

Nama : Tri Dinda Rukmana

Nim : 09011182328015

Kelas : SK1C

1. Hitung Total Harga

The screenshot shows the VS Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Editor:** The active file is `HitungTotalHarga.java`. The code initializes a Scanner, prompts for item count, and item price, then calculates total price before discount.
- Explorer:** Shows a project structure under `UAS` containing files: `AutentikasiPengguna.java`, `DeretFibonacci.java`, `Faktorisasi.java`, and `HitungTotalHarga.java`.
- Outline:** Shows the outline of the current file with sections like `OUTLINE` and `TIMELINE`.
- VS Code PETS:** A section for managing extensions.

```
1 import java.util.Scanner;
2
3 public class HitungTotalHarga {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6
7         // Input jumlah barang yang dibeli oleh pelanggan
8         System.out.print("Masukkan jumlah barang yang dibeli: ");
9         int jumlahBarang = scanner.nextInt();
10
11        // Input harga per barang
12        System.out.print("Masukkan harga per barang: ");
13        double hargaPerBarang = scanner.nextDouble();
14
15        // Hitung total harga sebelum diskon
16        double totalHarga = jumlahBarang * hargaPerBarang;
17
18        // Hitung diskon berdasarkan jumlah pembelian
19        double diskon = 0.0;
20        if (jumlahBarang >= 5 && jumlahBarang <= 10) {
21            diskon = 0.05;
22        } else if (jumlahBarang >= 11 && jumlahBarang <= 20) {
23            diskon = 0.1;
24        } else if (jumlahBarang > 20) {
25            diskon = 0.2;
26        }
27
28        // Hitung total harga setelah diskon
29        double totalHargaSetelahDiskon = totalHarga - (totalHarga * diskon);
30
31        // Tampilkan total harga setelah diskon
32        System.out.println("Total harga setelah diskon: " + totalHargaSetelahDiskon);
33
34        // Tutup scanner
35        scanner.close();
36    }
37
38
39 }
```

The screenshot shows the VS Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Editor:** The active file is `HitungTotalHarga.java`. The code now includes a calculation for the total price after applying a discount based on the number of items purchased.
- Explorer:** Shows a project structure under `UAS` containing files: `AutentikasiPengguna.java`, `DeretFibonacci.java`, `Faktorisasi.java`, and `HitungTotalHarga.java`.
- Outline:** Shows the outline of the current file with sections like `OUTLINE` and `TIMELINE`.
- VS Code PETS:** A section for managing extensions.
- JAVA PROJECTS:** Shows the status of the Java projects.

```
1 import java.util.Scanner;
2
3 public class HitungTotalHarga {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6
7         // Input jumlah barang yang dibeli oleh pelanggan
8         System.out.print("Masukkan jumlah barang yang dibeli: ");
9         int jumlahBarang = scanner.nextInt();
10
11        // Input harga per barang
12        System.out.print("Masukkan harga per barang: ");
13        double hargaPerBarang = scanner.nextDouble();
14
15        // Hitung total harga sebelum diskon
16        double totalHarga = jumlahBarang * hargaPerBarang;
17
18        // Hitung diskon berdasarkan jumlah pembelian
19        double diskon = 0.0;
20        if (jumlahBarang >= 5 && jumlahBarang <= 10) {
21            diskon = 0.05;
22        } else if (jumlahBarang >= 11 && jumlahBarang <= 20) {
23            diskon = 0.1;
24        } else if (jumlahBarang > 20) {
25            diskon = 0.2;
26        }
27
28        // Hitung total harga setelah diskon
29        double totalHargaSetelahDiskon = totalHarga - (totalHarga * diskon);
30
31        // Tampilkan total harga setelah diskon
32        System.out.println("Total harga setelah diskon: " + totalHargaSetelahDiskon);
33
34        // Tutup scanner
35        scanner.close();
36    }
37
38
39 }
```

Hasil Terminal :

The screenshot shows a Java IDE interface with the following details:

- File Explorer:** Shows a project named "UAS" containing files: AutentikasiPengguna.java, DeretFibonacci.java, Faktorisasi.java, and HitungTotalHarga.java.
- Code Editor:** The active file is HitungTotalHarga.java, displaying the following code:

```
27 // Hitung total harga setelah diskon
28 double totalHargaSetelahDiskon = totalHarga - (totalHarga * diskon);
29
30 // Tampilkan total harga setelah diskon
31 System.out.println("Total harga setelah diskon: " + totalHargaSetelahDiskon);
32
33 // Tutup scanner
34 scanner.close();
35
36 }
37
38
39 }
```

- Terminal:** The terminal window shows the command being run and the output:

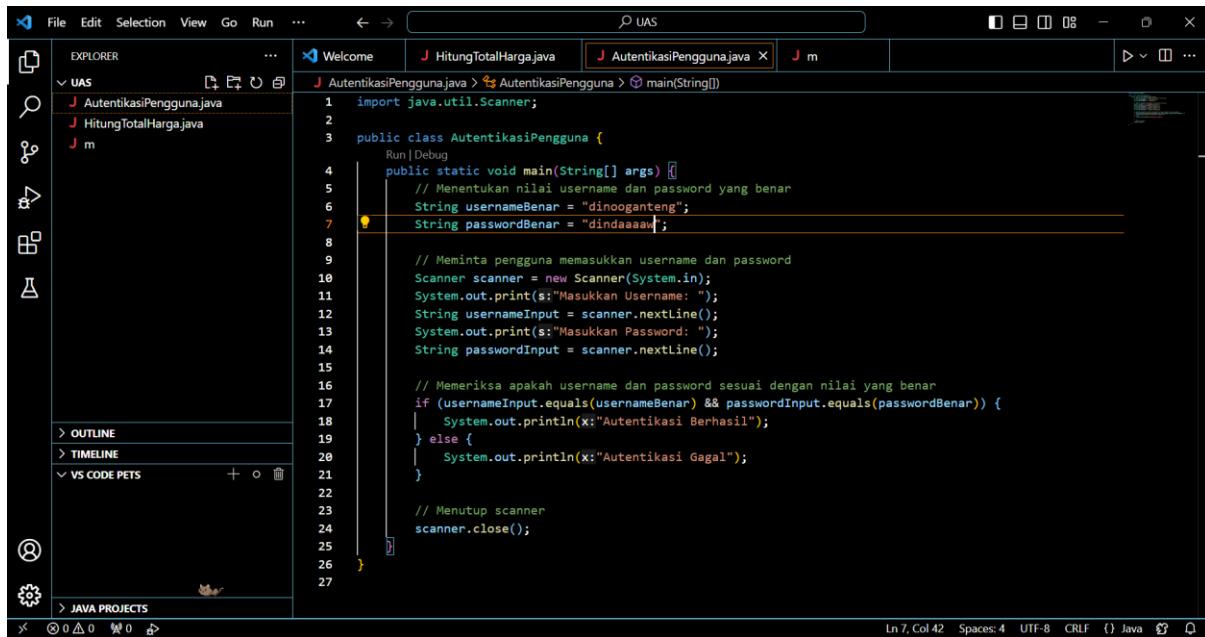
```
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b560b440e78311565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'HitungTotalHarga'
Masukkan jumlah barang yang dibeli: 5
Masukkan harga per barang: 155.000
Total harga setelah diskon: 2092.5
```

- Run Options:** A sidebar on the right lists run configurations for other Java files in the project.

This screenshot is nearly identical to the first one, showing the same Java IDE interface and terminal output. The only difference is the input provided in the terminal:

```
\UAS_bb14e730\bin\ 'HitungTotalHarga'
Masukkan jumlah barang yang dibeli: 3
Masukkan harga per barang: 155.000
Total harga setelah diskon: 465.0
```

2. Autentikasi Pengguna



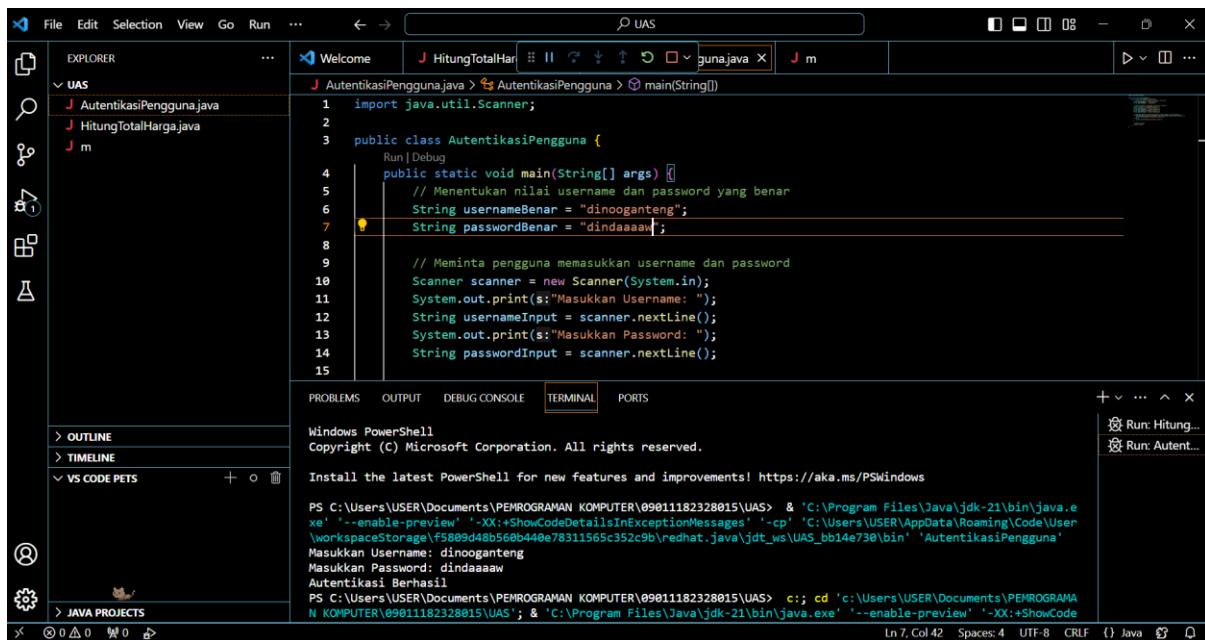
The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a project named "UAS" containing three files: "AutentikasiPengguna.java", "HitungTotalHarga.java", and "m".
- Code Editor:** Displays the "AutentikasiPengguna.java" file with the following code:

```
1 import java.util.Scanner;
2
3 public class AutentikasiPengguna {
4     public static void main(String[] args) {
5         // Menentukan nilai username dan password yang benar
6         String usernameBenar = "dinooganteng";
7         String passwordBenar = "dindaaaaaw";
8
9         // Meminta pengguna memasukkan username dan password
10        Scanner scanner = new Scanner(System.in);
11        System.out.print("Masukkan Username: ");
12        String usernameInput = scanner.nextLine();
13        System.out.print("Masukkan Password: ");
14        String passwordInput = scanner.nextLine();
15
16        // Memeriksa apakah username dan password sesuai dengan nilai yang benar
17        if (usernameInput.equals(usernameBenar) && passwordInput.equals(passwordBenar)) {
18            System.out.println("Autentikasi Berhasil");
19        } else {
20            System.out.println("Autentikasi Gagal");
21        }
22
23        // Menutup scanner
24        scanner.close();
25    }
26}
27
```

- Terminal:** Shows the command "ln 7, Col 42 Spaces: 4 UTF-8 CRLF () Java".

