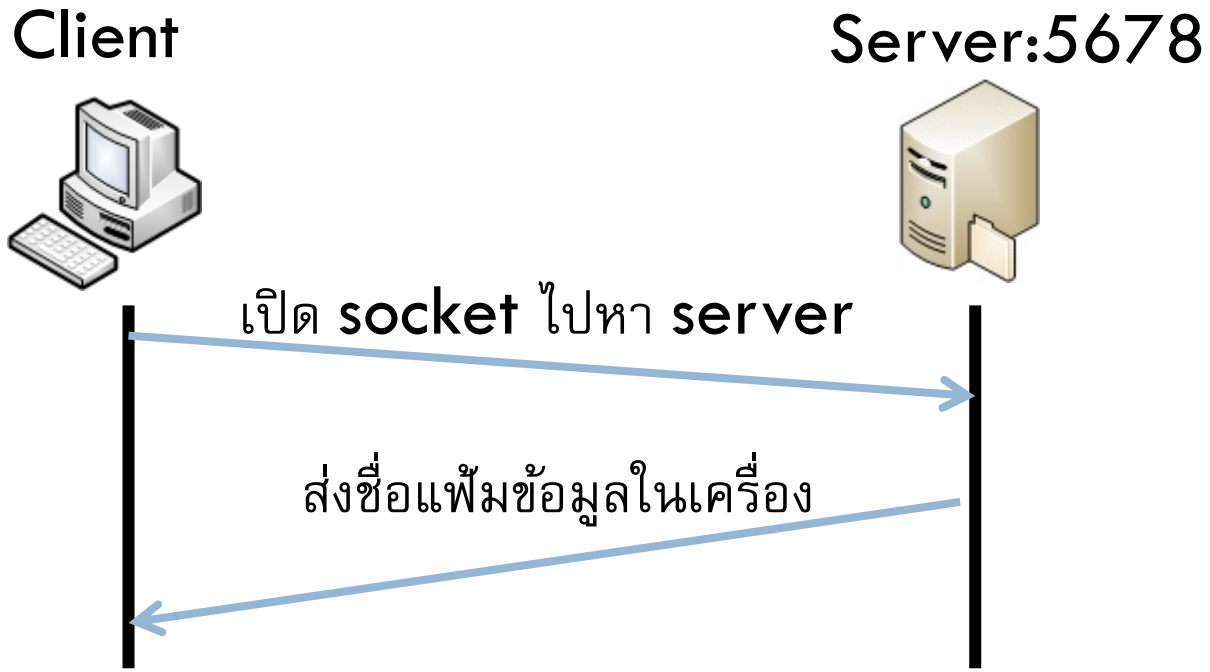
CLIENT/SERVER APPLICATION (FILE SERVER)

030523313 - Network programming
Asst. Prof. Dr. Choopan Rattanapoka

Introduction

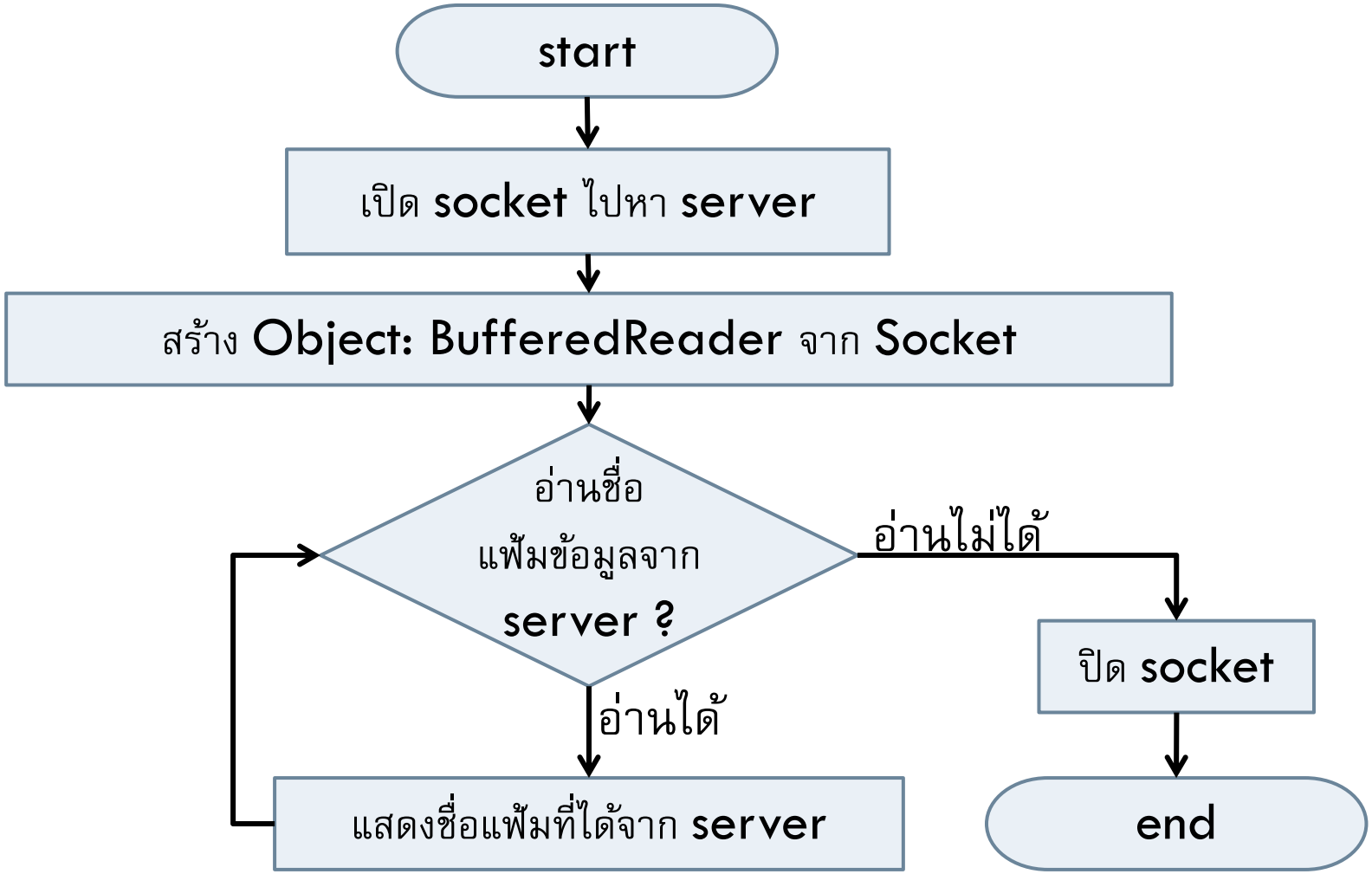
- Server/Client ของระบบแฟ้มข้อมูล โดยจะมีตัวอย่างการทำงานแบบง่ายๆ ของการทำงาน 3 อย่างคือ
  - **List** ดูรายชื่อแฟ้มข้อมูลที่เครื่อง **server**
  - **Upload** แฟ้มข้อมูล
  - **Download** แฟ้มข้อมูล

การดูรายชื่อของแฟ้มข้อมูลบน Server (LIST)

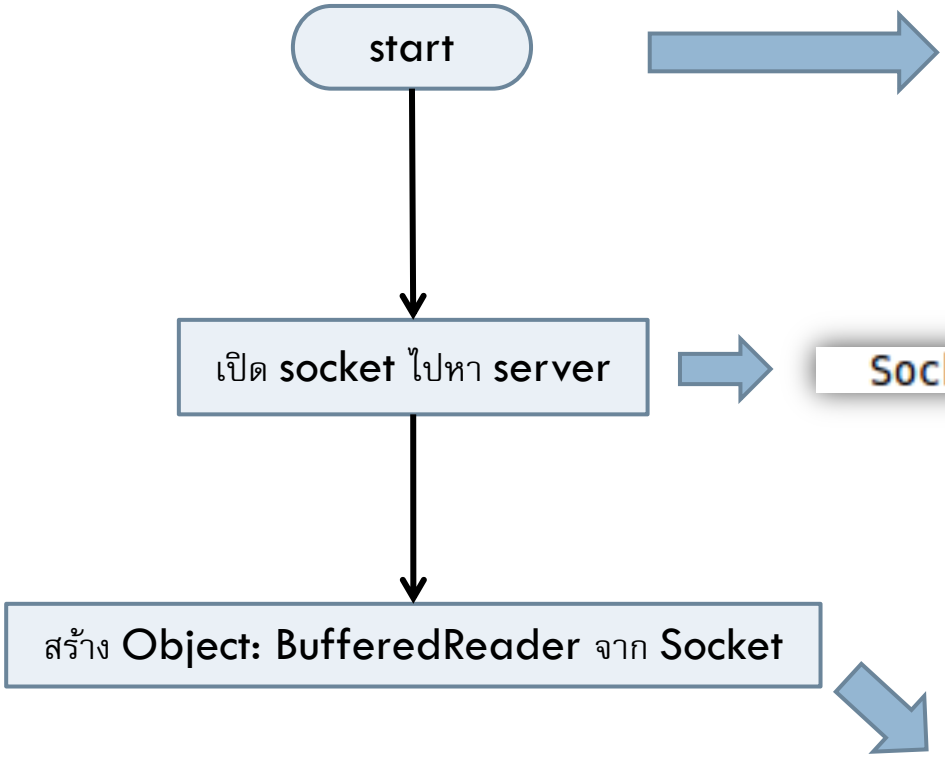


- **เตือนความจำ**
  - ใน Java สามารถดูรายชื่อใน Directory ได้ด้วยการใช้เมธอด **list()** ของ Object **File**

Flow Chart : Client (LIST)



Flow Chart to Code (1)

