

Bagian 1: Identifikasi Kesalahan

Contoh 1: Program Stream TCP

Kode dengan Kesalahan (FileServerTCP.java)

```
import java.io.*;

import java.net.*;

public class FileServerTCP {
    private static final int PORT = 12345;

    public static void main(String[] args) {
        try (ServerSocket serverSocket = new ServerSocket(PORT)) {
            System.out.println("Server file TCP berjalan pada port " + PORT);

            while (true) {
                Socket clientSocket = serverSocket.accept();
                System.out.println("Klien terhubung");

                // Menangani klien dalam thread terpisah
                new Thread(() -> {
                    try (InputStream inputStream = clientSocket.getInputStream();
                        DataInputStream dataInputStream = new DataInputStream(inputStream)) {

                        // Menerima nama file
                        String fileName = dataInputStream.readUTF();

                        // Kesalahan: Tidak memeriksa apakah fileName kosong atau tidak
                        System.out.println("Menerima file: " + fileName);

                        // Menerima dan menyimpan file
                        try (FileOutputStream fileOutputStream = new FileOutputStream(fileName)) {
                            byte[] buffer = new byte[4096];
                            int bytesRead;
                            while ((bytesRead = dataInputStream.read(buffer)) != -1) {
                                fileOutputStream.write(buffer, 0, bytesRead);
                            }
                            System.out.println("File berhasil diterima: " + fileName);
                        }

                    } catch (IOException e) {
                        System.err.println("Terjadi kesalahan saat menerima file: " + e.getMessage());
                    } finally {
                        try {
                            clientSocket.close();
                        } catch (IOException e) {
                        }
                    }
                }).start();
            }
        }
    }
}
```

```

        e.printStackTrace();
    }
}
}).start();
}
} catch (IOException e) {
    e.printStackTrace();
}
}
}

```

KESALAHAN YANG DITEMUKAN

1. Kesalahan Logika: Tidak memeriksa apakah fileName kosong atau tidak.
2. Kurangnya Penanganan Eksepsi: Tidak menangani potensi kesalahan saat menyimpan file.

Contoh 2: Program Datagram UDP

Kode dengan Kesalahan (ChatServerUDP.java)

```

import java.net.*;

public class ChatServerUDP {
    private static final int PORT = 8000;

    public static void main(String[] args) {
        try (DatagramSocket serverSocket = new DatagramSocket(PORT)) {
            byte[] buffer = new byte[1024];

            while (true) {
                DatagramPacket packet = new DatagramPacket(buffer, buffer.length);
                serverSocket.receive(packet);

                String message = new String(packet.getData(), 0, packet.getLength());
                System.out.println("Pesan diterima: " + message);
                // Kesalahan: Tidak mengirim balasan ke pengirim
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

KESALAHAN YANG DITEMUKAN

1. **Kesalahan Logika:** Server tidak mengirimkan balasan kepada klien setelah menerima pesan.

Bagian 2: Perbaiki Program

- Perbaiki Program Stream TCP FileServerTCP.java (Diperbaiki)

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

public class FileServerTCP {
    private static final int PORT = 12345;

    public static void main(String[] args) {
        try (ServerSocket serverSocket = new ServerSocket(PORT)) {
            System.out.println("Server file TCP berjalan pada port " + PORT);

            while (true) {
                Socket clientSocket = serverSocket.accept();
                System.out.println("Klien terhubung");

                // Menangani klien dalam thread terpisah
                new Thread(() -> {
                    try (InputStream inputStream = clientSocket.getInputStream();
                        DataInputStream dataInputStream = new DataInputStream(inputStream)) {

                        // Menerima nama file
                        String fileName = dataInputStream.readUTF();
                        if (fileName.isEmpty()) {
                            System.err.println("Nama file tidak valid.");
                            return; // Menghentikan eksekusi jika nama file tidak valid
                        }

                        System.out.println("Menerima file: " + fileName);

                        // Menerima dan menyimpan file
                        try (FileOutputStream fileOutputStream = new FileOutputStream(fileName)) {
                            byte[] buffer = new byte[4096];
                            int bytesRead;
                            while ((bytesRead = dataInputStream.read(buffer)) != -1) {
                                fileOutputStream.write(buffer, 0, bytesRead);
                            }
                            System.out.println("File berhasil diterima: " + fileName);
                        }
                    } catch (IOException e) {
                        System.err.println("Terjadi kesalahan saat menerima file: " + e.getMessage());
                    } finally {

