

CSE3011 NETWORK PROGRAMMING

LAB EXPERIMENT 4

NAME – B PRATYUSH

REGISTRATION NUMBER – 19BCN7114

LAB SLOT – L1+L2

FACULTY – PROF. MUNEESWARI

Experiment Description: IP Characteristics and URL Content Extraction

CODE 1

URLProg.java

```
package Lab4;

import java.net.*;
import java.util.*;
import java.io.*;

public class URLProg {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        System.out.println("URL Info extraction");
        System.out.println("How many URL's going to be entered");
        Scanner sin=new Scanner(System.in);
        int n=sin.nextInt();
        for(int i=0;i<n;i++)
        {
```

```

        try
    {
        System.out.println("Enter URL");

        String urlin = sin.next();

        URL u =new URL(urlin);

        System.out.println("Entered URL is :"+u);

        System.out.println("The Scheme/Protocol used is:
"+u.getProtocol());

        System.out.println("The User info is: "+u.getUserInfo());

        String host = u.getHost();

        System.out.println("The host is: "+host);

        int atSign = host.indexOf('@');

        if (atSign != -1)
        {
            host = host.substring(atSign+1);

            System.out.println("The host is " + host);
        }

        else
        {
            System.out.println("The host is null.");
        }

        System.out.println("The port number is " + u.getPort());

        System.out.println("The path is " + u.getPath());

        System.out.println("The anchor of the URL is " + u.getRef());

        System.out.println("The query string is " + u.getQuery());

        System.out.println("The Hashcode value is " + u.hashCode());

        System.out.println("The URL CONTENT: "+u.getContent());
    }

```

```
System.out.println("File Name: "+u.getFile());
```

```
}
```

```
catch(MalformedURLException e)
```

```
{
```

```
    System.out.println("Cannot understand the entered URL");
```

```
    System.out.println(e);
```

```
}
```

```
    catch(IOException e)
```

```
    {
```

```
        System.out.println(e);
```

```
    }
```

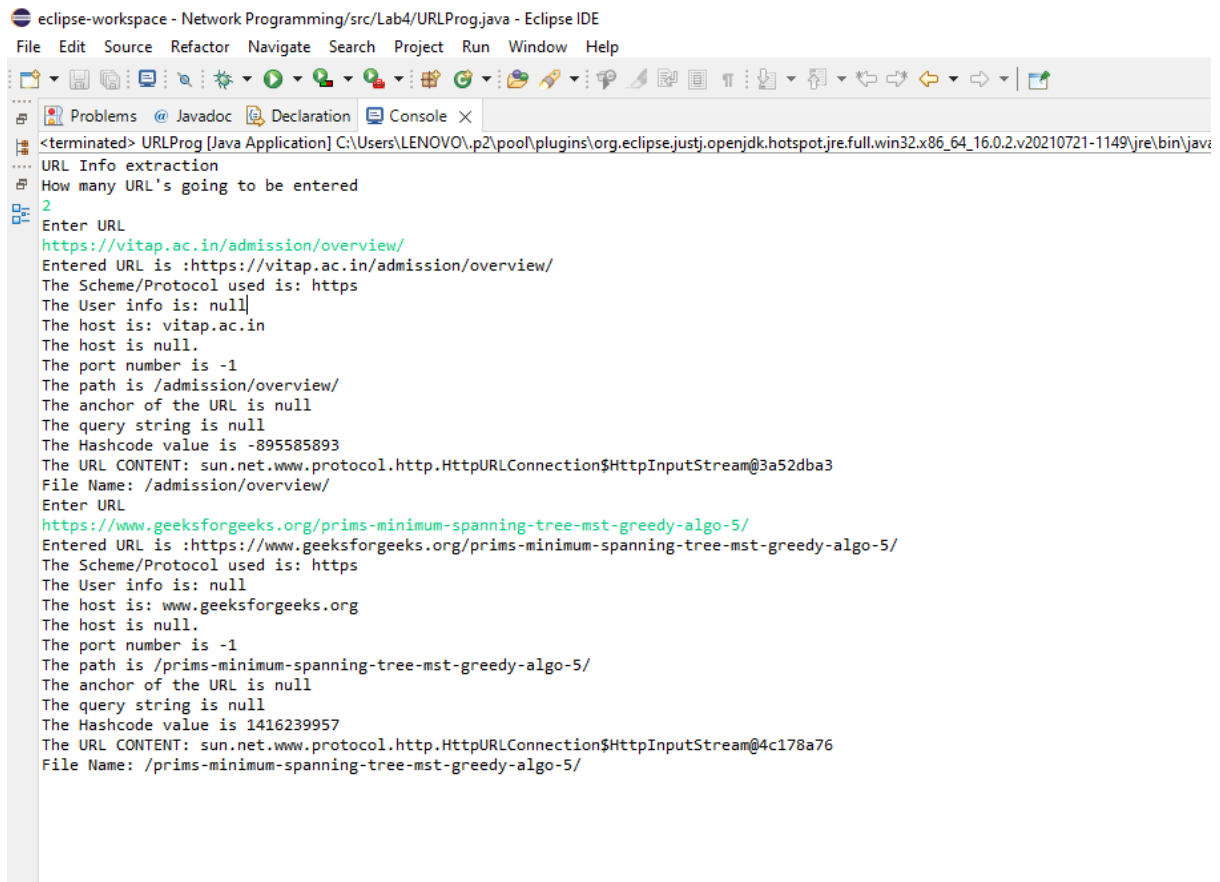
```
}
```

```
sin.close();
```

```
}
```

```
}
```