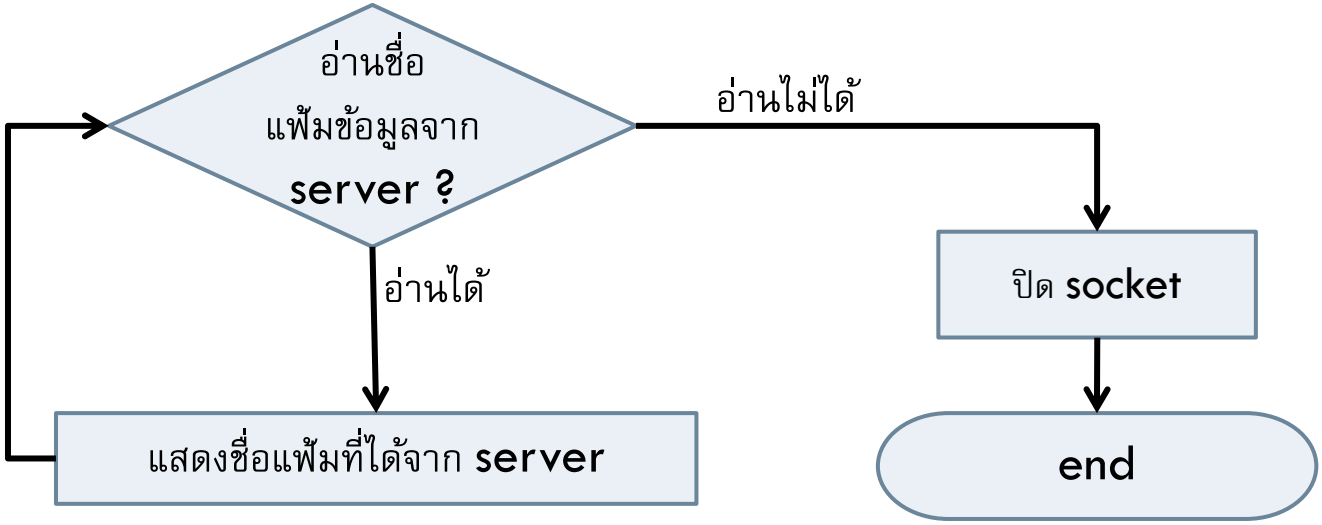



```
import java.io.*;
import java.net.*;

public class FileList {
     public static void main(String[] args)
```

```
Socket s = new Socket("127.0.0.1", 5678);
```

```
InputStream in   = s.getInputStream();
OutputStream out = s.getOutputStream();

BufferedReader br = new BufferedReader(new InputStreamReader(in));
```

Flow Chart to Code (2)

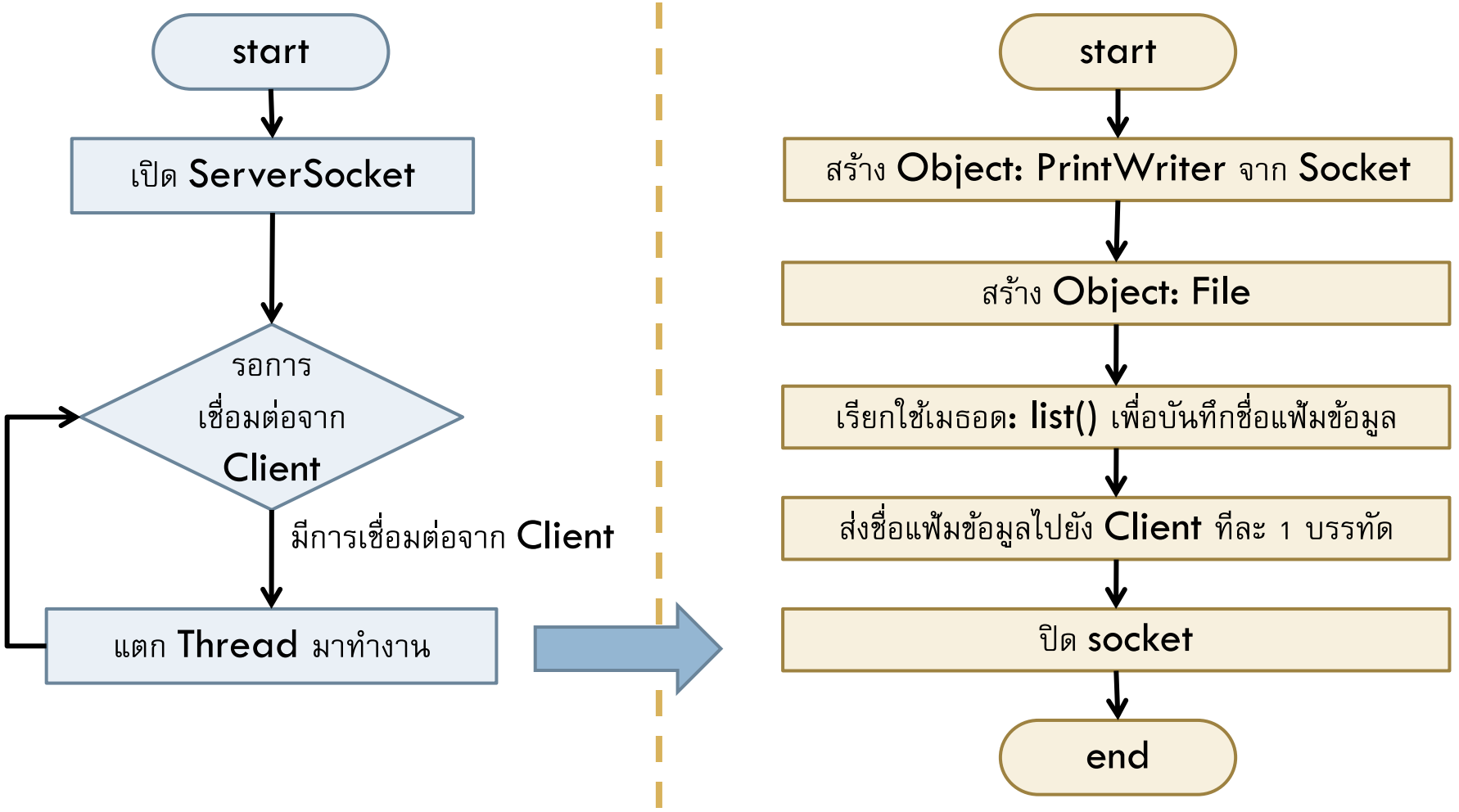



```
String filename;
while((filename = br.readLine()) != null) {
        System.out.println(filename);
}
in.close();
out.close();
s.close();
```

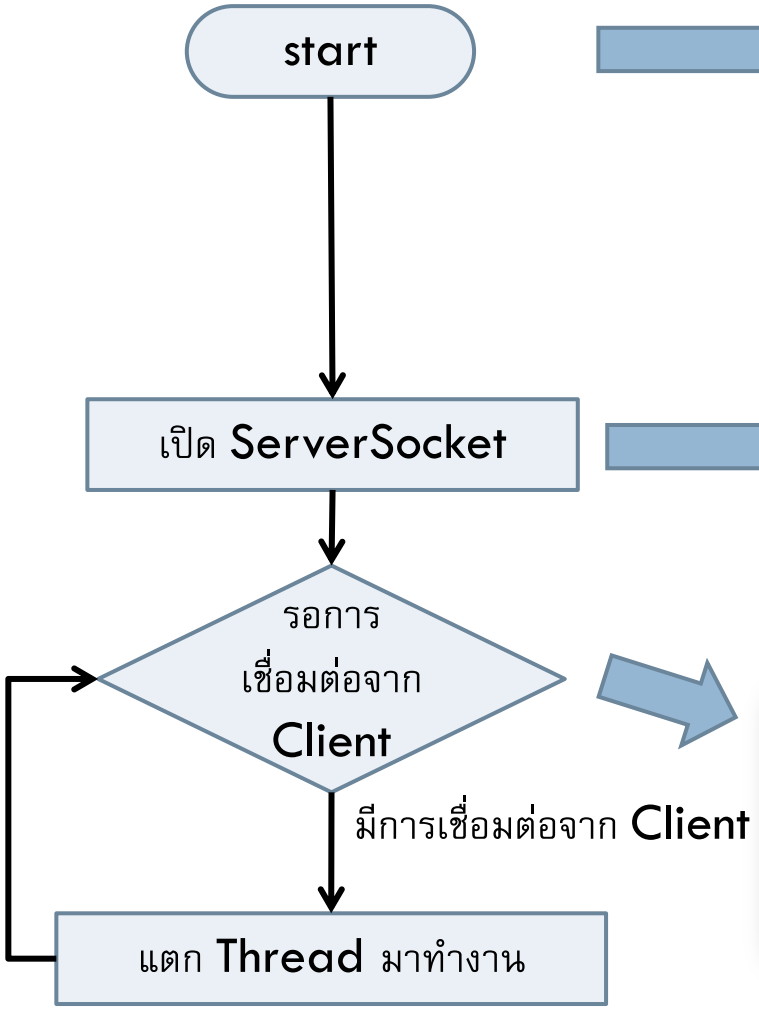
Source Code : FileList.java

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

public class FileList {
     public static void main(String[] args) {
              try {
                       Socket s = new Socket("127.0.0.1", 5678);
                       InputStream in   = s.getInputStream();
                       OutputStream out = s.getOutputStream();

                       BufferedReader br = new BufferedReader(new InputStreamReader(in));

                       String filename;
                       while((filename = br.readLine()) != null) {
                                System.out.println(filename);
                       }
                       in.close();
                       out.close();
                       s.close();
              } catch(Exception e) { e.printStackTrace(); }