Hasil Terminal :



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a project named "UAS" containing three files: "AutentikasiPengguna.java", "HitungTotalHarga.java", and "m".
- Code Editor:** Displays the "AutentikasiPengguna.java" file with the same code as the previous screenshot.
- Terminal:** Shows the command "ln 7, Col 42 Spaces: 4 UTF-8 CRLF () Java". Below it, the terminal output is displayed:

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

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

PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b560b440e78311565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'AutentikasiPengguna'
Masukkan Username: dinooganteng
Masukkan Password: dindaaaaaw
Autentikasi Berhasil
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> c:; cd 'c:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS'; & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCode
```

```

import java.util.Scanner;
public class AutentikasiPengguna {
    public static void main(String[] args) {
        // Menentukan nilai username dan password yang benar
        String usernameBenar = "dinooganteng";
        String passwordBenar = "dindaaaaaw";
        // Meminta pengguna memasukkan username dan password
        Scanner scanner = new Scanner(System.in);
        System.out.print("Masukkan Username: ");
        String usernameInput = scanner.nextLine();
        System.out.print("Masukkan Password: ");
        String passwordInput = scanner.nextLine();
    }
}

```

Terminal output:

```

xe' '--enable-preview' '--XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b560b440e78311565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'AutentikasiPengguna'
Masukkan Username: dinooganteng
Masukkan Password: dindaaaaaw
Autentikasi Berhasil
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> c;; cd 'c:\Users\USER\Documents\PEMROGRAMA N KOMPUTER\09011182328015\UAS'; & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '--XX:+ShowCode DetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b560b440e78 31565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'AutentikasiPengguna'
Masukkan Username: dinoganteng
Masukkan Password: dindaw
Autentikasi Gagal
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS>

```

3. Deret Fibonacci

```

import java.util.Scanner;
public class DeretFibonacci {
    public static void main(String[] args) {
        // Meminta pengguna memasukkan nilai n
        Scanner scanner = new Scanner(System.in);
        System.out.print("Masukkan nilai n untuk deret Fibonacci: ");
        int n = scanner.nextInt();
        // Menampilkan deret Fibonacci hingga suku ke-n
        System.out.println("Deret Fibonacci hingga suku ke-" + n + ":");
        for (int i = 0; i < n; i++) {
            System.out.print(fibonacci(i) + " ");
        }
        // Menutup scanner
        scanner.close();
    }

    // Fungsi untuk menghitung nilai suku ke-n dalam deret Fibonacci
    public static int fibonacci(int n) {
        if (n <= 1) {
            return n;
        } else {
            return fibonacci(n - 1) + fibonacci(n - 2);
        }
    }
}

```

Terminal output:

```

Ln 28, Col 2   Spaces: 4   UTF-8   CRLF   () Java   ⚡   ⌂

```

Hasil Terminal:

The screenshot shows the VS Code interface with the terminal tab selected. The terminal window displays the following Java code for calculating Fibonacci numbers:

```
import java.util.Scanner;
public class DeretFibonacci {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Masukkan nilai n untuk deret Fibonacci: ");
        int n = scanner.nextInt();

        // Menampilkan deret Fibonacci hingga suku ke-n
        System.out.println("Deret Fibonacci hingga suku ke-" + n + ":");
        for (int i = 0; i < n; i++) {
            System.out.print(fibonacci(i) + " ");
        }
    }
}
```

Below the code, the terminal shows the command run and its output:

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

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

PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '--showCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5889d4ab568b44e78311565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'DeretFibonacci'
Masukkan nilai n untuk deret Fibonacci: 8
Deret Fibonacci hingga suku ke-8:
0 1 1 2 3 5 8 13
```

At the bottom right of the terminal, there are three run configurations: 'Run: Hitung...', 'Run: Autent...', and 'Run: Dereti...'. The status bar at the bottom indicates: Ln 28 Col 2 Spaces:4 UTF-8 CR/LF {} Java ⚡

4. Faktorisasi

The screenshot shows the VS Code interface with the terminal tab selected. The terminal window displays the following Java code for factoring integers:

```
import java.util.Scanner;
public class Faktorisasi {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Masukkan bilangan bulat positif: ");
        int bilangan = scanner.nextInt();

        // Menampilkan faktorisasi bilangan
        System.out.print("Faktorisasi " + bilangan + ": ");
        faktorisasi(bilangan);

        // Menutup scanner
        scanner.close();
    }

    // Fungsi untuk melakukan faktorisasi bilangan
    public static void faktorisasi(int n) {
        for (int i = 2; i <= n; i++) {
            while (n % i == 0) {
                System.out.print(i);
                n /= i;
                if (n > 1) {
                    System.out.print(" * ");
                }
            }
        }
    }
}
```