Output

The screenshot shows the Eclipse IDE interface. The title bar reads 'eclipse-workspace - Network Programming/src/Lab4/URLProg.java - Eclipse IDE'. The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar contains various icons for file operations, running, and debugging. The 'Console' tab is active, displaying the output of the 'URLProg [Java Application]' program. The output shows the program asking for the number of URLs to enter (2), then processing two URLs: 'https://vitap.ac.in/admission/overview/' and 'https://www.geeksforgeeks.org/prims-minimum-spanning-tree-mst-greedy-algo-5/'. For each URL, it displays details such as the scheme (https), host (vitap.ac.in and www.geeksforgeeks.org), path, anchor, query string, hashcode, and the full URL content. The console output is as follows:

```
<terminated> URLProg [Java Application] C:\Users\LENOVO\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.2.v20210721-1149\jre\bin\java
.... URL Info extraction
How many URL's going to be entered
2
Enter URL
https://vitap.ac.in/admission/overview/
Entered URL is :https://vitap.ac.in/admission/overview/
The Scheme/Protocol used is: https
The User info is: null
The host is: vitap.ac.in
The host is null.
The port number is -1
The path is /admission/overview/
The anchor of the URL is null
The query string is null
The Hashcode value is -895585893
The URL CONTENT: sun.net.www.protocol.http.HttpURLConnection$HttpInputStream@3a52dba3
File Name: /admission/overview/
Enter URL
https://www.geeksforgeeks.org/prims-minimum-spanning-tree-mst-greedy-algo-5/
Entered URL is :https://www.geeksforgeeks.org/prims-minimum-spanning-tree-mst-greedy-algo-5/
The Scheme/Protocol used is: https
The User info is: null
The host is: www.geeksforgeeks.org
The host is null.
The port number is -1
The path is /prims-minimum-spanning-tree-mst-greedy-algo-5/
The anchor of the URL is null
The query string is null
The Hashcode value is 1416239957
The URL CONTENT: sun.net.www.protocol.http.HttpURLConnection$HttpInputStream@4c178a76
File Name: /prims-minimum-spanning-tree-mst-greedy-algo-5/
```

Code 2

IPCharac.java

```
package Lab4;
```

```
import java.net.InetAddress;
```

```
import java.net.*;
```

```
import java.util.Scanner;
```

```
import java.io.*;
```

```
public class IPCharac {
```

```

public static void main(String[] args) {
    // TODO Auto-generated method stub
    System.out.println("Ip characteristics");
    System.out.println("Enter number of IP s");
    Scanner sin=new Scanner(System.in);
    int n=sin.nextInt();
    for(int i=0;i<n;i++)
    {
        try
        {
            System.out.println("Enter IP");
            String ipadd = sin.next();
            InetAddress address = InetAddress.getByName(ipadd);
            System.out.println("Host Name of IP
:"+address.getHostName());
            System.out.println("Canonical Host Name of IP
:"+address.getCanonicalHostName());
            System.out.println("Loopback addres of IP
:"+address.getLoopbackAddress());
            System.out.println("Is the ip address reachable
:"+address.isReachable(300));

            if (address.isAnyLocalAddress())
            {
                System.out.println(address + " is a wildcard
address");
            }
            if (address.isLoopbackAddress())

```

```

        {
            System.out.println(address + " is loopback
address.");

        }
        if (address.isLinkLocalAddress()) {
            System.out.println(address + " is a link-local
address.");

        }
        else if (address.isSiteLocalAddress())
        {
            System.out.println(address + " is a site-local
address.");

        }
        else
        {
            System.out.println(address + " is a global address.");
        }
        if (address.isMulticastAddress())
        {
            if (address.isMCGlobal())
            {
                System.out.println(address + " is a global multicast
address.");

            }
            else if (address.isMCOrgLocal())
            {
                System.out.println(address + " is an organization
wide multicast address.");
            }
        }
    }
}