     }
}
```

Flow Chart : Server (LIST)



Flow Chart to Code (1)

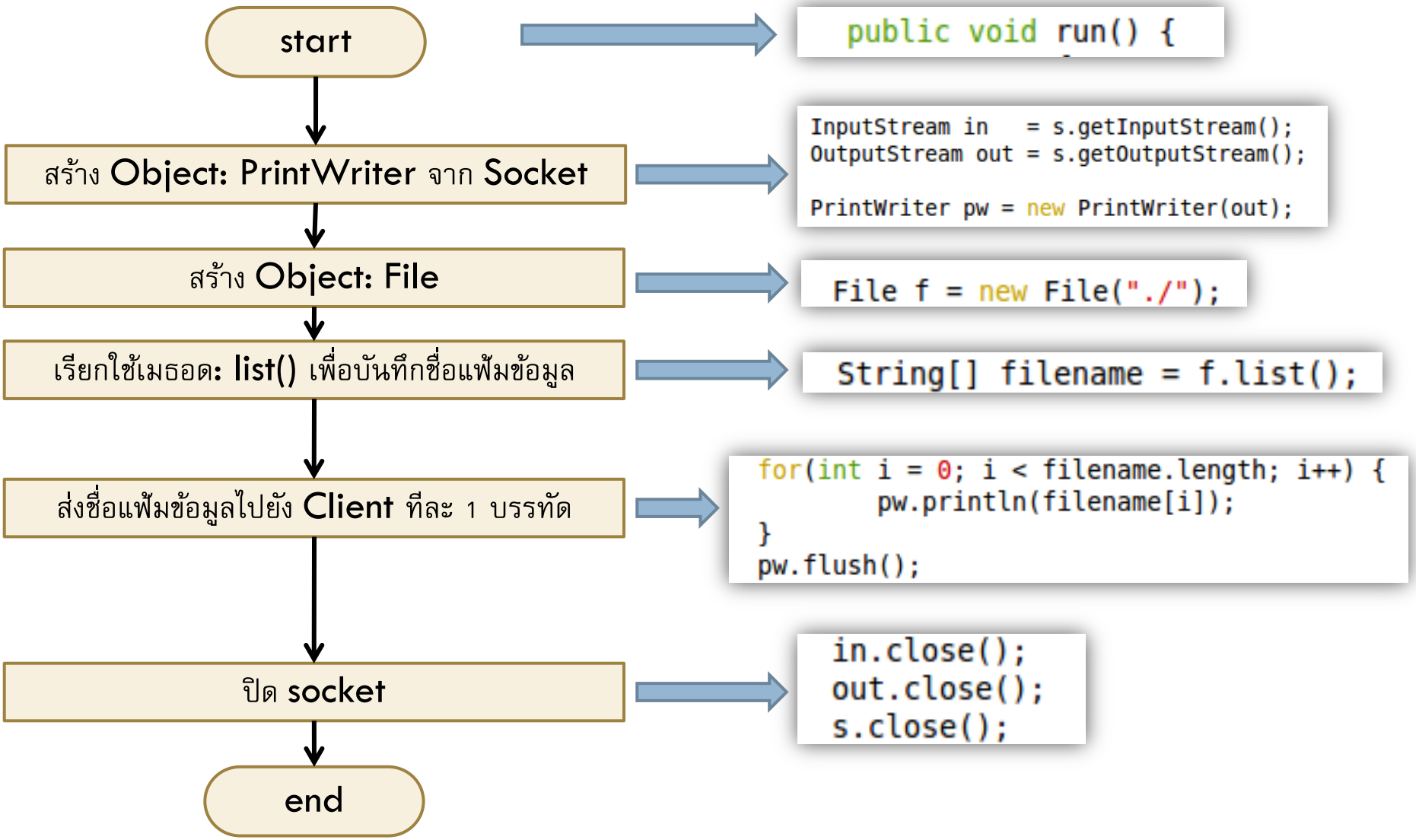



```
import java.io.*;
import java.net.*;

public class FileServerList implements Runnable {
     Socket s = null;

     public FileServerList(Socket s) {
             this.s = s;
     }
```

```
public static void main(String[] args) {
    ServerSocket serv = new ServerSocket(5678);
```

```
while(true) {
        Socket s = serv.accept();
        FileServerList fs = new FileServerList(s);
        Thread t = new Thread(fs);
        t.start();
}
```

Flow Chart to Code (2)



Source Code : FileServerList.java

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

public class FileServerList implements Runnable {
	Socket s = null;

	public FileServerList(Socket s) {
		this.s = s;
	}

	public void run() {
		try {
			InputStream in  = s.getInputStream();
			OutputStream out = s.getOutputStream();

			PrintWriter pw = new PrintWriter(out);

			File f = new File("./");
			String[] filename = f.list();
			for(int i = 0; i < filename.length; i++) {
				pw.println(filename[i]);
			}
			pw.flush();
			in.close();
			out.close();
			s.close();
		} catch (Exception e) { e.printStackTrace(); }
	}
```

```java
	public static void main(String[] args) {
		try {
			ServerSocket serv = new ServerSocket(5678);
			while(true) {
				Socket s = serv.accept();
				FileServerList fs = new FileServerList(s);
				Thread t = new Thread(fs);
				t.start();
			}
		} catch(Exception e) { e.printStackTrace(); }
	}
}
```

Client-Server Communication

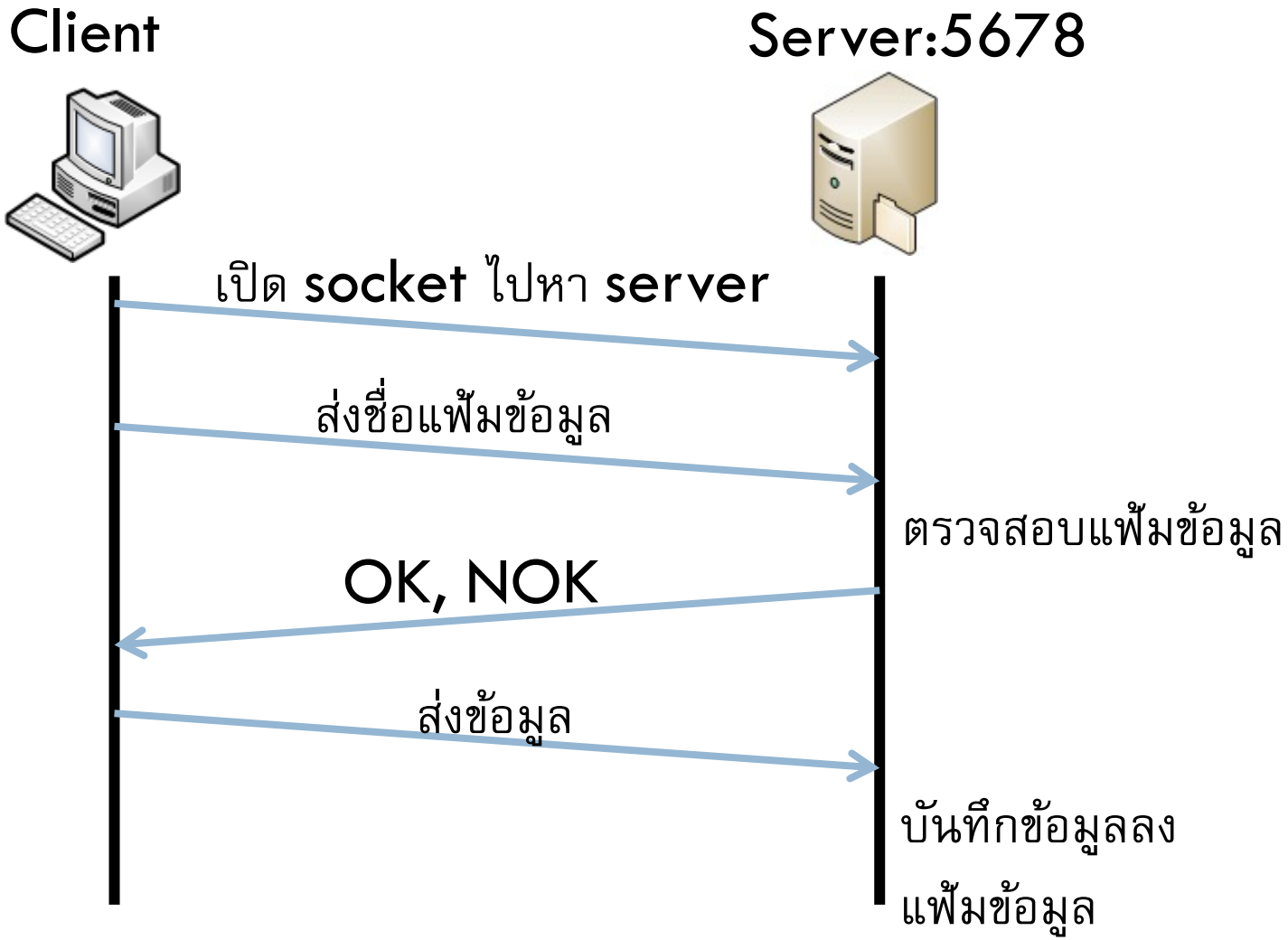
Client

```
try {
        Socket s = new Socket("127.0.0.1", 5678);
        InputStream in   = s.getInputStream();
        OutputStream out = s.getOutputStream();

        BufferedReader br = new BufferedReader(new InputStreamReader(in));

        String filename;
        while((filename = br.readLine()) != null) {
                System.out.println(filename);
        }
        in.close();
        out.close();
        s.close();
} catch(Exception e) { e.printStackTrace(); }
```