Below the code, the terminal shows the command run and its output:

```
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '--showCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5889d4ab568b44e78311565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'Faktorisasi'
Masukkan bilangan bulat positif: 100
Faktorisasi 100 : 2 * 2 * 5 * 5
```

At the bottom right of the terminal, there are three run configurations: 'Run: Hitung...', 'Run: Autent...', and 'Run: Dereti...'. The status bar at the bottom indicates: Ln 29 Col 16 Spaces:4 UTF-8 CR/LF {} Java ⚡

Hasil Terminal :

The screenshot shows the VS Code interface with the terminal tab selected. The terminal window displays the output of running the `Faktorisasi.java` program. The code prompts the user to enter a positive integer, calculates its factors, and prints the result. The terminal also shows the PowerShell environment and some system messages.

```
riksaUrutanKurungBukaTutup.java J Bukujava J PalindromCheckjava J KalkulatorSederhana.java J Faktorisasi.java
J Faktorisasi.java > Faktorisasi > faktorisasi(int)
1 import java.util.Scanner;
2
3 public class Faktorisasi {
Run | Debug
4     public static void main(String[] args) {
5         // Meminta pengguna memasukkan bilangan bulat positif
6         Scanner scanner = new Scanner(System.in);
7         System.out.print("Masukkan bilangan bulat positif: ");
8         int bilangan = scanner.nextInt();
9
10        // Menampilkan faktorisasi bilangan
11        System.out.print("Faktorisasi " + bilangan + ": ");
12        faktorisasi(bilangan);
13
14        // Menutup scanner
15        scanner.close();
PROBLEMS ① OUTPUT DEBUG CONSOLE TERMINAL PORTS
W indows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

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

PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\javexe' '--enable-preview' '--XX:+ShowCodeDetailsInExceptionMessages' '--cp' 'C:\Users\USER\AppData\Roaming\Code\UAS\workspaceStorage\f5889d48b56eb44e078311565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'Faktorisasi'
Masukkan bilangan bulat positif: 50
Faktorisasi 50: 2 * 5 * 5
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS>
In 29, Col 6 Spaces: 4 UTF-8 CRLF {} Java ⚡
```

5. KalkulatorSederhana

The screenshot shows the VS Code interface with the terminal tab selected. The terminal window displays the output of running the `KalkulatorSederhana.java` program. The code prompts the user to enter two numbers and a choice of arithmetic operation (addition, subtraction, multiplication, or division), then performs the calculation and prints the result. The terminal also shows the PowerShell environment and some system messages.

```
elcome J HitungTotalHarga.java J AutentikasiPengguna.java J m J KalkulatorSederhana.java J Faktorisasi D < < ...
J KalkulatorSederhana.java > KalkulatorSederhana > pembagian(double, double)
1 import java.util.Scanner;
2
3 public class KalkulatorSederhana {
Run | Debug
4     public static void main(String[] args) {
5         // Meminta pengguna memasukkan dua bilangan
6         Scanner scanner = new Scanner(System.in);
7         System.out.print("Masukkan bilangan pertama: ");
8         double bilangan1 = scanner.nextDouble();
9         System.out.print("Masukkan bilangan kedua: ");
10        double bilangan2 = scanner.nextDouble();
11
12        // Menampilkan menu operasi
13        System.out.println("Pilih operasi:");
14        System.out.println("1. Penjumlahan");
15        System.out.println("2. Pengurangan");
16        System.out.println("3. Perkalian");
17        System.out.println("4. Pembagian");
18
19        // Meminta pengguna memilih operasi
20        System.out.print("Masukkan nomor operasi: ");
21        int pilihan = scanner.nextInt();
22
23        // Melakukan operasi sesuai pilihan pengguna
24        switch (pilihan) {
25            case 1:
26                System.out.println("Hasil penjumlahan: " + penjumlahan(bilangan1, bilangan2));
27                break;
28            case 2:
29                System.out.println("Hasil pengurangan: " + pengurangan(bilangan1, bilangan2));
30                break;
31            case 3:
32                System.out.println("Hasil perkalian: " + perkalian(bilangan1, bilangan2));
33                break;
34            case 4:
35                System.out.println("Hasil pembagian: " + pembagian(bilangan1, bilangan2));
36                break;
37            default:
38                System.out.println("Operasi tidak dikenali.");
39        }
PROBLEMS ① OUTPUT DEBUG CONSOLE TERMINAL PORTS
W indows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

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

PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\javexe' '--enable-preview' '--XX:+ShowCodeDetailsInExceptionMessages' '--cp' 'C:\Users\USER\AppData\Roaming\Code\UAS\workspaceStorage\f5889d48b56eb44e078311565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'KalkulatorSederhana'
In 68, Col 6 Spaces: 4 UTF-8 CRLF {} Java ⚡
```

```
File Edit Selection View Go Run ... ← → ⌂ UAS
EXPLORER ... /welcome J HitungTotalHarga.java J AutentikasiPengguna.java J m J KalkulatorSederhana.java X J Faktorisasi.java ...
UAS J AutentikasiPengguna.java J DeretFibonacci.java J Faktorisasi.java J HitungTotalHarga.java J KalkulatorSederhana.java J m
> OUTLINE
> TIMELINE
VS CODE PETS + o 🗑
JAVA PROJECTS 🐱

J KalkulatorSederhana.java > KalkulatorSederhana > pembagian(double, double)
29 |     System.out.println("Hasil pengurangan: " + pengurangan(bilangan1, bilangan2));
30 |     break;
31 |     case 3:
32 |         System.out.println("Hasil perkalian: " + perkalian(bilangan1, bilangan2));
33 |         break;
34 |     case 4:
35 |         System.out.println("Hasil pembagian: " + pembagian(bilangan1, bilangan2));
36 |         break;
37 |     default:
38 |         System.out.println(x:"Pilihan operasi tidak valid.");
39 |
40 |     // Menutup scanner
41 |     scanner.close();
42 |
43 |
44 |
45 |     // Metode untuk penjumlahan
46 |     public static double penjumlahan(double a, double b) {
47 |         return a + b;
48 |     }
49 |
50 |     // Metode untuk pengurangan
51 |     public static double pengurangan(double a, double b) {
52 |         return a - b;
53 |     }
54 |
55 |     // Metode untuk perkalian
56 |     public static double perkalian(double a, double b) {
57 |         return a * b;
58 |
59 |
60 |         // Metode untuk pembagian
61 |         public static double pembagian(double a, double b) {
62 |             if (b != 0) {
63 |                 return a / b;
64 |             } else {
65 |                 System.out.println(x:"Pembagian oleh nol tidak valid.");
66 |                 return Double.NaN; // NaN (Not a Number) sebagai hasil pembagian tidak valid
67 |             }
68 |         }
69 |
70 |     }
```

```
File Edit Selection View Go Run ... ← → ⌂ UAS
EXPLORER ... /welcome J HitungTotalHarga.java J AutentikasiPengguna.java J m J KalkulatorSederhana.java X J Faktorisasi.java ...
UAS J AutentikasiPengguna.java J DeretFibonacci.java J Faktorisasi.java J HitungTotalHarga.java J KalkulatorSederhana.java J m
> OUTLINE
> TIMELINE
VS CODE PETS + o 🗑
JAVA PROJECTS 🐱

J KalkulatorSederhana.java > ...
58 |
59 |
60 |         // Metode untuk pembagian
61 |         public static double pembagian(double a, double b) {
62 |             if (b != 0) {
63 |                 return a / b;
64 |             } else {
65 |                 System.out.println(x:"Pembagian oleh nol tidak valid.");
66 |                 return Double.NaN; // NaN (Not a Number) sebagai hasil pembagian tidak valid
67 |             }
68 |         }
69 |
70 |     }
```

