecanismo de comunicación Java RMI.

Se desea crear un servicio básico de *eco*, que ofrece únicamente un método remoto que retorna la cadena de caracteres recibida como argumento, pero pasándola a mayúsculas.

1.- Actualice la siguiente definición del servicio “eco”, para que pueda ser utilizado de forma remota.

```
import java.rmi.*;

interface ServicioEco {
    String eco (String s)
}
```

2.- Actualice la clase *ServicioEcoImpl* (que se muestra a continuación) para que implemente el servicio remoto.

```
import java.rmi.*;

class ServicioEcoImpl implements ServicioEco {
    public String eco (String s){
        return s.toUpperCase();
    }
}
```

3.- En la clase *ServidorEco*, que actúa como servidor, indique cómo se inicia el servicio remoto y cómo se hace accesible usando *rmiregistry*. ¿Con qué nombre se ha registrado el servicio?

```
import java.rmi.*;
import java.rmi.server.*;

class ServidorEco{
    static public void main (String args[]){
        if (args.length!=1) {
            System.err.println("Uso: ServidorEco numPuertoRegistro");
            return;
        }

        try {
            ServicioEcoImpl srv = new ServicioEcoImpl();

            Registry reg = LocateRegistry.getRegistry("localhost",
                1099);
```

```

        ServicioEco srv= reg.rebind("Eco", srv);
    }
    catch (RemoteException e) {
        System.err.println("Error de comunicacion: " +
            e.toString());
        System.exit(1);
    }
    catch (Exception e) {
        System.err.println("Excepcion en ServidorEco:");
        e.printStackTrace();
        System.exit(1);
    }
}
}

```

4. Finalmente, se muestra a continuación el código del cliente (fichero ClienteEco.java). Actualice dicho código para que el cliente obtenga una referencia remota asociada al servicio “eco” que hemos implementado.

```

import java.rmi.*;
class ClienteEco {
    static public void main (String args[]) {
        if (args.length<2) {
            System.err.println("Uso: ClienteEco hostregistro
                numPuertoRegistro ...");
            return;
        }

        try {

            for (int i=2; i<args.length; i++)
                System.out.println(srv.eco(args[i]));
        } catch (Exception e) {
            System.err.println("Excepcion en ClienteEco:");
            e.printStackTrace();
        }
    }
}

```

ACTIVIDAD 9 OBJETIVOS: Caracterizar las referencias a recursos en REST

Para las siguientes llamadas, indique:

- Si siguen el estándar REST o no. En caso de no ser REST, realice los cambios necesarios para que sean REST.
- ¿Qué es lo que pretende realizar esa llamada?

- 1) GET `https://api.github.com?type=users&id=captainkidd`
- 2) GET `https://api.githum.com/users/captainkidd/edit`
- 3) GET `https://api.githum.com/gists/page/22/per_page/2`
- 4) POST `https://weatherapp.com/messages`
- 5) GET `weatherapp.com/wheaterLookup.do?zipcode=46017`
- 6) GET `weatherapp.com/getMessages.do?id=10`
- 7) GET `https://myapp.com/deleteOrder.do?id=10`
- 8) DELETE `https://myapp.com/messages/10`
- 9) GET `https://myapp.com/messages`

ACTIVIDAD 10 OBJETIVOS: Caracterizar las operaciones en REST

Dadas las siguientes URIs, indique el resultado que se obtiene al aplicar sobre ellas los métodos HTTP indicados:

Método	URI	Resultado
GET	/messages	
POST		
GET	/messages/10	
PUT		
DELETE		
GET	/messages/10/comments	
DELETE		
POST		
PUT		

ACTIVIDAD 11 OBJETIVOS: Caracterizar el mecanismo de comunicación JMS

ENUNCIADO: Dadas las siguientes afirmaciones, justifique si son Verdaderas (V) o Falsas (F):

1. Generalmente es preferible usar JMS frente a Java RMI cuando es necesario que todos los componentes de la aplicación estén simultáneamente en ejecución. <i>JUSTIFICACIÓN:</i>	
2. La comunicación se considera débilmente acoplada. <i>JUSTIFICACIÓN:</i>	
3. Un cliente JMS es un objeto administrado. <i>JUSTIFICACIÓN:</i>	
4. Las colas de mensajes se crean normalmente utilizando las herramientas administrativas del proveedor JMS. <i>JUSTIFICACIÓN:</i>	
5. Un proveedor JMS es una empresa que ofrece servicios de consultoría relativos a JMS. <i>JUSTIFICACIÓN:</i>	
6. Los objetos que implementan la interfaz Queue se crean llamando a métodos de la interfaz JMSConsumer. <i>JUSTIFICACIÓN:</i>	
7. Los objetos que implementan la interfaz JMSProducer se crean llamando a métodos de la interfaz JMSContext. <i>JUSTIFICACIÓN:</i>	
8. La comunicación normalmente es persistente. <i>JUSTIFICACIÓN:</i>	
9. La comunicación es sincrónica en la respuesta. <i>JUSTIFICACIÓN:</i>	
10. El direccionamiento empleado es del tipo directo. <i>JUSTIFICACIÓN:</i>	

ACTIVIDAD 12 OBJETIVOS: Caracterizar los diferentes mecanismos de comunicación.

Complete la siguiente tabla sobre características de los mecanismos de comunicación.

Mecanismo	Direccionamiento	Sincronización	Persistencia (Sí / No)
ROI			
Java RMI			
Servicios Web RESTful			
Java Message Service			