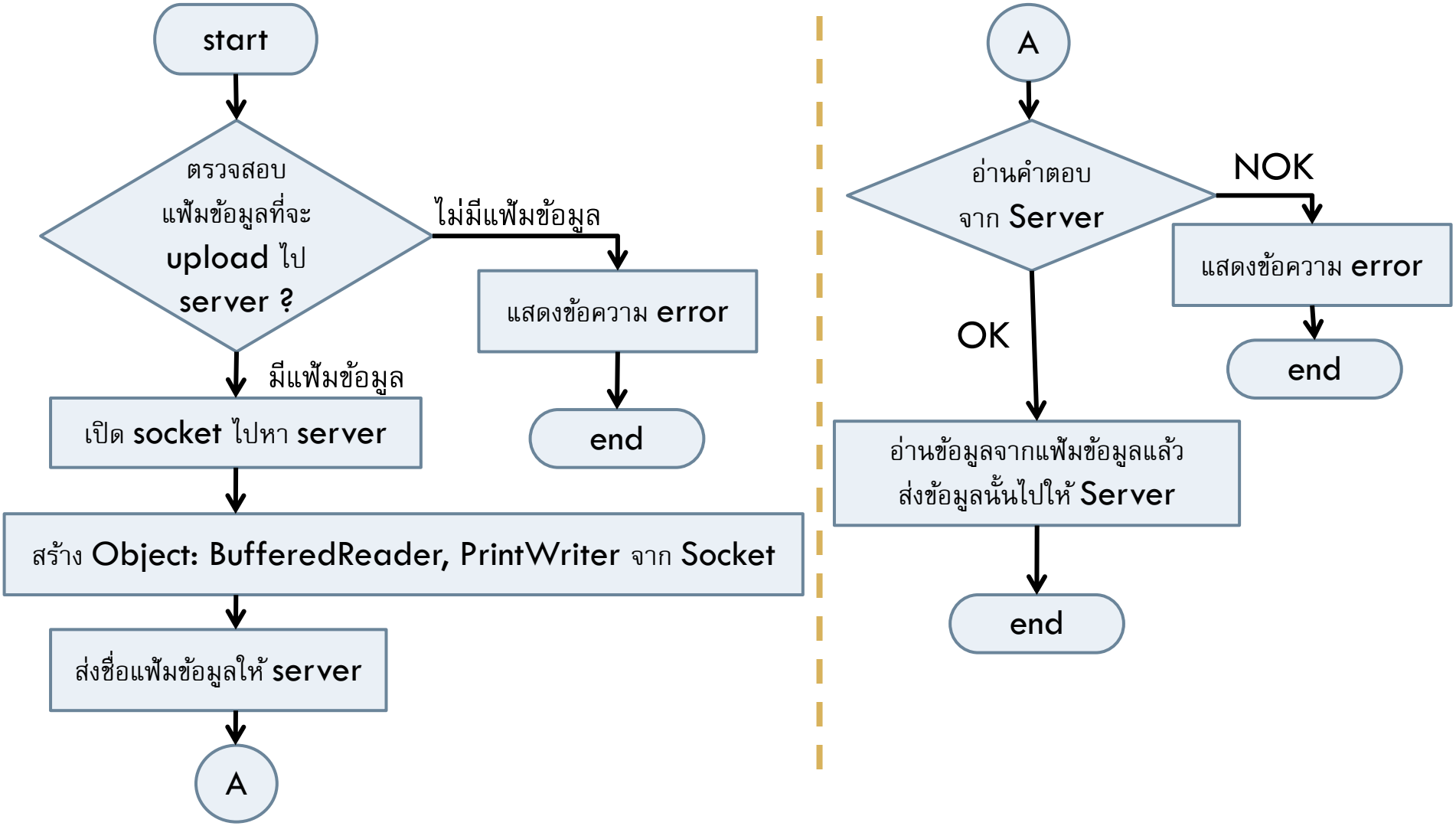
Server

```
public void run() {
        try {
                InputStream in   = s.getInputStream();
                OutputStream out = s.getOutputStream();

                PrintWriter pw = new PrintWriter(out);

                File f = new File("./");
                String[] filename = f.list();
                for(int i = 0; i < filename.length; i++) {
                        pw.println(filename[i]);
                }
                pw.flush();
                in.close();
                out.close();
                s.close();
        } catch (Exception e) { e.printStackTrace(); }
}
```

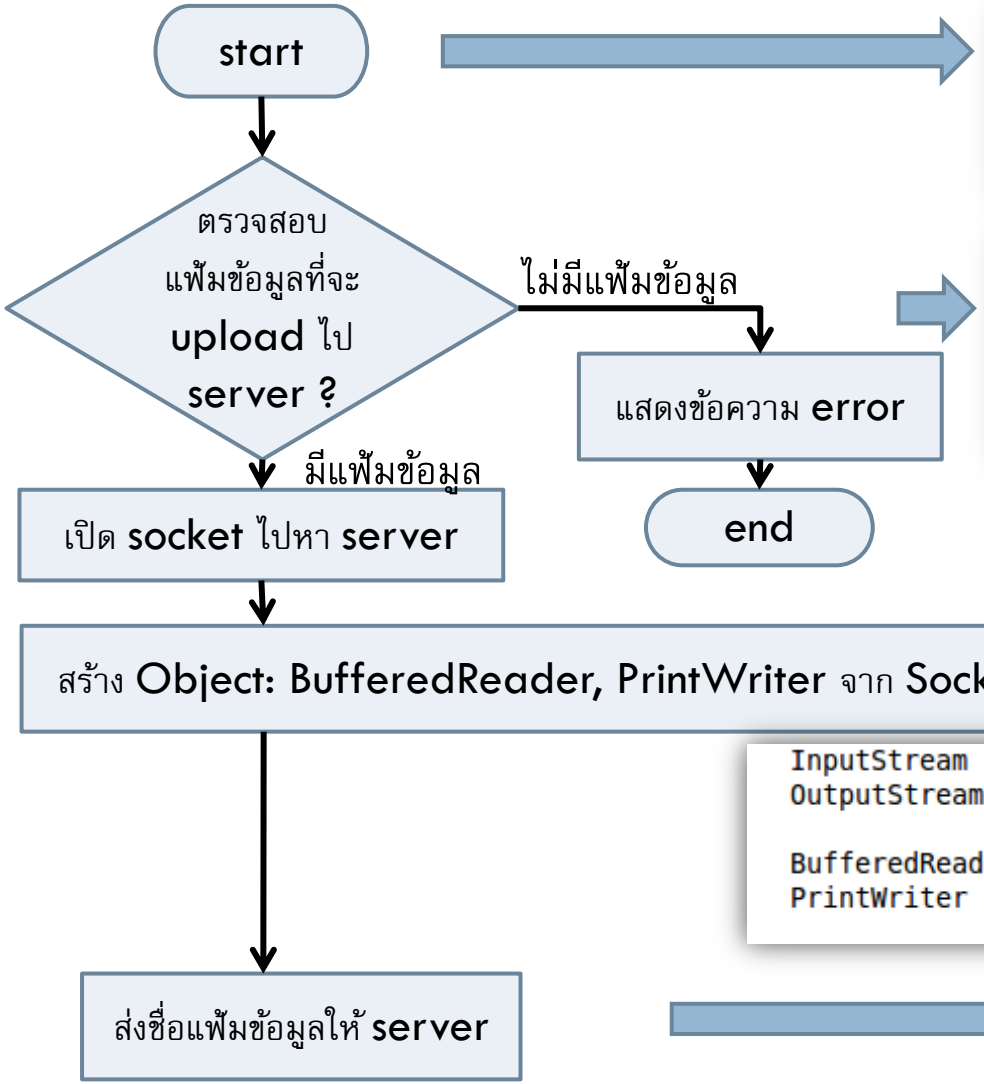
การ Upload แฟ้มข้อมูลเข้าสู่ Server



Flow Chart : Client (Upload)



Flow Chart to Code (1)



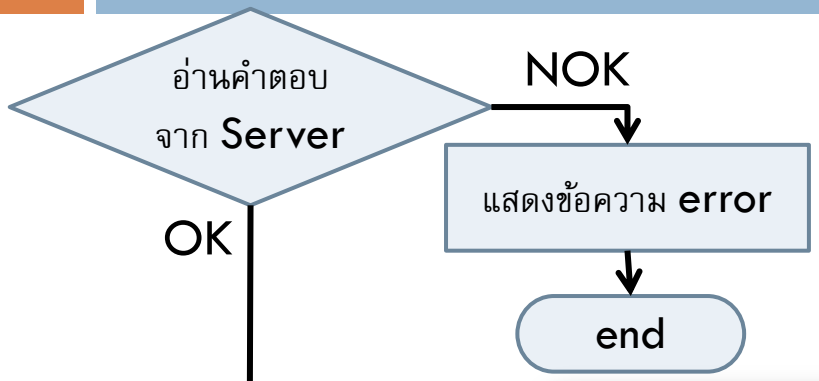

```
import java.io.*;
import java.net.*;

public class FileUpload {
     public static void main(String[] args) {
```

```
   File f = new File(args[0]);
   if(!f.exists()) {
           System.out.println("File upload not found");
           System.exit(1);
   }

   Socket s = new Socket("127.0.0.1", 5678);
```

```
InputStream in   = s.getInputStream();
OutputStream out = s.getOutputStream();

BufferedReader br = new BufferedReader(new InputStreamReader(in));
PrintWriter pw = new PrintWriter(out);
```

```
pw.println(args[0]);
pw.flush();
```

Flow Chart to Code (2)

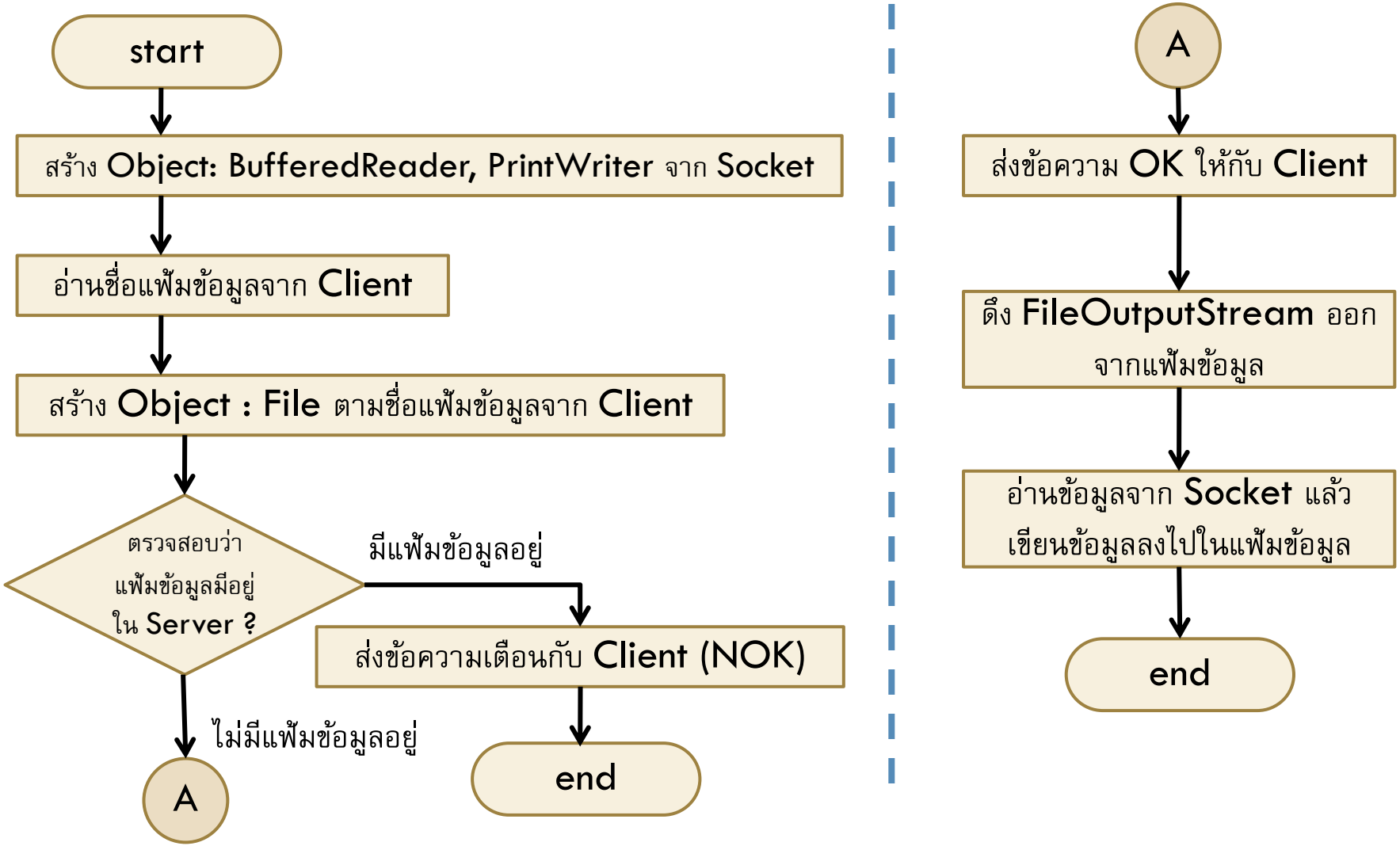



```
String response = br.readLine();
if(response.equals("OK")){
        FileInputStream fin = new FileInputStream(f);
        byte[] buffer = new byte[65536];
        int size;
        while((size = fin.read(buffer)) > 0) {
                out.write(buffer, 0, size);
        }
        out.flush();
        fin.close();
} else {
        System.out.println("File already exists on server");
}
in.close();
out.close();
s.close();
```