Hasil Terminal :

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows a project named "UAS" containing several Java files: AutentikasiPengguna.java, DeretFibonacci.java, Faktorisasi.java, HitungTotalHarga.java, KalkulatorSederhana.java, and m.
- Code Editor:** Displays the content of KalkulatorSederhana.java. The code prompts the user to input two numbers and then prints a menu of operations (Penjumlahan, Pengurangan, Perkalian, Pembagian). It then asks for the operation choice and performs the calculation.
- Terminal:** Shows the output of running the program. It asks for the first number (Masukkan bilangan pertama: 46), the second number (Masukkan bilangan kedua: 46), and the operation choice (Pilih operasi: 1. Penjumlahan). The output indicates that the result is 2300.0.
- Run Configuration:** A context menu in the terminal area lists run configurations for various Java files in the UAS project.

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows a project named "UAS" containing several Java files: AutentikasiPengguna.java, DeretFibonacci.java, Faktorisasi.java, HitungTotalHarga.java, KalkulatorSederhana.java, and m.
- Code Editor:** Displays the content of KalkulatorSederhana.java. The code is identical to the one in the previous screenshot, prompting for two numbers and a menu of operations.
- Terminal:** Shows the output of running the program. It asks for the first number (Masukkan bilangan pertama: 50), the second number (Masukkan bilangan kedua: 46), and the operation choice (Pilih operasi: 3. Perkalian). The output indicates that the result is 2300.0.
- Run Configuration:** A context menu in the terminal area lists run configurations for various Java files in the UAS project.

6. PalindromCheck

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a project named "UAS" containing files: AkunPengguna.java, AutentikasiPengguna.java, m, PalindromCheck.java, and Buku.java.
- Code Editor:** The active file is PalindromCheck.java. The code implements a class that checks if a given string is a palindrome. It uses a Scanner to read input from the user, removes whitespace, and then compares characters from both ends of the string to see if they match.
- Status Bar:** Shows the current line (Ln 34), column (Col 2), and other settings like spaces: 4, UTF-8, CRLF, Java, and a search icon.

```
import java.util.Scanner;
public class PalindromCheck {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Masukkan kata atau frase: ");
        String input = scanner.nextLine();

        // Menghapus spasi dan mengubah ke huruf kecil untuk mempermudah pengecekan
        String cleanedInput = input.replaceAll("\\s", "").toLowerCase();

        // Memanggil metode untuk mengecek apakah kata atau frase adalah palindrom
        if (isPalindrome(cleanedInput)) {
            System.out.println("'" + input + "' adalah palindrom.");
        } else {
            System.out.println("'" + input + "' bukan palindrom.");
        }

        // Menutup scanner
        scanner.close();
    }

    // Metode untuk mengecek apakah suatu kata atau frase adalah palindrom
    public static boolean isPalindrome(String str) {
        int length = str.length();
        for (int i = 0; i < length / 2; i++) {
            if (str.charAt(i) != str.charAt(length - i - 1)) {
                return false;
            }
        }
        return true;
    }
}
```