```

```

        try {
            clientSocket.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    }).start();
}
} catch (IOException e) {
    e.printStackTrace();
}
}
}

```

- **Perbaikan Program Datagram UDP**
ChatServerUDP.java (Diperbaiki)

```

import java.net.*;

public class ChatServerUDP {
    private static final int PORT = 8000;

    public static void main(String[] args) {
        try (DatagramSocket serverSocket = new DatagramSocket(PORT)) {
            byte[] buffer = new byte[1024];

            while (true) {
                DatagramPacket packet = new DatagramPacket(buffer, buffer.length);
                serverSocket.receive(packet);

                String message = new String(packet.getData(), 0, packet.getLength());
                System.out.println("Pesan diterima: " + message);

                // Mengirim balasan ke klien
                String responseMessage = "Pesan diterima: " + message;
                byte[] responseBuffer = responseMessage.getBytes();
                DatagramPacket responsePacket = new DatagramPacket(
                    responseBuffer, responseBuffer.length,
                    packet.getAddress(), packet.getPort()
                );
                serverSocket.send(responsePacket);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}

```

Bagian 3: Dokumentasikan Perubahan

Perubahan pada FileServerTCP.java

Menambahkan Validasi Nama File: Memeriksa apakah nama file yang diterima kosong. Jika kosong, program akan menghentikan eksekusi.

Alasan: Mencegah program mencoba untuk menyimpan file dengan nama tidak valid.

Menangani Kesalahan dalam Penyimpanan File: Menambahkan penanganan eksepsi saat menyimpan file.

Alasan: Memastikan program tidak crash saat terjadi kesalahan dalam menyimpan file.

Perubahan pada ChatServerUDP.java

Menambahkan Pengiriman Balasan ke Klien: Mengirim balasan kepada klien setelah menerima pesan.

Alasan: Memberikan umpan balik kepada klien bahwa pesan telah diterima, meningkatkan interaksi dan kejelasan komunikasi.

Bagian 4: Uji Coba

Uji Coba untuk FileServerTCP

Jalankan FileServerTCP.java.

Jalankan FileClientTCP.java, dan masukkan path file yang valid.

Periksa apakah file berhasil diterima dan disimpan di server.

Uji Coba untuk ChatServerUDP

Jalankan ChatServerUDP.java.

Jalankan beberapa instansi dari klien UDP yang mengirimkan pesan ke server.

Periksa apakah server menerima pesan dan mengirimkan balasan ke klien.

Bagian 3: Dokumentasikan Perubahan

Perubahan pada FileServerTCP.java

Menambahkan Validasi Nama File: Memeriksa apakah nama file yang diterima kosong. Jika kosong, program akan menghentikan eksekusi.

Alasan: Mencegah program mencoba untuk menyimpan file dengan nama tidak valid.

