

# **SATHYABAMA UNIVERSITY**

**(Established under Section 3, UGC Act 1956)**

## ***DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING***



**SCSX4012 NETWORK PROGRAMMING LAB**

## SCSX4012 : NETWORK PROGRAMMING LAB

### List of experiments

1. A simple Program to print the IP address of the system
2. Creation of Date and Time Server.
3. Printing Client address at Server side.
4. Creation of a simple Chat program.
5. Session Tracking in Servlet.
6. Accessing Database in a Servlet.
7. Servlet to Applet communication.
8. Application combining HTML, Javascript , and Servlet.
9. Program to implement HTTP Protocol
10. Create a servlet program in web application
11. Program to display all the cookie information using this session lifetime.
12. Create a simple program to send mail using java mail.
13. A Simple Program to implement FTP using TCP
14. A Simple Program using JSP.

**EX NO: 1      A SIMPLE PROGRAM TO PRINT THE IP ADDRESS OF THE SYSTEM**

**AIM:** To create a simple program to print the IP address of the system

**ALGORITHM**

**STEP1:** Import the java.net package

**STEP2:** Create instance of the InetAddress class and capture the IP address of the local host by invoking the getLocalHost() method .

**STEP3:**Print the IP address of the local host using getHostAddress() method of the InetAddress class.

**EX NO: 2      CREATION OF DATE AND TIME SERVER**

**AIM:** To create a network program for creating date and time server

**ALGORITHM**

**SERVER**

**STEP1:** Create instances for socket and ServerSocket class.

**STEP2:** Use the port 8020 for TCP.

**STEP3:** Make the PrintStream object connect to the OutputStream using Socket.

**STEP4:** Create an instance of the Date class and write it into the Socket.

**STEP5:** Get the IP address of the client using the socket and getInetAddress().

**CLIENT**

**STEP1:** Create instances for socket class with the port number 8020.

**STEP2:** Create an object of DataInputStream and make it to get data from server through the socket.

**STEP3:** Read the Date object.

**STEP4:** Print the obtained date.

### **EX NO: 3      PRINTING CLIENT ADDRESS IN SERVER SIDE**

**AIM:** To print the client address in the server side

#### **ALGORITHM**

##### **SERVER**

**STEP1:** Create instances for socket and ServerSocket class.

**STEP2:** Use the port 9000 for TCP.

**STEP3:** Make the PrintStream object connect to the OutputStream using Socket.

**STEP4:** Create an instance of the Date class and write it into the Socket.

**STEP5:** Get the IP address of the client using the socket and  
getInetAddress ().

**STEP6:** Print the client's IPAddress.

##### **CLIENT**

**STEP1:** Create instances for socket class with the port number 9000.

**STEP2:** Create an object of DataInputStream and make it to get data from  
server through the socket.

**STEP3:** Read the Date object.

**STEP4:** Print the obtained date.

### **EX NO: 4      A SIMPLE CHAT PROGRAM**

**AIM:** To create a simple chat application

#### **ALGORITHM**

##### **SERVER**

**STEP1:** Instances of vector class is used to keep track of number of clients  
that can be connected and currently logged.

**STEP2:** The method that is responsible for sending the message to the  
clients is made synchronized.

**STEP3:** Server is capable of keeping into account the number of users. It  
adds and removes the client from the vector list as and when the  
connections are established and terminated.

##### **CLIENT**

**STEP1:** The client receives the name of the user and message of that user  
and sends it to client. Server then passes it on to all clients connected.