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows a project named "UAS" containing files: AkunPengguna.java, AutentikasiPengguna.java, Buku.java, DeretFibonacci.java, Faktoriasi.java, HitungTotalHarga.java, KalkulatorSederhana.java, m, and PalindromCheck.java.
- Code Editor:** The active file is PalindromCheck.java. The cursor is positioned at the closing brace of the isPalindrome method. The code is identical to the one in the previous screenshot.
- Status Bar:** Shows the current line (Ln 34), column (Col 2), and other settings like spaces: 4, UTF-8, CRLF, Java, and a search icon.

```
import java.util.Scanner;
public class PalindromCheck {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Masukkan kata atau frase: ");
        String input = scanner.nextLine();

        // Menghapus spasi dan mengubah ke huruf kecil untuk mempermudah pengecekan
        String cleanedInput = input.replaceAll("\\s", "").toLowerCase();

        // Memanggil metode untuk mengecek apakah kata atau frase adalah palindrom
        if (isPalindrome(cleanedInput)) {
            System.out.println("'" + input + "' adalah palindrom.");
        } else {
            System.out.println("'" + input + "' bukan palindrom.");
        }

        // Menutup scanner
        scanner.close();
    }

    // Metode untuk mengecek apakah suatu kata atau frase adalah palindrom
    public static boolean isPalindrome(String str) {
        int length = str.length();
        for (int i = 0; i < length / 2; i++) {
            if (str.charAt(i) != str.charAt(length - i - 1)) {
                return false;
            }
        }
        return true;
    }
}
```

Hasil Terminal :

The screenshot shows the VS Code interface with the terminal tab selected. The terminal window displays the following output:

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

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

PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b560b440e7831565c352c9b\redhat\java\jdt_ws\UAS_bb14e730\bin' 'PalindromCheck'
Masukkan kata atau frase: kakak
"kakak" adalah palindrom.

PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS>
```

7. Buku

The screenshot shows the VS Code interface with the terminal tab selected. The terminal window displays the following output:

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

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

PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b560b440e7831565c352c9b\redhat\java\jdt_ws\UAS_bb14e730\bin' 'Buku'
public class Buku {
    private String judul;
    private String penulis;
    private int tahunTerbit;
    private boolean tersedia;

    // Konstruktor untuk inisialisasi atribut buku
    public Buku(String judul, String penulis, int tahunTerbit) {
        this.judul = judul;
        this.penulis = penulis;
        this.tahunTerbit = tahunTerbit;
        this.tersedia = true; // Saat pertama kali, buku tersedia untuk dipinjam
    }

    // Metode untuk menampilkan informasi buku
    public void tampilanInfoBuku() {
        System.out.println("Judul: " + judul);
        System.out.println("Penulis: " + penulis);
        System.out.println("Tahun Terbit: " + tahunTerbit);
        System.out.println("Status: " + (tersedia ? "Tersedia" : "Dipinjam"));
    }

    // Metode untuk meminjam buku
    public void pinjamBuku() {
        if (tersedia) {
            System.out.println("Buku berhasil dipinjam.");
            tersedia = false;
        } else {
            System.out.println("Maaf, buku sedang tidak tersedia.");
        }
    }
}
```

Hasil Terminal :

The screenshot shows the Visual Studio Code interface with the following details:

- EXPLORER**: Shows a project named "UAS" containing several Java files: AkunPengguna.java, AutentikasiPengguna.java, Buku.java, DeretFibonacci.java, Faktorisasi.java, HitungTotalHarga.java, KalkulatorSederhana.java, and m.
- CODE**: The current file is PalindromCheck.java, which contains the following code:

```
27 |     for (int i = 0; i < length / 2; i++) {  
28 |         if (str.charAt(i) != str.charAt(length - i - 1)) {  
29 |             return false;  
30 |         }  
31 |     }  
32 |     return true;  
33 | }  
34 }  
35 }
```

- TERMINAL**: Displays the Windows PowerShell terminal with the following output:

```
Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
  
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows  
  
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b560b440e7831565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'Buku'  
Informasi buku sebelum dipinjam:  
Judul: Java Programming  
Penulis: John Doe  
Tahun Terbit: 2020  
Status: Tersedia  
Buku berhasil dipinjam.
```
- OUTPUT**: Shows the output of the Java command, including the classpath and the book information.
- PROBLEMS**: Shows one warning about the use of --enable-preview.
- PORTS**: Shows network port information.
- RUN**: A sidebar on the right lists run configurations for various Java files.

The screenshot shows the Visual Studio Code interface with the following details:

- EXPLORER**: Shows a project named "UAS" containing several Java files: AkunPengguna.java, AutentikasiPengguna.java, Buku.java, DeretFibonacci.java, Faktorisasi.java, HitungTotalHarga.java, KalkulatorSederhana.java, and m.
- CODE**: The current file is PalindromCheck.java, which contains the same code as the first screenshot.
- TERMINAL**: Displays the Windows PowerShell terminal with the following output:

```
Windows PowerShell  
Copyright (C) Microsoft Corporation. All rights reserved.  
  
Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows  
  
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b560b440e7831565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'Buku'  
Informasi buku sebelum dipinjam:  
Judul: Java Programming  
Penulis: John Doe  
Tahun Terbit: 2020  
Status: Tersedia  
Buku berhasil dipinjam.
```
- OUTPUT**: Shows the output of the Java command, including the classpath and the book information.
- PROBLEMS**: Shows one warning about the use of --enable-preview.
- PORTS**: Shows network port information.
- RUN**: A sidebar on the right lists run configurations for various Java files.

