

Client

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

public class Client {

    protected String serverName;
    protected int serverPort;
    protected String message;

    public Client(String serverName, int serverPort, String message) {
        this.serverName = serverName;
        this.serverPort = serverPort;
        this.message = message;
    }

    public void connect() {
        try (
            Socket socket = new Socket(serverName, serverPort);
            PrintWriter out = new PrintWriter(new GZIPOutputStream(socket.getOutputStream()), true);
        ) {
            out.println(message);
        } catch (UnknownHostException e) {
            System.err.println(e);
            System.exit(-1);
        }
    }
}
```

Client

```
    } catch (IOException e) {
        System.err.println(e);
        System.exit(-2);
    } catch (SecurityException e) {
        System.err.println(e);
        System.exit(-3);
    } catch (IllegalArgumentException e) {
        System.err.println(e);
        System.exit(-4);
    }
}

public static void main(String[] args) {
    Client c = new Client("127.0.0.01", 12345, "This is a test");
    c.connect();
}
}
```

Server

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

public class Server {

    protected int port;

    public Server(int port) {
        this.port = port;
    }

    public void serve() {
        try (
            ServerSocket serverSocket = new ServerSocket(port);
            Socket clientSocket = serverSocket.accept();
            InputStream in = clientSocket.getInputStream();
            FileOutputStream outFile = new FileOutputStream("Exercise.gz");
        ) {
            byte[] b = in.readAllBytes();
            outFile.write(b, 0, b.length);
            outFile.flush();
        } catch (IOException e) {
            System.err.println(e);
            System.exit(-2);
        }
    }
}
```

Server

```
} catch (SecurityException e) {  
    System.err.println(e);  
    System.exit(-3);  
}  
} catch (IllegalArgumentException e) {  
    System.err.println(e);  
    System.exit(-4);  
}  
} catch (IllegalBlockingModeException e) {  
    System.err.println(e);  
    System.exit(-6);  
}  
}  
  
public static void main(String[] args) {  
    Server s = new Server(12345);  
    s.serve();  
}  
}
```