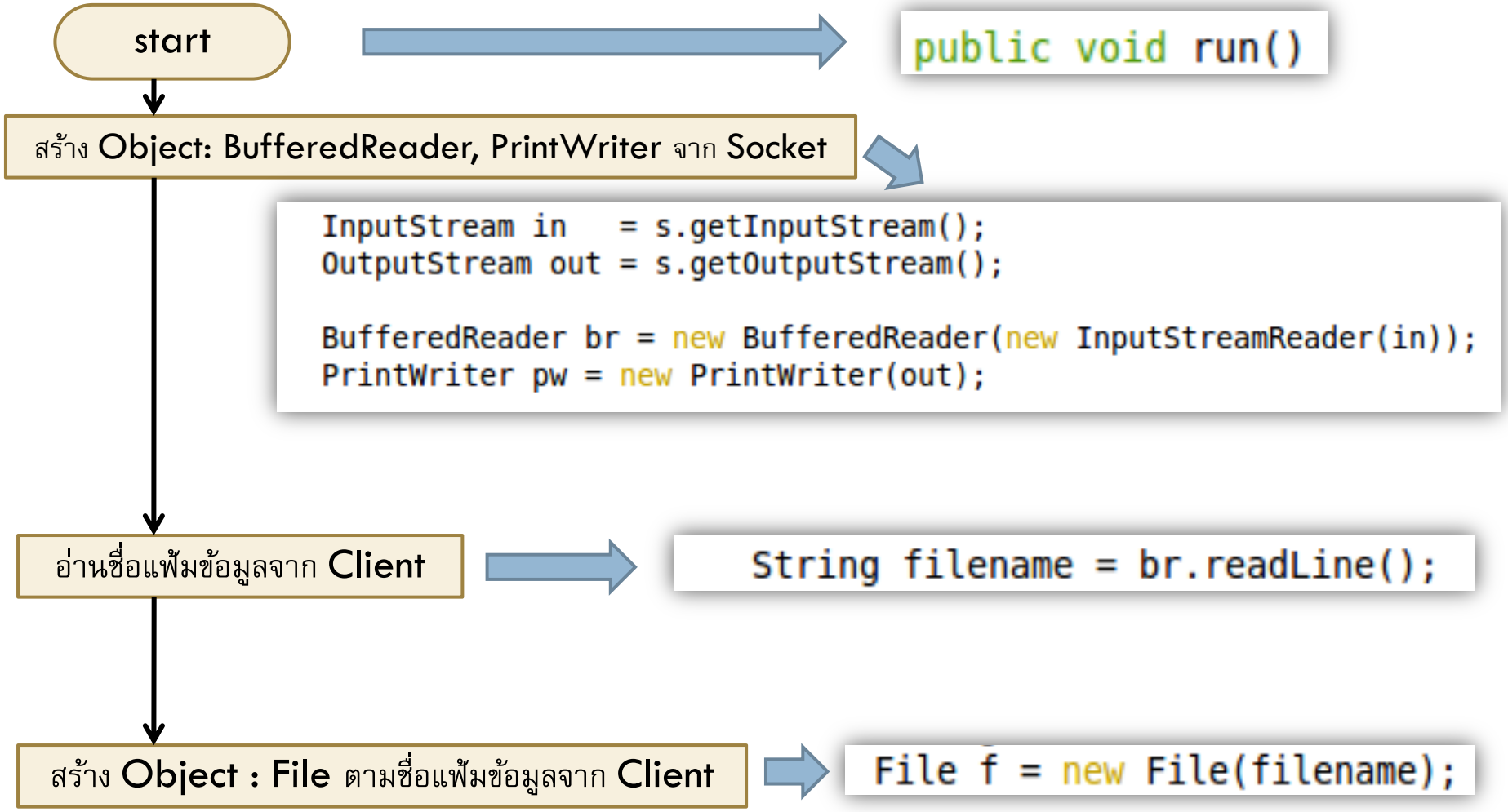
Source Code : FileUpload.java

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

public class FileUpload {
     public static void main(String[] args) {
               try {
                         File f = new File(args[0]);
                         if(!f.exists()) {
                                  System.out.println("File upload not found");
                                  System.exit(1);
                         }

                         Socket s = new Socket("127.0.0.1", 5678);
                         InputStream in   = s.getInputStream();
                         OutputStream out = s.getOutputStream();

                         BufferedReader br = new BufferedReader(new InputStreamReader(in));
                         PrintWriter pw = new PrintWriter(out);

                         pw.println(args[0]);
                         pw.flush();
                         String response = br.readLine();
                         if(response.equals("OK")){
                                  FileInputStream fin = new FileInputStream(f);
                                  byte[] buffer = new byte[65536];
                                  int size;
                                  while((size = fin.read(buffer)) > 0) {
                                           out.write(buffer, 0, size);
                                  }
                                  out.flush();
                                  fin.close();
                         } else {
                                  System.out.println("File already exists on server");
                         }
                         in.close();
                         out.close();
                         s.close();
               } catch(Exception e) { e.printStackTrace(); }

