**import** java.io.FileInputStream;

**import** java.io.InputStream;

**import** java.io.OutputStream;

**import** java.net.Socket;

**import** java.util.LinkedList;

**import** java.util.Queue;

**import** java.util.Random;

**import** android.util.Log;

**public** **class** SendingThread **implements** Runnable{

Socket socket;

Queue<**byte**[]> u= **new** LinkedList<**byte**[]>();

String TAG = "SendingThread";

**public** SendingThread(Socket socket){

**this**.socket = socket;

}

//Constructor takes a socket as an input. This socket is the socket returned by the accept().

@Override

**public** **void** run() {

// **TODO** Auto-generated method stub

**try**{

String fileName="/sdcard/Movies/ssomemovie.mp4";//The directory of the file to be sent.

InputStream in = **new** FileInputStream(fileName);//Open a file input stream to read from the file.

OutputStream out = socket.getOutputStream();//Get the output stream from the socket to send the bytes.

**byte**[] fileBytes = **new** **byte**[500];//This byte array will hold the bytes read from the file.

**int** len = 0;

**byte**[] buffer;

**int** count=0;

**int** n=1000;

**while** (n>0)

{

count = *getPoissonRandom*(10);

Log.*d*("SERVER", "count is value " + count);

buffer=**new** **byte**[len];

**while**(in.read(fileBytes)!=-1 && count>0){//Read the file until the end.

System.*arraycopy*(fileBytes, 0, buffer, 0, len);

u.offer(buffer);

count--;

}

**for**(**int** i=0; i<12; i++)

{

**if** (!u.isEmpty()) {

out.write(u.poll());//Write the file to storage.

out.flush();

}

}

n--;

Thread.*sleep*(500);

}

in.close();//Close the file input stream.

out.close();//Close the output stream of the socket.

socket.close();//Close the socket.

}**catch**(Exception e){

Log.*d*(TAG,"Wrong!!!!!!!");//Handle the exception

}

}

**private** **static** **int** getPoissonRandom(**double** mean) {

Random r = **new** Random();

**double** L = Math.*exp*(-mean);

**int** k = 0;

**double** p = 1.0;

**do** {

p = p \* r.nextDouble();

k++;

} **while** (p > L);

**return** k - 1;

}

}