**EX. NO. 5****FTP USING TCP**

**AIM:** To transfer a file from the server to the client

**ALGORITHM****CLIENT**

**STEP 1:** Create instance for the Socket class and establish connectivity with the server

**STEP 2:** Use the port number 4000

**STEP 3:** Receive the file from the server

**STEP 4:** Reset the connection with the server

**SERVER**

**STEP 1:** Create instances for the serversocket class and accept the server port

**STEP 2:** Read the filename to be opened

**STEP 3:** Send the file to the client

**EX. NO. 6****SIMPLE WEB APPLICATION USING SERVLET**

**AIM:** To prompt the user for name and email id and print a welcome message using servlet

**PROCEDURE:**

**STEP 1:** Create the HTML page with form tag to collect the name and mail\_id of the user

**STEP 2:** Create the servlet to process the request and generate the HTML to display greeting message.

**EX. NO. 7****SESSION LIFETIME USING COOKIES**

**AIM:** To display all the cookie information using this session lifetime

**PROCEDURE:**

**STEP 1:** Generate the web page showing sessionstatus, sessionID, creation time, last accessed time , maximum inactive interval using the methods present in HttpSession interface

**STEP 2:** When the servlet is accessed for the first time, a new session is created

**STEP 3:** Display the servlet session information using the corresponding methods

**STEP 4:** On clicking the invalidate session link, the current session is terminated and a new session is generated.

**STEP 5:** The page can be reloaded by clicking the 'reload' link.

## **EX. NO. 8           SESSION TRACKING IN SERVLET**

**AIM:** To develop a shopping cart using session

### **PROCEDURE:**

#### **CATALOG SERVLET**

- STEP 1:** Obtain the number of books form the data stored in HttpSession using request.getSession() method
- STEP 2:** HttpSession attribute cart is used to find and display the number of books in the cart
- STEP 3:** Shopping cart servlet stores and updates this attribute when items are added to the cart
- STEP 4:** On clicking the invalidate session link, the current session is terminated and a new session is generated.
- STEP 5:** The page can be reloaded by clicking the 'reload' link.

#### **SHOPPINGCART SERVLET**

- STEP 1:** Collects the list of books selected
- STEP 2:** Updates the shopping cart

## **EX. NO. 9    ACCESSING DATABASE IN A SERVLET**

**AIM:** To access the database information using servlet

### **PROCEDURE:**

- STEP 1:** Import the servlet package and other java packages
- STEP 2:** Create a database like student or employee
- STEP 3:** Obtain the database connection using JNDI lookup to the data source
- STEP 4:** Extract the data from the database using getParameter() method
- STEP 5:** Prepare the statements for insert
- STEP 6:** Set parameters for the prepared statement
- STEP 7:** Execute the prepared statement
- STEP 8:** Run another statement to get back the request id
- STEP 9:** Close the database connection in the finally block
- STEP 10:** Prepare the response

**EX. NO. 10****JAVA MAIL**

**AIM:** To send mail using java mail

**PROCEDURE:**

**STEP 1:** Set up the parameters like smtp server, to, from, email body etc., of javamail class

**STEP 2:** Create session and create a new mail message

**STEP 3:** Set from, to, date and subject field

**STEP 4:** Create the body of email

**STEP 5:** Using Transport. send(msg) deliver the message to the recipient

**EX. NO. 11****SERVLET APPLET COMMUNICATION**

**AIM:** To communicate between an applet and a servlet

**PROCEDURE:****APPLET:**

**STEP 1:** Create the applet class 'Myapplet'

**STEP 2:** Initialize the text fields t1,t2, t3 and button b

**STEP 3:** On clicking the button b ,and perform the the following action

**STEP 3.1:** Connect to the Servlet

**STEP 3.2:** Get the result from the servlet and store it in the textbox t3

**SERVLET:**

**STEP 1:** Create the servlet and using the request.getQueryString() method, receive the applet request

**STEP 2:** Tokenize the queries and calculate the result

**STEP 3:** Send the response to the servlet

**EX. NO. 12****JAVA SERVER PAGES**

**AIM:** To create a login page and check whether the user is a valid user or not using JSP.

**PROCEDURE:**

**STEP 1:** Create a HTML file to accept the user name and mail id using action attribute of the form tag

**STEP 2:** Create a jsp file 'forward.jsp' and check whether the user is a valid user or not

**STEP 2.1:** If the user is a valid user means the 'welcome.jsp' file is invoked displaying a welcome message

**STEP 2.2:** Else the same page is reloaded.