     }
}
```

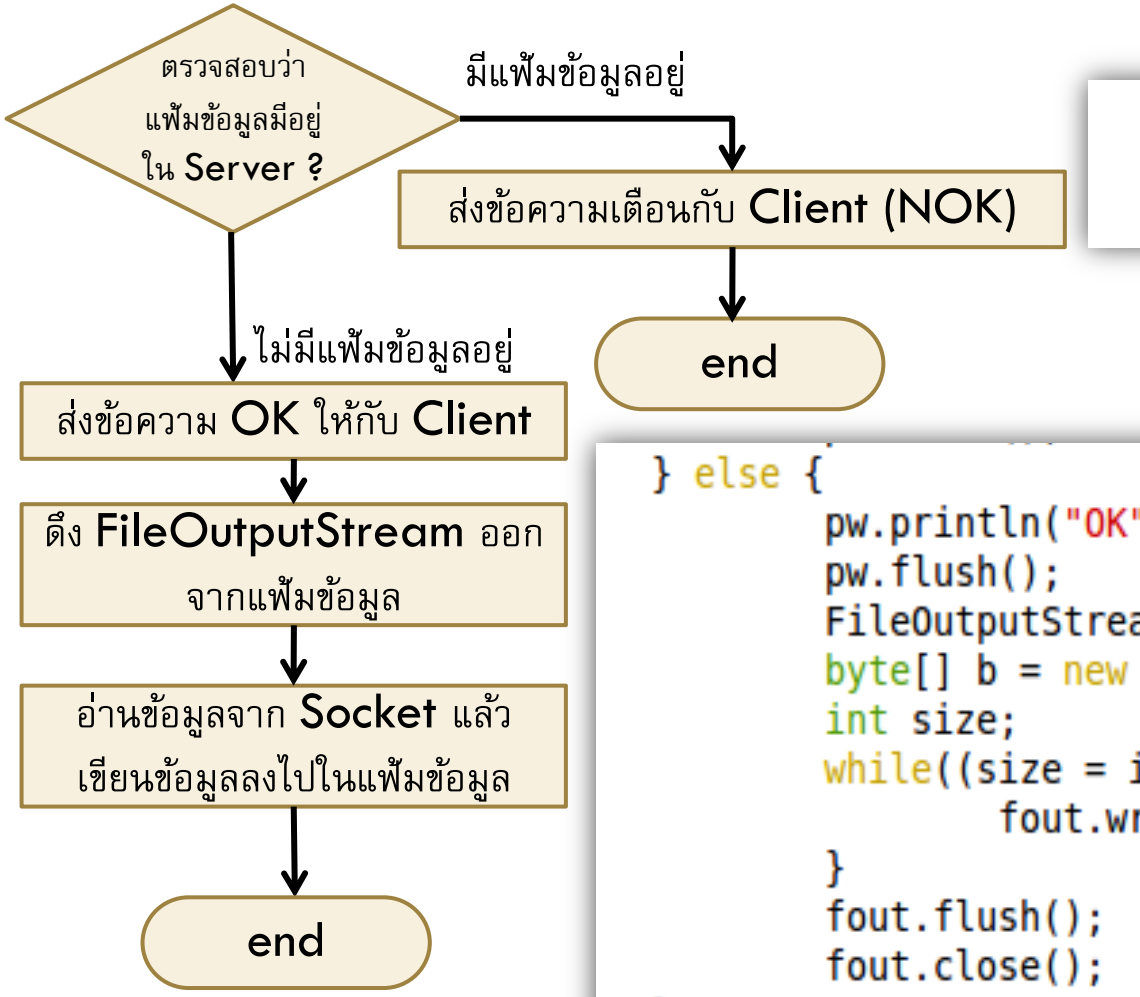
Flow Chart : Server (Upload)



Flow Chart to Code (1)



Flow Chard to Code (2)



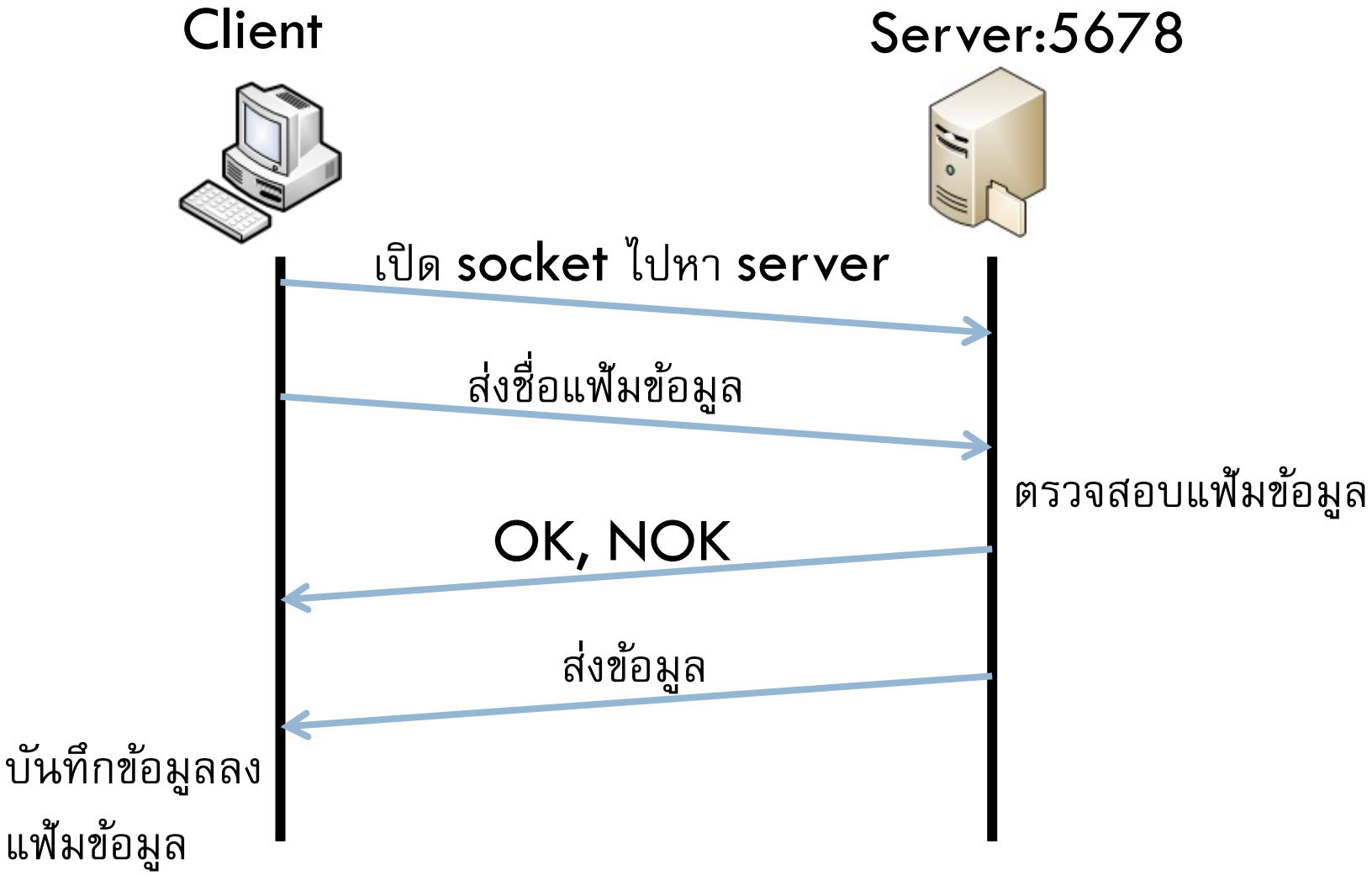

```
if(f.exists()) {
        pw.println("NOK");
        pw.flush();
```

```
} else {
        pw.println("OK");
        pw.flush();
        FileOutputStream fout = new FileOutputStream(f);
        byte[] b = new byte[65536];
        int size;
        while((size = in.read(b)) > 0) {
                fout.write(b, 0, size);
        }
        fout.flush();
        fout.close();
}
```

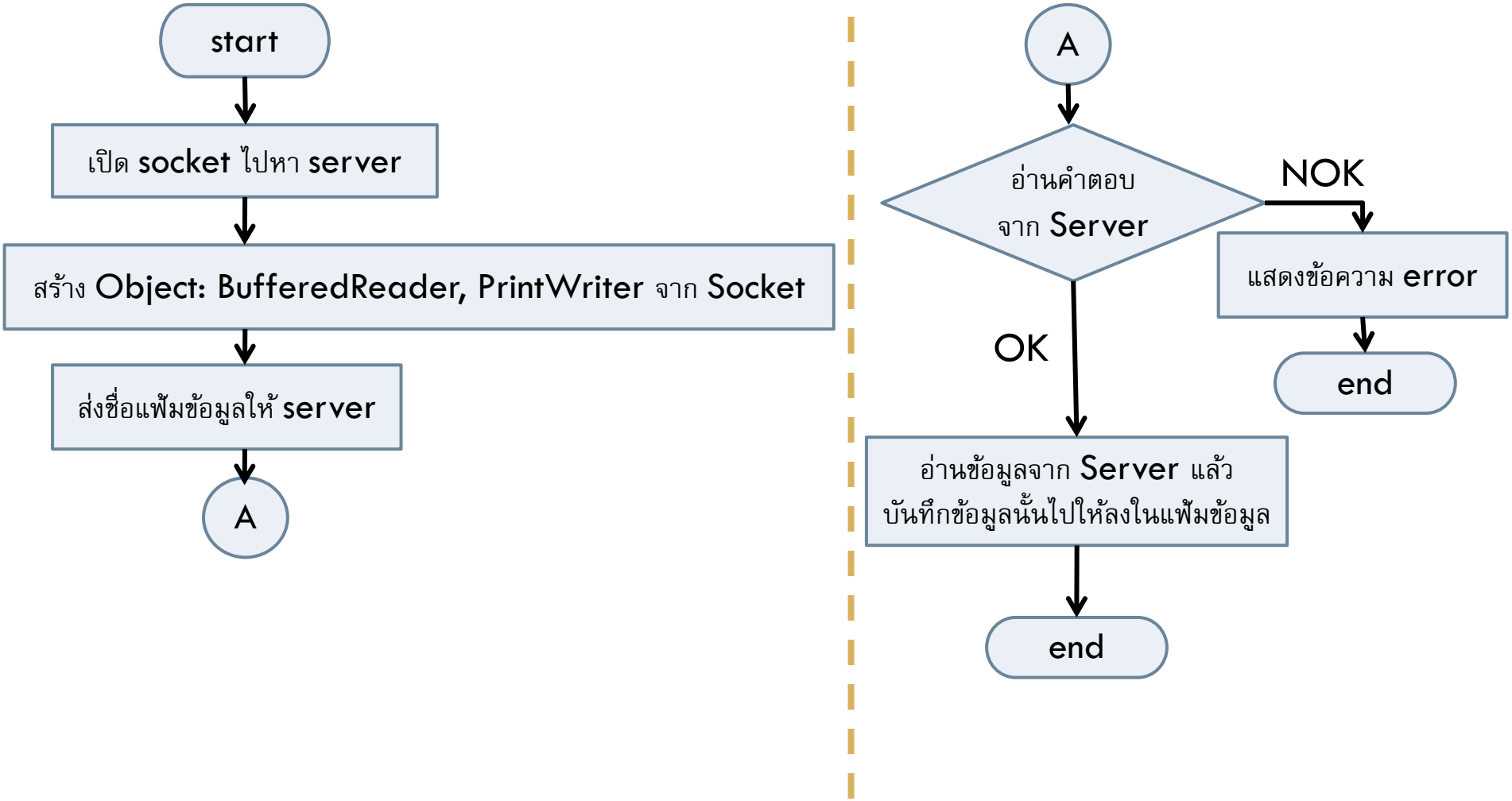
Source Code : FileServerUpload.java

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

public class FileServerUpload implements Runnable {
     Socket s = null;

     public FileServerUpload(Socket s) {
             this.s = s;
     }

     public void run() {
             try {
                     InputStream in   = s.getInputStream();
                     OutputStream out = s.getOutputStream();

                     BufferedReader br = new BufferedReader(new InputStreamReader(in));
                     PrintWriter pw = new PrintWriter(out);

                     String filename = br.readLine();
                     File f = new File(filename);
                     if(f.exists()) {
                             pw.println("NOK");
                             pw.flush();
                     } else {
                             pw.println("OK");
                             pw.flush();
                             FileOutputStream fout = new FileOutputStream(f);
                             byte[] b = new byte[65536];
                             int size;
                             while((size = in.read(b)) > 0) {
                                     fout.write(b, 0, size);
                             }
                             fout.flush();
                             fout.close();
                     }
                     in.close();
                     out.close();
                     s.close();
             } catch (Exception e) { e.printStackTrace(); }
     }
     public static void main(String[] args) {
             try {
                     ServerSocket serv = new ServerSocket(5678);
                     while(true) {
                             Socket s = serv.accept();
                             FileServerUpload fs = new FileServerUpload(s);
                             Thread t = new Thread(fs);
                             t.start();
                     }
             } catch(Exception e) { e.printStackTrace(); }
     }
}
```

