Practical No. 4(a)
Q. Create a Servlet application to upload a file.

index.html

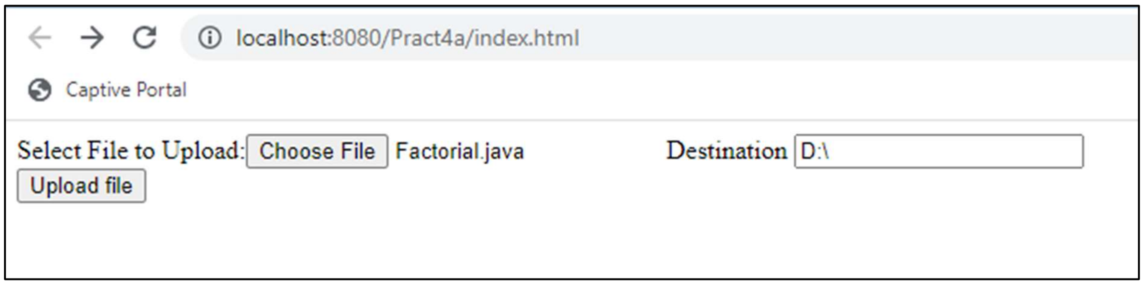
```
<html>
   <body>
      <form action="FileUploadServlet" method="post" enctype="multipart/form-data">
         Select File to Upload:<input type="file" name="file" id="file">
         Destination <input type="text" value="/" name="destination">
         <br>
         <input type="submit" value="Upload file" name="upload" id="upload">
      </form>
   </body>
</html>
```

FileUploadServlet.java

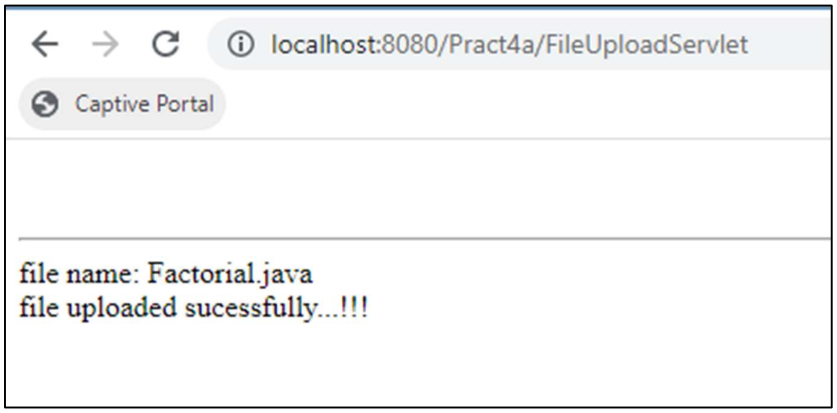
```
import java.io.*;
import javax.servlet.*;
import javax.servlet.annotation.MultipartConfig;
import javax.servlet.http.*;
@MultipartConfig
public class FileUploadServlet extends HttpServlet {
   @Override
   public void doPost(HttpServletRequest req,HttpServletResponse res) throws
ServletException,
   IOException
   {
    res.setContentType("text/html");
    PrintWriter out = res.getWriter();
    String path=req.getParameter("destination");
    Part filePart=req.getPart("file");
    String filename=filePart.getSubmittedFileName().toString();
    out.print("<br><br><hr> file name: "+filename);
    OutputStream os=null;
    InputStream is=null;
    try
    {
      os=new FileOutputStream(new File(path+File.separator+filename));
      is=filePart.getInputStream();

      int read=0;
```

```
        byte[] b=new byte[1024];
        while ((read = is.read(b)) != -1)
        {
            os.write(b, 0, read);
        }
        out.println("<br>file uploaded sucessfully...!!!");
    }
    catch(FileNotFoundException e)
    {
        out.print(e);
    }
}
}
```

Output:



Select File to Upload: Factorial.java Destination D:\



```
file name: Factorial.java
file uploaded sucessfully...!!!
```

Practical No. 4(b)
Q. Create a Servlet application to download a file.