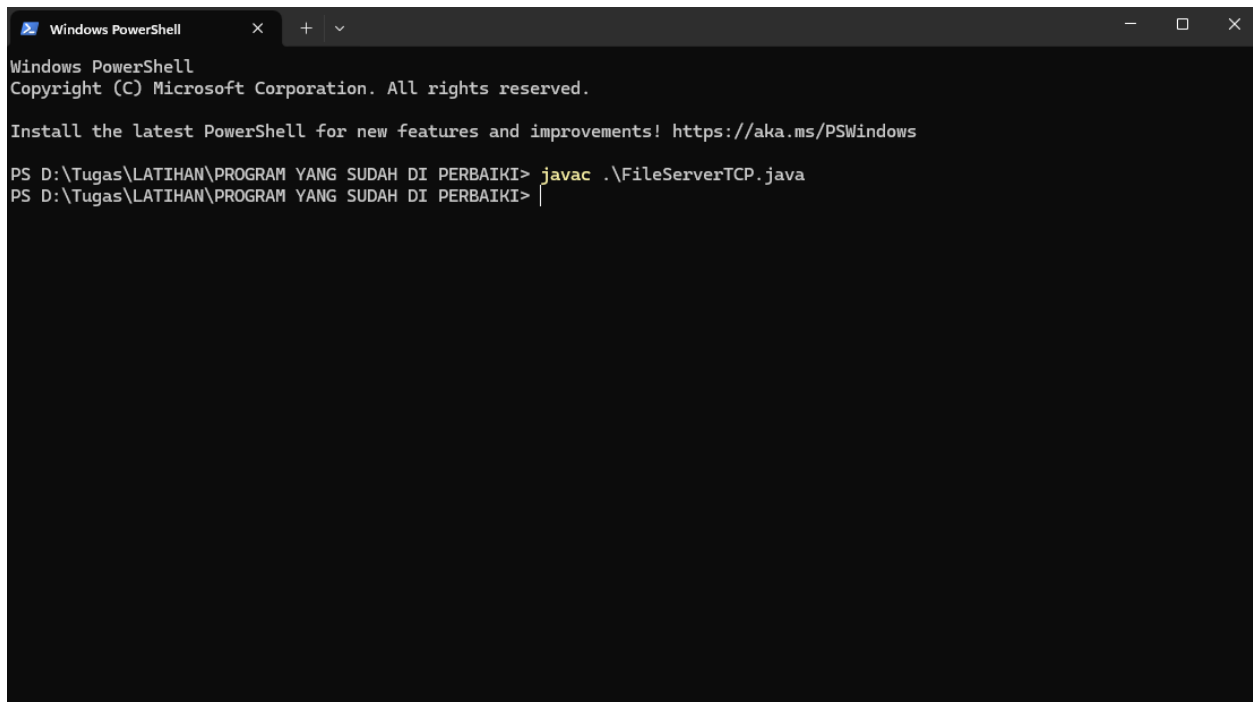
Menangani Kesalahan dalam Penyimpanan File: Menambahkan penanganan eksepsi saat menyimpan file.

Alasan: Memastikan program tidak crash saat terjadi kesalahan dalam menyimpan file.

Perubahan pada ChatServerUDP.java

Menambahkan Pengiriman Balasan ke Klien: Mengirim balasan kepada klien setelah menerima pesan.

Alasan: Memberikan umpan balik kepada klien bahwa pesan telah diterima, meningkatkan interaksi dan kejelasan komunikasi.