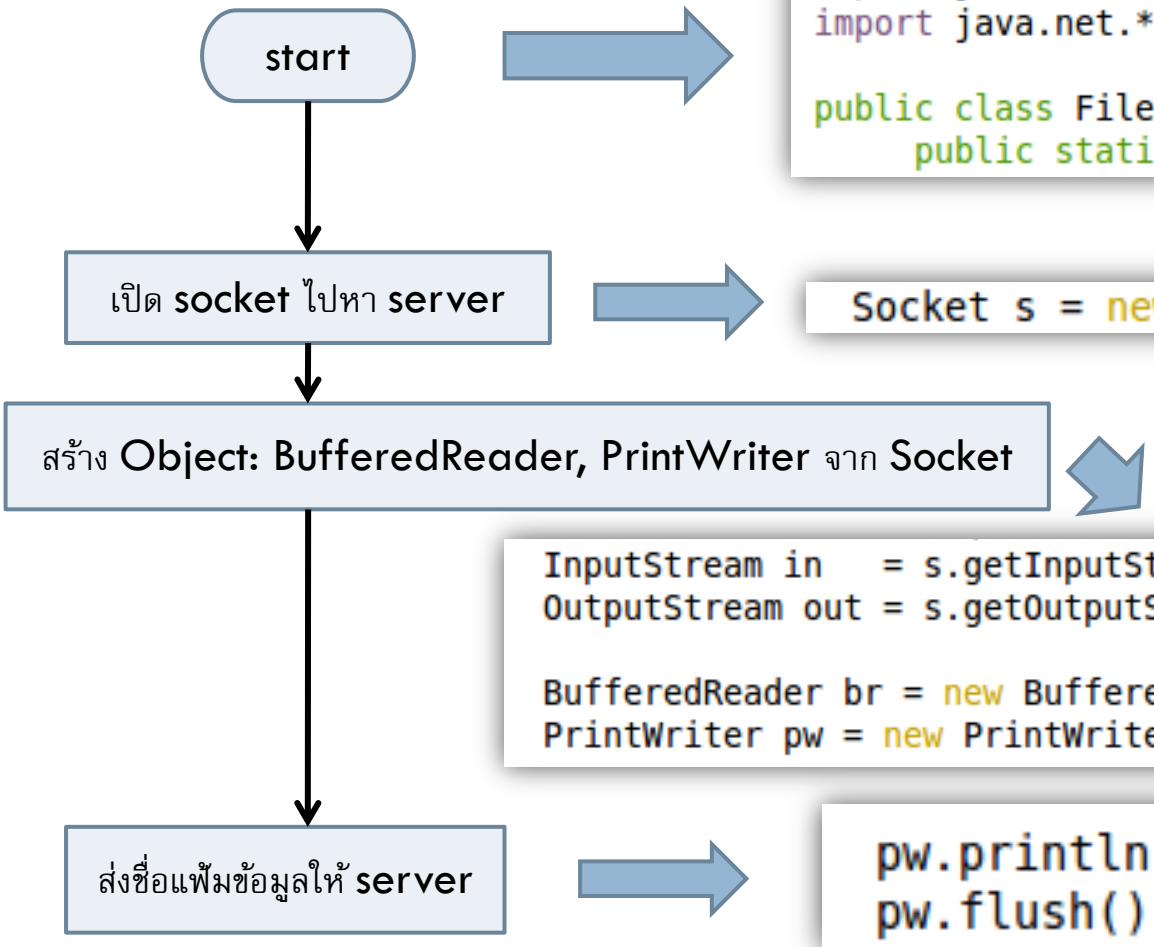
การ Download แฟ้มข้อมูลจาก Server



Flow Chart : Client (Download)



Flow Chart to Code (1)



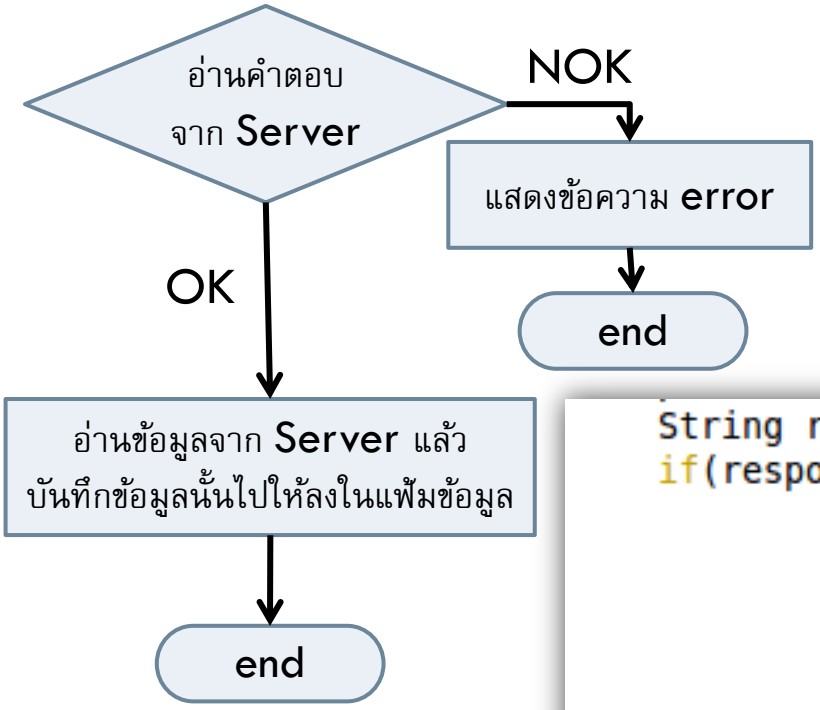

```
import java.io.*;
import java.net.*;

public class FileDownload {
     public static void main(String[] args) {
```

```
Socket s = new Socket("127.0.0.1", 5678);
```

```
InputStream in   = s.getInputStream();
OutputStream out = s.getOutputStream();

BufferedReader br = new BufferedReader(new InputStreamReader(in));
PrintWriter pw = new PrintWriter(out);
```

```
pw.println(args[0]);
pw.flush();
```

Flow Chart to Code (2)

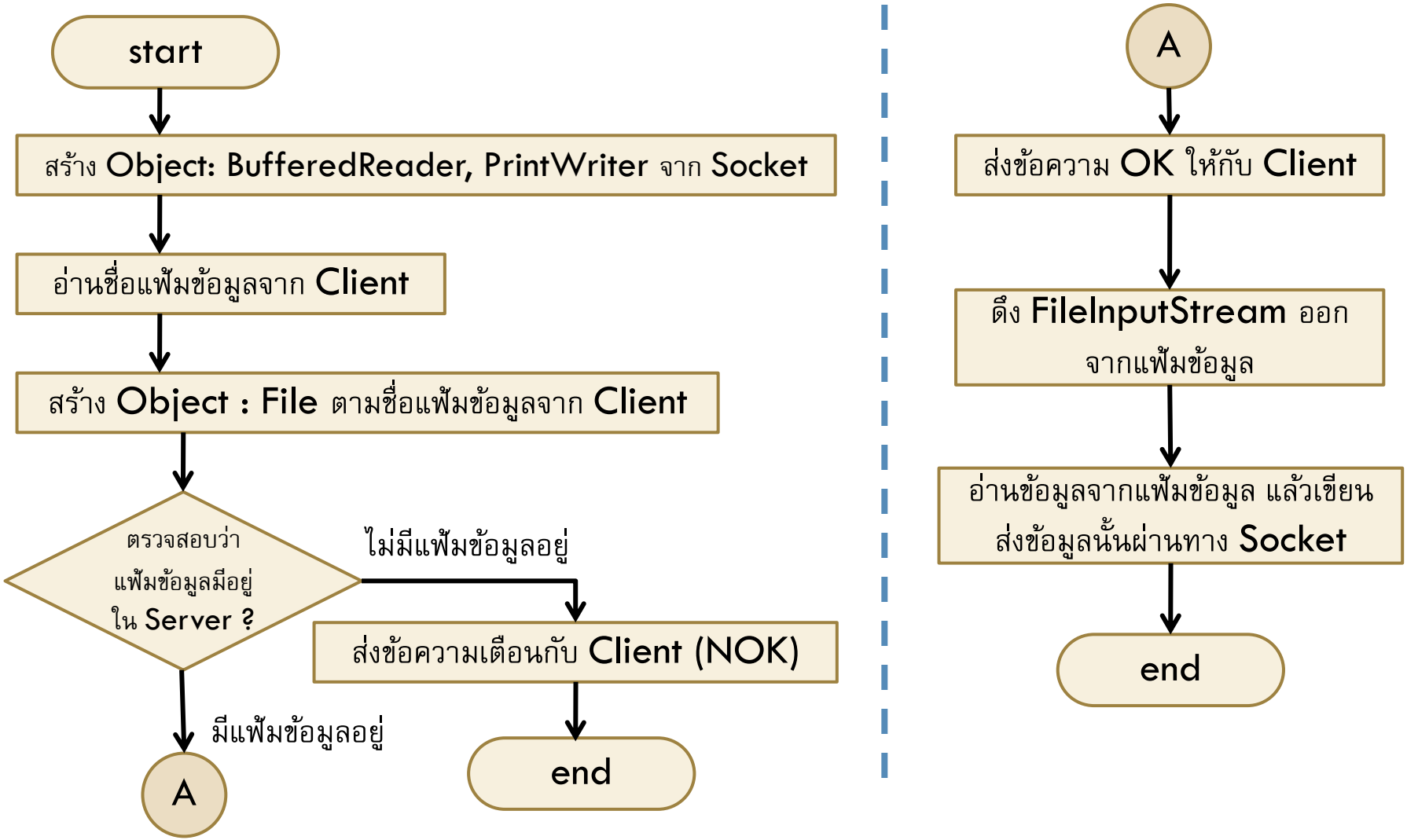



```
String response = br.readLine();
if(response.equals("OK")){
        File f = new File(args[0]);
        FileOutputStream fout = new FileOutputStream(f);
        byte[] b = new byte[65536];
        int size;
        while((size = in.read(b)) > 0) {
                fout.write(b, 0, size);
        }
        in.close();
} else {
        System.out.println("No file found on server");
}
```