index.html

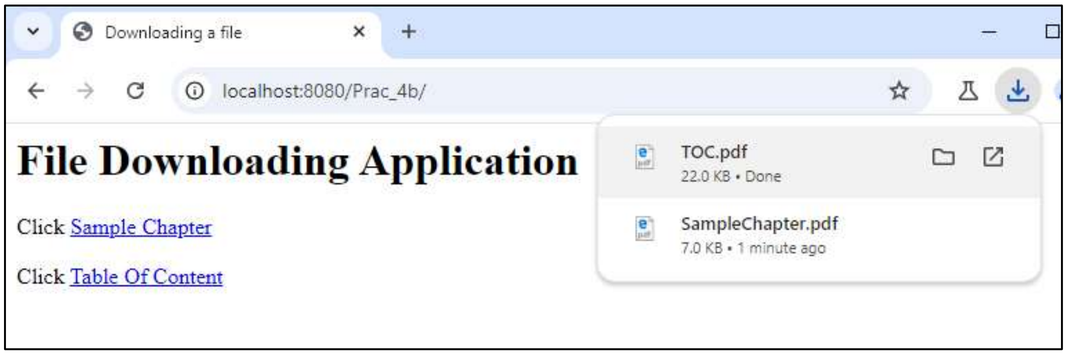
```
<html>
   <head>
      <title> Downloading a file </title>
   </head>
   <body>
      <h1>File Downloading Application</h1>
      Click <a href="FileDownloadServlet?filename=SampleChapter.pdf"> Sample
Chapter</a>
      <br/><br/>
      Click <a href="FileDownloadServlet?filename=TOC.pdf">Table Of Content</a>
   </body>
</html>
```

FileDownloadServlet.java

```
import java.io.IOException;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import java.io.*;
import javax.servlet.*;

public class FileDownloadServlet extends HttpServlet
{
   @Override
   public void doGet(HttpServletRequest request, HttpServletResponse response)throws
ServletException, IOException
   {
      response.setContentType("APPLICATION/OCTET-STREAM");
      String filename = request.getParameter("filename");
      ServletContext context = getServletContext();
      InputStream is = context.getResourceAsStream("/" + filename);
      ServletOutputStream os = response.getOutputStream();
      response.setHeader("Content-Disposition","attachment; filename=\"" + filename + "\"");
      int i;
      byte b[]=new byte[1024];
      while ((i=is.read(b)) != -1)
      {
         os.write(b);
      }
```

```
        is.close();
        os.close();
    }
}
```

Output:



Downloading a file

localhost:8080/Prac_4b/

File Downloading Application

Click Sample Chapter

Click Table Of Content

TOC.pdf
22.0 KB • Done

SampleChapter.pdf
7.0 KB • 1 minute ago

Practical No. 4(c)
Q. Create a Servlet application to download a file.

ReadingListener.java (java file)

```
package servlet;

import java.io.IOException;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.AsyncContext;
import javax.servlet.ReadListener;
import javax.servlet.ServletInputStream;

public class ReadingListener implements ReadListener
{
   private ServletInputStream inputStream = null;
   private AsyncContext context = null;

   public ReadingListener (ServletInputStream in, AsyncContext ac)
   {
      this.inputStream = in;
      this.context = ac;
   }
   @Override
   public void onDataAvailable()
   {
      try
      {
         StringBuilder stringBuilder = new StringBuilder();
         int len = -1;
         byte bytes[] = new byte[1024];
         while (inputStream.isReady() && (len = inputStream.read(bytes)) != -1)
         {
            String data = new String(bytes, 0, len);
         }
      }
      catch (IOException ex)
      {
         Logger.getLogger(ReadingListener.class.getName()).log(Level.SEVERE, null, ex);
      }
   }
   @Override
   public void onAllDataRead()
   {
      System.out.println("Invoked onAllDataRead()");
      context.complete();
```

```
    }
    @Override
    public void onError(Throwable t)
    {
        context.complete();
    }
}
```

ReadingNonBlockingServlet.java

```
package servlet;
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.AsyncContext;
import javax.servlet.ServletException;
import javax.servlet.ServletInputStream;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet(name = "ReadingNonBlockingServlet", urlPatterns =
{"/ReadingNonBlockingServlet"}, asyncSupported=true)
public class ReadingNonBlockingServlet extends HttpServlet
{
    @Override
    protected void service(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException
    {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter out = response.getWriter()) {
        out.println("<!DOCTYPE html>");
        out.println("<html>");
        out.println("<head>");
        out.println("<title>File Reading Servlet Using Non Blocking I/O</title>");
        out.println("</head>");
        out.println("<body>");
        AsyncContext context = request.startAsync();
        ServletInputStream inputStream = request.getInputStream();
        inputStream.setReadListener(new ReadingListener(inputStream, context));
        out.println("</body>");
        out.println("</html>");
    }
}
}
```