8. Akun Pengguna

The screenshot shows the Eclipse IDE interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** UAS
- Left Sidebar:** Explorer (UAS folder containing AkunPengguna.java, AutentikasiPengguna.java, Buku.java, DeretFibonacci.java, Faktorisasi.java, HitungTotalHarga.java, KalkulatorSederhana.java, J m, PalindromCheck.java), Outline, Timeline, VS Code PETS.
- Central Area:** Code editor showing Java code for the `AkunPengguna` class. The code includes:
 - Attributes: `username`, `password`, `aktif`.
 - Constructor: `public AkunPengguna(String username, String password) { this.username = username; this.password = password; this.aktif = false; }`
 - Method `tampilkanInfoAkun()`: Prints `Username: <username>` and `Status: <aktif ? "Aktif" : "Nonaktif">`.
 - Method `aktifkanAkun()`: Prints `"Akun berhasil diaktifkan."` if `aktif` is false, and `"Akun sudah aktif."` otherwise.
 - Method `nonaktifkanAkun()`: Prints `"Metode untuk menonaktifkan akun"`.
- Bottom Status Bar:** In 61 Col 2, Spaces 4, UTER-B, CRLE, Java, etc.



The screenshot shows a Java application named "UAS" in the Explorer sidebar. The "AkunPengguna.java" file is open in the editor, displaying code for activating and deactivating accounts. The code uses a main method to create an account object and then calls methods to print pre-activation information, activate the account, and print post-activation information. It also includes methods to deactivate and print post-deactivation information.

```
public void nonaktifkanAkun() {
    if (aktif) {
        System.out.println("Akun berhasil dinonaktifkan.");
        aktif = false;
    } else {
        System.out.println("Akun sudah nonaktif.");
    }
}

public static void main(String[] args) {
    // Membuat objek akun
    AkunPengguna akun1 = new AkunPengguna(username:"user123", password:"password123");

    // Menampilkan informasi akun sebelum diaktifkan
    System.out.println("Informasi akun sebelum diaktifkan:");
    akun1.tampilkanInfoAkun();

    // Mengaktifkan akun
    akun1.aktifkanAkun();

    // Menampilkan informasi akun setelah diaktifkan
    System.out.println("\nInformasi akun setelah diaktifkan:");
    akun1.tampilkanInfoAkun();

    // Menonaktifkan akun
    akun1.nonaktifkanAkun();

    // Menampilkan informasi akun setelah dinonaktifkan
}
```

A screenshot of the Visual Studio Code interface. The title bar says "UAS". The left sidebar shows a file tree under "EXPLORER" with a folder "UAS" containing several Java files: AkunPengguna.java, AutentikasiPengguna.java, m, AkunPengguna.java (the active file), Buku.java, PalindromCheck.java, DeretFibonacci.java, Faktorasi.java, HitungTotalHarga.java, and KalkulatorSederhana.java. Below the file tree are sections for "OUTLINE" and "TIMELINE". The main editor area displays the following Java code:

```
58     System.out.println("Informasi akun setelah dinonaktifkan:");
59     akun1.tampilkanInfoAkun();
60 }
61 }
62 }
```

Hasil Terminal :

A screenshot of the Visual Studio Code interface. The title bar says "UAS". The left sidebar shows a file tree under "EXPLORER" with a folder "UAS" containing several Java files: AkunPengguna.java (the active file), AutentikasiPengguna.java, m, Buku.java, PalindromCheck.java, DeretFibonacci.java, Faktorasi.java, HitungTotalHarga.java, and KalkulatorSederhana.java. Below the file tree are sections for "OUTLINE" and "TIMELINE". The main editor area displays the following Java code:

```
1 public class AkunPengguna {
2     private String username;
3     private String password;
4     private boolean aktif;
5
6     // Konstruktor untukinisialisasi atribut akun
7     public AkunPengguna(String username, String password) {
8         this.username = username;
9         this.password = password;
10        this.aktif = false; // Saat pertama kali, akun belum aktif
11    }
12
13    // Metode untuk menampilkan informasi akun
14    public void tampilkanInfoAkun() {
15        System.out.println("Username: " + username);
16        System.out.println("Status: " + (aktif ? "Aktif" : "Nonaktif"));
17    }
18}
```

The terminal tab is selected, showing the following command-line output:

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

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

PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09811182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b5e0b440e78311565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'AkunPengguna'
Informasi akun sebelum diaktifkan:
Username: user123
Status: Nonaktif
Akun berhasil diaktifkan.
```

On the right side of the terminal, there is a list of run configurations:

- Run: Hitung...
- Run: Autent...
- Run: DeretF...
- Run: Faktori...
- Run: Kalkula...
- Run: Palindr...
- Run: Buku
- Run: AkunP...