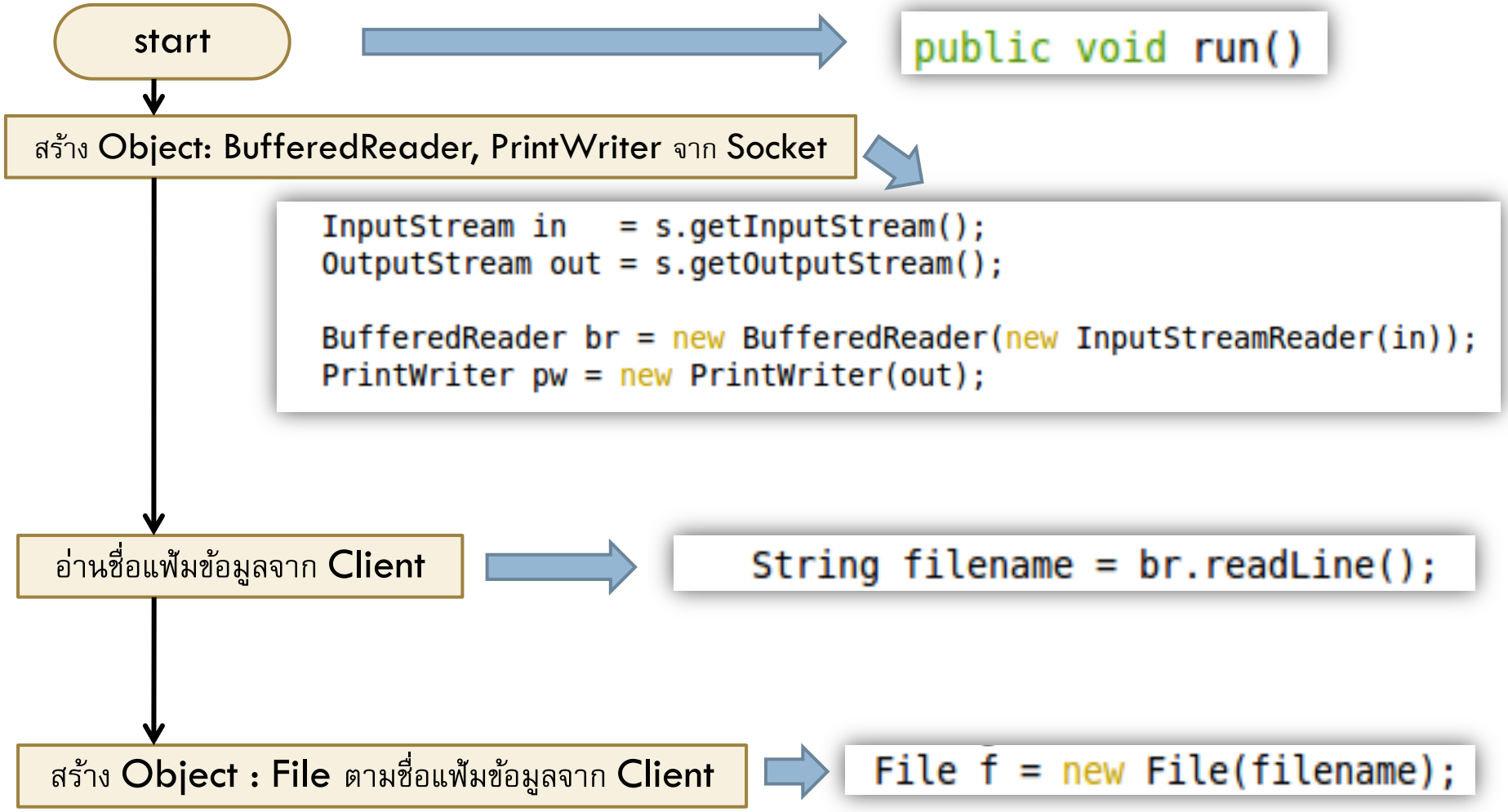
Source Code : FileDownload.java

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

public class FileDownload {
     public static void main(String[] args) {
             try {
                     Socket s = new Socket("127.0.0.1", 5678);
                     InputStream in   = s.getInputStream();
                     OutputStream out = s.getOutputStream();

                     BufferedReader br = new BufferedReader(new InputStreamReader(in));
                     PrintWriter pw = new PrintWriter(out);

                     pw.println(args[0]);
                     pw.flush();
                     String response = br.readLine();
                     if(response.equals("OK")){
                             File f = new File(args[0]);
                             FileOutputStream fout = new FileOutputStream(f);
                             byte[] b = new byte[65536];
                             int size;
                             while((size = in.read(b)) > 0) {
                                     fout.write(b, 0, size);
                             }
                             in.close();
                     } else {
                             System.out.println("No file found on server");
                     }
                     s.close();
             } catch(Exception e) { e.printStackTrace(); }

     }
}
```

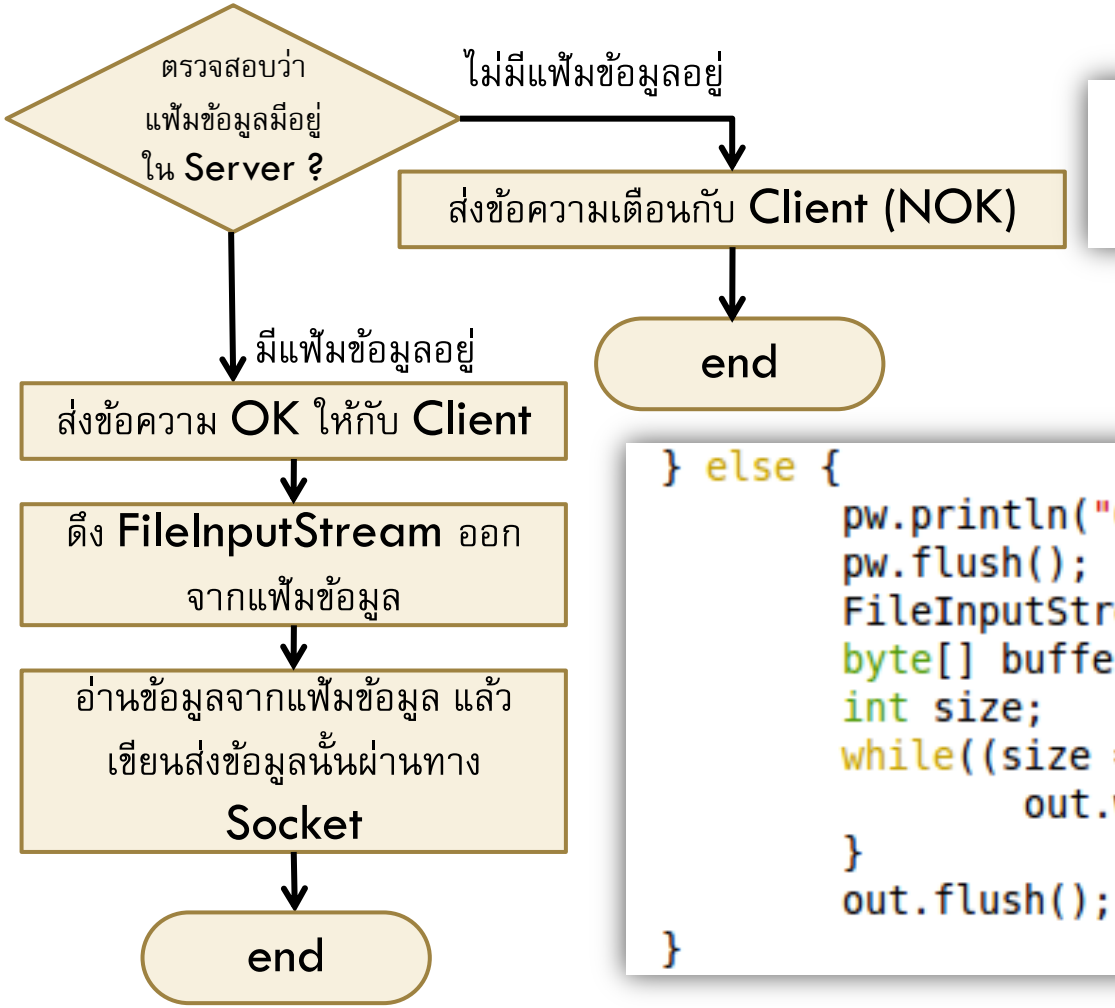
Flow Chart : Server (Download)



Flow Chart to Code (1)



Flow Chard to Code (2)




```
if(!f.exists()) {
        pw.println("NOK");
        pw.flush();
```

```
} else {
        pw.println("OK");
        pw.flush();
        FileInputStream fin = new FileInputStream(f);
        byte[] buffer = new byte[65536];
        int size;
        while((size = fin.read(buffer)) > 0) {
                out.write(buffer, 0, size);
        }
        out.flush();
}
```

Source Code : FileServerDownload.java

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

public class FileServerDownload implements Runnable {
     Socket s = null;

     public FileServerDownload(Socket s) {
             this.s = s;
     }

     public void run() {
             try {
                     InputStream in   = s.getInputStream();
                     OutputStream out = s.getOutputStream();

                     BufferedReader br = new BufferedReader(new InputStreamReader(in));
                     PrintWriter pw = new PrintWriter(out);

                     String filename = br.readLine();
                     File f = new File(filename);
                     if(!f.exists()) {
                             pw.println("NOK");
                             pw.flush();
                     } else {
                             pw.println("OK");
                             pw.flush();
                             FileInputStream fin = new FileInputStream(f);
                             byte[] buffer = new byte[65536];
                             int size;
                             while((size = fin.read(buffer)) > 0) {
                                     out.write(buffer, 0, size);
                             }
                             out.flush();
                     }
                     out.close();
                     s.close();
             } catch (Exception e) { e.printStackTrace(); }
     }
```

```
     public static void main(String[] args) {
             try {
                     ServerSocket serv = new ServerSocket(5678);
                     while(true) {
                             Socket s = serv.accept();
                             FileServerDownload fs = new FileServerDownload(s);
                             Thread t = new Thread(fs);
                             t.start();
                     }
             } catch(Exception e) { e.printStackTrace(); }
     }
}
```