NonBlockingServlet.java

```
package servlet;
```

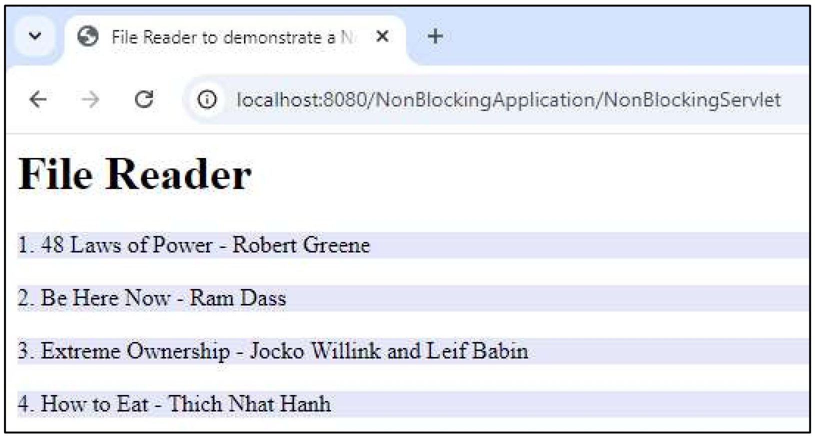
```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.OutputStreamWriter;
import java.io.PrintWriter;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.servlet.ServletContext;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
@WebServlet(name = "NonBlockingServlet", urlPatterns = {"/NonBlockingServlet"})
public class NonBlockingServlet extends HttpServlet
{
  @Override
  protected void service(HttpServletRequest request, HttpServletResponse response) throws
ServletException, IOException {
  response.setContentType("text/html;charset=UTF-8");
  String filename = "/WEB-INF/booklist.txt";
  ServletContext context = getServletContext();

  InputStream inputStream = context.getResourceAsStream(filename);
  try (PrintWriter out = response.getWriter())
  {
    String path = "http://" + request.getServerName() + ":" + request.getServerPort() +
    request.getContextPath() + "/ReadingNonBlockingServlet";
    out.println("<html>");
    out.println("<head>");
    out.println("<title>File Reader to demonstrate a Non Blocking I/O Servlet</title>");
    out.println("</head>");
    out.println("<body>");
    out.println("<h1>File Reader</h1>");
    out.flush();
    URL url = new URL(path);
    HttpURLConnection conn = (HttpURLConnection) url.openConnection();
    conn.setChunkedStreamingMode(2);
    conn.setDoOutput(true);
    conn.connect();
    if (inputStream != null)
    {
      InputStreamReader inputStreamReader = new InputStreamReader(inputStream);
      BufferedReader bufferReader = new BufferedReader(inputStreamReader);
      String text = "";
      System.out.println("Reading started...");
```

```
         try (BufferedWriter bufferWriter = new BufferedWriter(new
OutputStreamWriter(conn.getOutputStream())))
         {
            out.println("<div style='width=100%;height:450px;overflow:scroll;'>");
            while ((text = bufferReader.readLine()) != null)
            {
               out.println("<div style='background-color:lavender;width=100%;'>");
               out.println(text);
               out.println("</div><br/>");
               out.flush();
               bufferWriter.write(text);
               Thread.sleep(1000);
               out.flush();
            }
            out.println("</div>");
            System.out.println("Reading completed...");

            bufferWriter.flush();
            bufferWriter.close();
         }
      }
      out.println("</body>");
      out.println("</html>");
   }
   catch (InterruptedException | IOException ex)
   {
      Logger.getLogger(NonBlockingServlet.class.getName()).log(Level.SEVERE, null, ex);
   }
}
}
```

index.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
   <head>
      <meta http-equiv="Refresh" content="0; URL=NonBlockingServlet">
      <title>Non-blocking I/O</title>
   </head>
   <body>
      <h1>Hello World!</h1>
   </body>
</html>
```

Output:


File Reader to demonstrate a N
localhost:8080/NonBlockingApplication/NonBlockingServlet

File Reader

```
1. 48 Laws of Power - Robert Greene

2. Be Here Now - Ram Dass

3. Extreme Ownership - Jocko Willink and Leif Babin

4. How to Eat - Thich Nhat Hanh
```