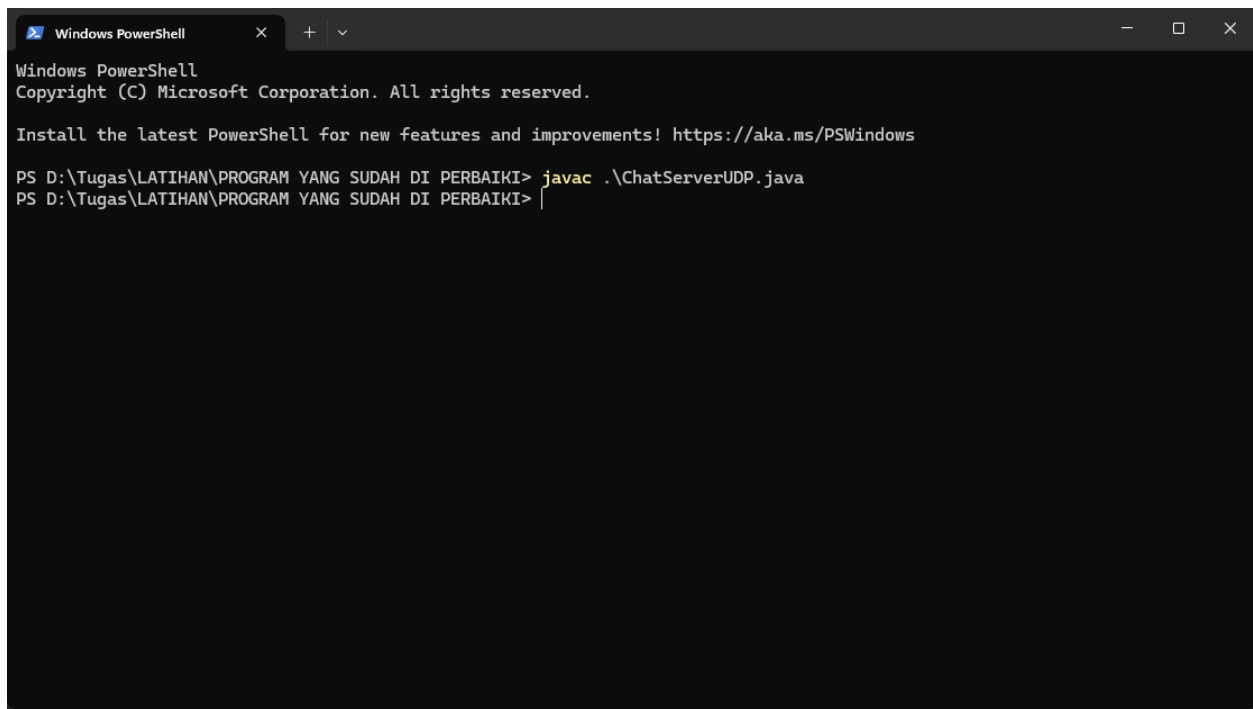
A screenshot of a Windows PowerShell terminal window. The title bar at the top says "Windows PowerShell" with standard window controls. The terminal content shows the Windows PowerShell logo, copyright information for Microsoft Corporation, and a link to update PowerShell. The current directory is "D:\Tugas\LATIHAN\PROGRAM YANG SUDAH DI PERBAIKI". The user has entered the command "javac .\FileServerTCP.java" and the prompt is waiting for the next input.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Tugas\LATIHAN\PROGRAM YANG SUDAH DI PERBAIKI> javac .\FileServerTCP.java
PS D:\Tugas\LATIHAN\PROGRAM YANG SUDAH DI PERBAIKI> |
```

Gambar compile program FileServerTCP.java yang telah diperbaiki berjalan tanpa error



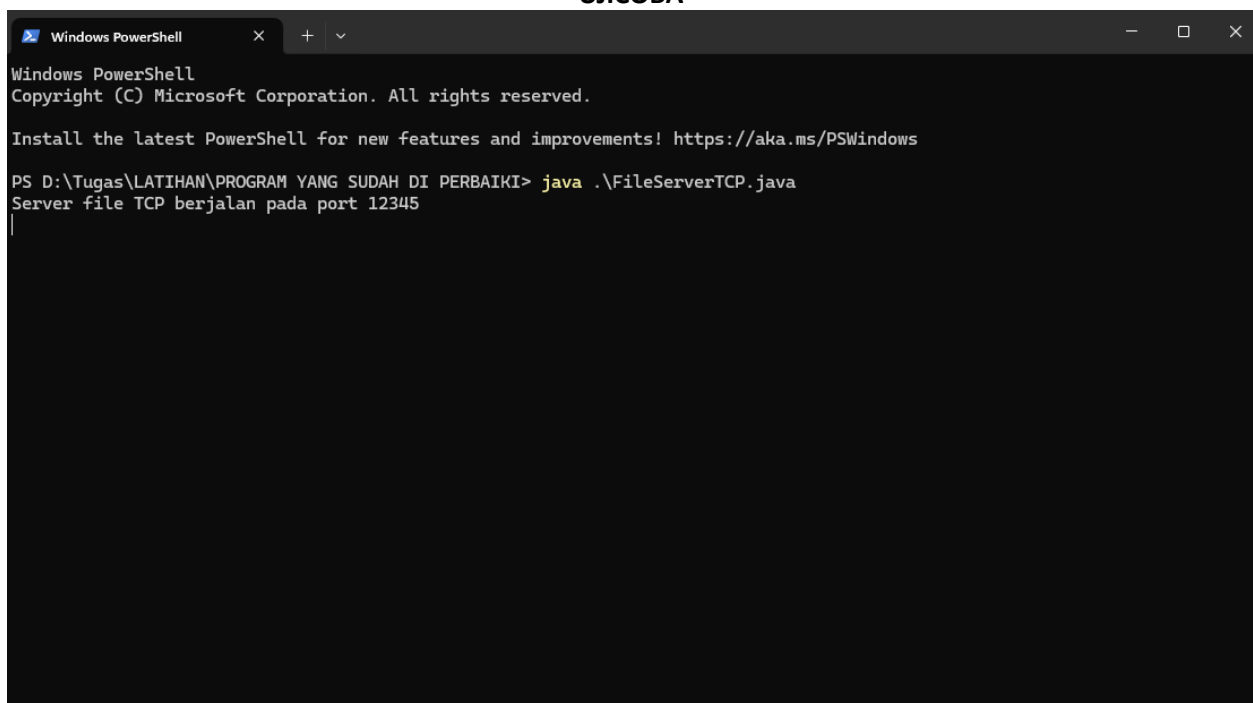
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Tugas\LATIHAN\PROGRAM YANG SUDAH DI PERBAIKI> javac .\ChatServerUDP.java
PS D:\Tugas\LATIHAN\PROGRAM YANG SUDAH DI PERBAIKI> |
```