```

```

    }
    else if (address.isMCSiteLocal())
    {
        System.out.println(address + " is a site wide
multicast address.");
    }
    else if (address.isMCLinkLocal())
    {
        System.out.println(address + " is a subnet wide
multicast address.");
    }
    else if (address.isMCNodeLocal())
    {
        System.out.println(address+ " is an interface-
local multicast address.");
    }
    else
    {
        System.out.println(address + " is an unknown
multicast address type.");
    }
}
else
{
    System.out.println(address + " is a unicast
address.");
}

```

```

    }

    catch(UnknownHostException e)
    {
        System.out.println("Cannot understand the entered IP");
        System.out.println(e);
    }

    catch(IOException e)
    {
        System.out.println(e);
    }
}

sin.close();
}

}

```

Output

The screenshot shows the Eclipse IDE interface with the console window open. The title bar reads "eclipse-workspace - Network Programming/src/Lab4/IPCharac.java - Eclipse IDE". The menu bar includes File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The toolbar contains various icons for file operations, running, and debugging. The console window shows the following output:

```

<terminated> IPCharac [Java Application] C:\Users\LENOVO\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.2
....
Ip characteristics
Enter number of IP s
2
Enter IP
127.0.0.1
Host Name of IP :127.0.0.1
Canonical Host Name of IP :127.0.0.1
Loopback address of IP :localhost/127.0.0.1
Is the ip address reachable :true
127.0.0.1/127.0.0.1 is loopback address.
127.0.0.1/127.0.0.1 is a global address.
127.0.0.1/127.0.0.1 is a unicast address.
Enter IP
78.47.226.171
Host Name of IP :static.171.226.47.78.clients.your-server.de
Canonical Host Name of IP :static.171.226.47.78.clients.your-server.de
Loopback address of IP :localhost/127.0.0.1
Is the ip address reachable :true
static.171.226.47.78.clients.your-server.de/78.47.226.171 is a global address.
static.171.226.47.78.clients.your-server.de/78.47.226.171 is a unicast address.

```


Code 3

IPUrlProg.java

```
package Lab4;

import java.net.*;
import java.util.*;
import java.io.*;

public class IPUrlProg {

    public static void main(String args[]) {

        System.out.println("URL and IP Info extraction");

        System.out.println("Enter number of url addresses");

        Scanner sin=new Scanner(System.in);

        int n=sin.nextInt();

        for(int i=0;i<n;i++)

        {

            try

            {

                System.out.println("Enter IP");

                String url = sin.next();

                URL u = new URL(url);

                System.out.println("Entered URL is :"+u);

                System.out.println("The Scheme/Protocol used is: "+u.getProtocol());

                System.out.println("The User info is: "+u.getUserInfo());

                String host = u.getHost();

                System.out.println("The host is: "+host);

                int atSign = host.indexOf('@');

                if (atSign != -1)

                {

                    host = host.substring(atSign+1);
```

```
        System.out.println("The host is " + host);
    }
    else
    {
        System.out.println("The host is null.");
    }
}
```

```
System.out.println("The port number is " + u.getPort());
System.out.println("The path is " + u.getPath());
System.out.println("The anchor of the URL is " + u.getRef());
System.out.println("The query string is " + u.getQuery());
System.out.println("The Hashcode value is " + u.hashCode());
System.out.println("The URL CONTENT: "+u.getContent());
System.out.println("File Name: "+u.getFile());
System.out.println();
```

```
System.out.println("Passing the url host into getByName method to
get IP address to obtain further IP characteristics");
```

```
System.out.println();
```

```
        InetAddress address = InetAddress.getByName(host);

        System.out.println("Host Name of IP
:"+address.getHostName());

        System.out.println("Canonical Host Name of IP
:"+address.getCanonicalHostName());

        System.out.println("Loopback address of IP
:"+address.getLoopbackAddress());

        System.out.println("Is the ip address reachable
:"+address.isReachable(300));
```

```

        if (address.isAnyLocalAddress())
        {
            System.out.println(address + " is a wildcard
address.");

        }
        if (address.isLoopbackAddress())
        {
            System.out.println(address + " is loopback
address.");

        }
        if (address.isLinkLocalAddress()) {
            System.out.println(address + " is a link-local
address.");

        }
        else if (address.isSiteLocalAddress())
        {
            System.out.println(address + " is a site-local
address.");

        }
        else
        {
            System.out.println(address + " is a global
address.");

        }
        if (address.isMulticastAddress())
        {

