

Ejercicios temas 3 y 4

1. Realizar un programa que muestre por pantalla:
 - a. La dirección IP de una URL por nombre.
 - b. Nombre a partir de la dirección.
 - c. Dirección IP actual de LocalHost.
 - d. Nombre de LocalHost a partir de la dirección.
 - e. Nombre actual del LocalHost.
2. Definir un objeto InetAddress de nombre dir y realizar en un mismo programa lo siguiente:
 - a. Utilizar este objeto para obtener la dirección IP de la máquina local en la que se ejecuta el programa llamando a un método pruebaMetodos() llevando el objeto creado y probar en él los métodos de la clase InetAddress:
 - getByName(String host)
 - getLocalHost()
 - getHostName()
 - getHostAddress()
 - toString()
 - getCanonicalHostName()
 - b. Hacer lo mismo para obtener la dirección Ip de la URL www.google.es.
3. Realizar un proyecto cliente-servidor en el que el cliente muestre su dirección y le mande al servidor un número y el servidor le devuelve su cuadrado. El servidor mostrará en su pantalla su LocalHost e IP, así como la dirección y el puerto del cliente, el número de entrada y el resultado. El servidor permanecerá activo y el cliente se cerrará una vez que vea el resultado.
4. Realizar un proyecto cliente-servidor en el que el cliente le mande al servidor un número y el servidor le devuelve su factorial. Tanto el servidor como el cliente finalizarán cuando el número sea un 0.
5. Realizar un proyecto cliente-servidor en el que el cliente le mande al servidor dos números y el servidor le devuelve su m.c.d. Tanto el servidor como el cliente finalizarán cuando el primer número sea un 0.
6. Realizar un proyecto cliente-servidor en el que el cliente tenga que averiguar un número aleatorio generado por el servidor entre 1 y 1000; el servidor informará al cliente si el número generado es mayor o menor que el introducido por el cliente. El cliente finalizará cuando introduzca un cero. El servidor seguirá activo.
7. Realizar un proyecto cliente-servidor en el que el cliente elija en un menú la operación que quiere realizar de las cuatro elementales. Luego le enviará al servidor dos números y el servidor le devolverá el resultado. La conexión seguirá en marcha mientras el cliente no elija la quinta opción del menú que será Salir.
8. Realizar un proyecto cliente-servidor en el que el servidor haga eco de lo que dice el cliente hasta que el cliente le envíe adiós en cuyo caso se cerrará el cliente y el servidor seguirá activo.
9. Realizar un proyecto cliente-servidor de un chat en el que se cerrará el cliente cuando éste diga adiós y el servidor seguirá activo.
10. Realizar un proyecto cliente-servidor en el que el servidor deberá leer un fichero de texto del disco y enviarlo al cliente que lo imprimirá por su salida estándar. Utilizar para ello *sockets stream*.