Gambar compile program ChatServerUDP.java yang telah diperbaiki berjalan tanpa error

UJICOBA

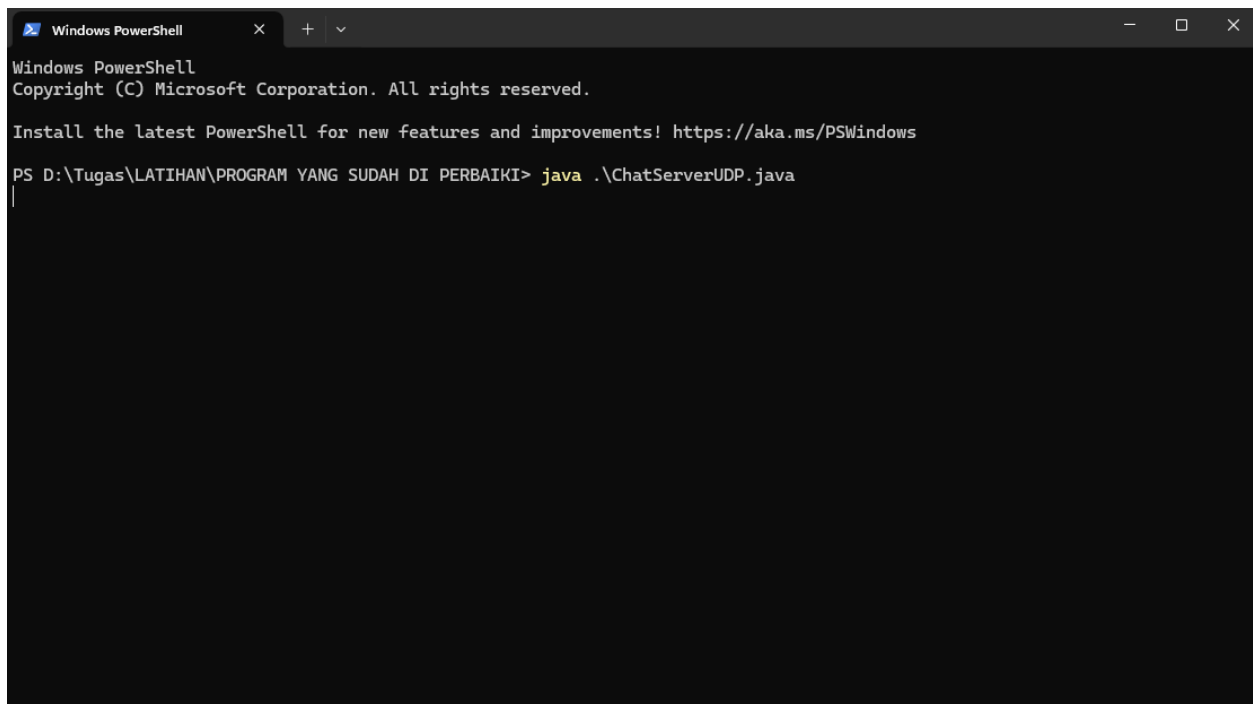


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Tugas\LATIHAN\PROGRAM YANG SUDAH DI PERBAIKI> java .\FileServerTCP.java
Server file TCP berjalan pada port 12345
|
```

Ujicoba Program FileServerTCP.java berjalan dengan baik



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Tugas\LATIHAN\PROGRAM YANG SUDAH DI PERBAIKI> java .\ChatServerUDP.java
|
```

Ujicoba Program ChatServerUDP.java berjalan dengan baik