```

```

        multicast address.");

        organization wide multicast address.");

        multicast address.");

        wide multicast address.");

        interface-local multicast address.");

        unknown multicast address type.");

        if (address.isMCGlobal())
        {
            System.out.println(address + " is a global

        }

        else if (address.isMCOrgLocal())
        {
            System.out.println(address + " is an

        }

        else if (address.isMCSiteLocal())
        {
            System.out.println(address + " is a site wide

        }

        else if (address.isMCLinkLocal())
        {
            System.out.println(address + " is a subnet

        }

        else if (address.isMCNodeLocal())
        {
            System.out.println(address+ " is an

        }

        else
        {
            System.out.println(address + " is an

        }

    }
}

```

```

        else
        {
            System.out.println(address + " is a unicast
address.");
        }

    }

    catch(MalformedURLException e)
    {
        System.out.println("Cannot understand the entered URL");
        System.out.println(e);
    }

    catch(IOException e)
    {
        System.out.println(e);
    }

}

sin.close();

}

}

```

Output

```
eclipse-workspace - Network Programming/src/Lab4/IPUrlProg.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

<terminated> IPUrlProg [Java Application] C:\Users\LENOVO\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.2.v20210
....
URL and IP Info extraction
Enter number of url addresses
2
Enter IP
https://vitap.ac.in/admission/overview/
Entered URL is :https://vitap.ac.in/admission/overview/
The Scheme/Protocol used is: https
The User info is: null
The host is: vitap.ac.in
The host is null.
The port number is -1
The path is /admission/overview/
The anchor of the URL is null
The query string is null
The Hashcode value is -895585893
The URL CONTENT: sun.net.www.protocol.http.HttpURLConnection$HttpInputStream@3a52dba3
File Name: /admission/overview/

Passing the url host into getByName method to get IP address to obtain further IP characteristics

Host Name of IP :vitap.ac.in
Canonical Host Name of IP :static.52.36.9.5.clients.your-server.de
Loopback address of IP :localhost/127.0.0.1
Is the ip address reachable :true
vitap.ac.in/5.9.36.52 is a global address.
vitap.ac.in/5.9.36.52 is a unicast address.
Enter IP
https://www.geeksforgeeks.org/prims-minimum-spanning-tree-mst-greedy-algo-5/
Entered URL is :https://www.geeksforgeeks.org/prims-minimum-spanning-tree-mst-greedy-algo-5/
The Scheme/Protocol used is: https
The User info is: null
The host is: www.geeksforgeeks.org
The host is null.
The port number is -1
The path is /prims-minimum-spanning-tree-mst-greedy-algo-5/
The anchor of the URL is null
The query string is null
The Hashcode value is 126306724
<
```

eclipse-workspace - Network Programming/src/Lab4/IPUrlProg.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help

Problems @ Javadoc Declaration Console X

```
<terminated> IPUrlProg [Java Application] C:\Users\LENOVO\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.2.v20210721-1
.... The anchor of the URL is null
The query string is null
The Hashcode value is -895585893
The URL CONTENT: sun.net.www.protocol.http.HttpURLConnection$HttpInputStream@3a52dba3
File Name: /admission/overview/

Passing the url host into getByName method to get IP address to obtain further IP characteristics

Host Name of IP :vitap.ac.in
Canonical Host Name of IP :static.52.36.9.5.clients.your-server.de
Loopback address of IP :localhost/127.0.0.1
Is the ip address reachable :true
vitap.ac.in/5.9.36.52 is a global address.
vitap.ac.in/5.9.36.52 is a unicast address.
Enter IP
https://www.geeksforgeeks.org/prims-minimum-spanning-tree-mst-greedy-algo-5/
Entered URL is :https://www.geeksforgeeks.org/prims-minimum-spanning-tree-mst-greedy-algo-5/
The Scheme/Protocol used is: https
The User info is: null
The host is: www.geeksforgeeks.org
The host is null.
The port number is -1
The path is /prims-minimum-spanning-tree-mst-greedy-algo-5/
The anchor of the URL is null
The query string is null
The Hashcode value is 126306724
The URL CONTENT: sun.net.www.protocol.http.HttpURLConnection$HttpInputStream@4c178a76
File Name: /prims-minimum-spanning-tree-mst-greedy-algo-5/

Passing the url host into getByName method to get IP address to obtain further IP characteristics

Host Name of IP :www.geeksforgeeks.org
Canonical Host Name of IP :202.83.30.234
Loopback address of IP :localhost/127.0.0.1
Is the ip address reachable :true
www.geeksforgeeks.org/202.83.30.234 is a global address.
www.geeksforgeeks.org/202.83.30.234 is a unicast address.
```