```

File Edit Selection View Go Run ...
File Explorer J AutentikasiPengguna.java J m J AkunPengguna.java 1 X J Buku.java J PalindromCheck.java D v ...
EXPLORER ... jjava J AkunPengguna.java 1 J AutentikasiPengguna.java J m J AkunPengguna.java 1 X J Buku.java J PalindromCheck.java D v ...
J AkunPengguna.java J AutentikasiPengguna.java J Buku.java J PalindromCheck.java
J AutentikasiPengguna.java J Buku.java J PalindromCheck.java
J Buku.java J DeretFibonacci.java J Faktorisasi.java J HitungTotalHarga.java J KalkulatorSederhana.java J m J PalindromCheck.java
J DeretFibonacci.java J Faktorisasi.java J HitungTotalHarga.java J KalkulatorSederhana.java J m J PalindromCheck.java
J Faktorisasi.java J HitungTotalHarga.java J KalkulatorSederhana.java J m J PalindromCheck.java
J HitungTotalHarga.java J KalkulatorSederhana.java J m J PalindromCheck.java
J KalkulatorSederhana.java J m J PalindromCheck.java
J m J PalindromCheck.java
J PalindromCheck.java

1 public class AkunPengguna {
2     private String username;
3     private String password;
4     private boolean aktif;
5
6     // Konstruktor untuk inisialisasi atribut akun
7     public AkunPengguna(String username, String password) {
8         this.username = username;
9         this.password = password;
10        this.aktif = false; // Saat pertama kali, akun belum aktif
11    }
12
13    // Metode untuk menampilkan informasi akun
14    public void tampilanInfoAkun() {
15        System.out.println("Username: " + username);
16        System.out.println("Status: " + (aktif ? "Aktif" : "Nonaktif"));
17    }
18
19    // Metode untuk mengubah status akun
20    public void setAktif(boolean aktif) {
21        this.aktif = aktif;
22    }
23
24    // Metode untuk menampilkan status akun
25    public boolean getAktif() {
26        return aktif;
27    }
28
29 }

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
Username: user123
Status: Nonaktif
Akun berhasil diaktifkan.

Informasi akun setelah diaktifkan:
Username: user123
Status: Aktif
Akun berhasil dinonaktifkan.

Informasi akun setelah dinonaktifkan:
Username: user123
Status: Nonaktif
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS>
In 6. Col 49 Spaces:4 UTF-8 CR/LF {} Java ⌂

```

9. Periksa Urutan Kurung Buka dan Kurung Tutup

```

File Edit Selection View Go Run ...
File Explorer J guna.java 1 J PeriksaUrutanKurungBukaTutup.java ● J Buku.java J PalindromCheck.java J KalkulatorSederhana.java D v ...
EXPLORER ... guna.java 1 J PeriksaUrutanKurungBukaTutup.java ...
J AkunPengguna.java J AutentikasiPengguna.java J Buku.java J PalindromCheck.java
J AutentikasiPengguna.java J Buku.java J PalindromCheck.java
J Buku.java J DeretFibonacci.java J Faktorisasi.java J HitungTotalHarga.java J KalkulatorSederhana.java J m J PalindromCheck.java
J DeretFibonacci.java J Faktorisasi.java J HitungTotalHarga.java J KalkulatorSederhana.java J m J PalindromCheck.java
J Faktorisasi.java J HitungTotalHarga.java J KalkulatorSederhana.java J m J PalindromCheck.java
J HitungTotalHarga.java J KalkulatorSederhana.java J m J PalindromCheck.java
J KalkulatorSederhana.java J m J PalindromCheck.java
J m J PalindromCheck.java
J PalindromCheck.java

1 import java.util.Stack;
2
3 public class PeriksaUrutanKurungBukaTutup {
4
5     Run | Debug
6     public static void main(String[] args) {
7         String ekspresi = "(a + b) * (c - d)";
8
9         // Inisialisasi stack
10        Stack<Character> stack = new Stack<>();
11
12        // Periksa setiap karakter dalam ekspresi
13        for (int i = 0; i < ekspresi.length(); i++) {
14            char karakter = ekspresi.charAt(i);
15
16            // Jika karakter adalah kurung buka, tambahkan ke stack
17            if (karakter == '(') {
18                stack.push(karakter);
19            }
20
21            // Jika karakter adalah kurung tutup, keluarkan dari stack
22            else if (karakter == ')') {
23                if (stack.empty() || stack.peek() != '(') {
24                    System.out.println("Urutan kurung tidak benar");
25                    return;
26                }
27
28                stack.pop();
29            }
30
31        }
32
33        if (!stack.isEmpty()) {
34            System.out.println("Urutan kurung tidak benar");
35        }
36    }
37
38 }

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
In 30. Col 1 Spaces:4 UTF-8 CR/LF {} Java ⌂

```

The screenshot shows the Visual Studio Code interface with a dark theme. The Explorer sidebar on the left lists several Java files under a 'UAS' folder. The 'PERIKSAURUTANKURUNGBUKATUTUP.java' file is open in the main editor area. The code checks if a given string has correctly nested opening and closing parentheses. If the stack is empty at the end, it prints that the sequence is incorrect; otherwise, it prints that it is correct.

```
29     }
30
31     // Jika stack masih berisi kurung, maka urutan kurung tidak benar
32     if (!stack.empty()) {
33         System.out.println(x:"Urutan kurung tidak benar");
34     } else {
35         System.out.println(x:"Urutan kurung benar");
36     }
37 }
38
39 }
```

Hasil Terminal :

The screenshot shows the Visual Studio Code interface with a dark theme, focusing on the terminal tab. The terminal window displays the command 'java -jar UAS.jar' being run, followed by the output of the program which correctly identifies the input as having a balanced set of parentheses.

```
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '--enable-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\USER\AppData\Roaming\Code\User\workspaceStorage\f5809d48b560b440e78311565c352c9b\redhat.java\jdt_ws\UAS_bb14e730\bin' 'PeriksaUrutanKurungBukatutup'
Urutan kurung benar
PS C:\Users\USER\Documents\PEMROGRAMAN KOMPUTER\09011182328015\UAS>
```