1. public class DataPenjualan {

public static void main(String[] args) {

        // Nilai harga barang dan jumlah pembelian (contoh)

        double hargaBarang = 100.0;

        int jumlahPembelian = 15;

        // Hitung total harga setelah diskon

        double totalHarga = hargaBarang \* jumlahPembelian \* (1 -    hitungDiskon(jumlahPembelian));

        // Tampilkan total harga setelah diskon

        System.out.println("Total harga setelah diskon: " +         totalHarga);

    }

    // Metode untuk menghitung diskon berdasarkan jumlah pembelian

    static double hitungDiskon(int jumlahPembelian) {

        if (jumlahPembelian < 5) {

            return 0.0; // Tidak ada diskon

        } else if (jumlahPembelian <= 10) {

            return 0.05; // Diskon 5%

        } else if (jumlahPembelian <= 20) {

            return 0.1; // Diskon 10%

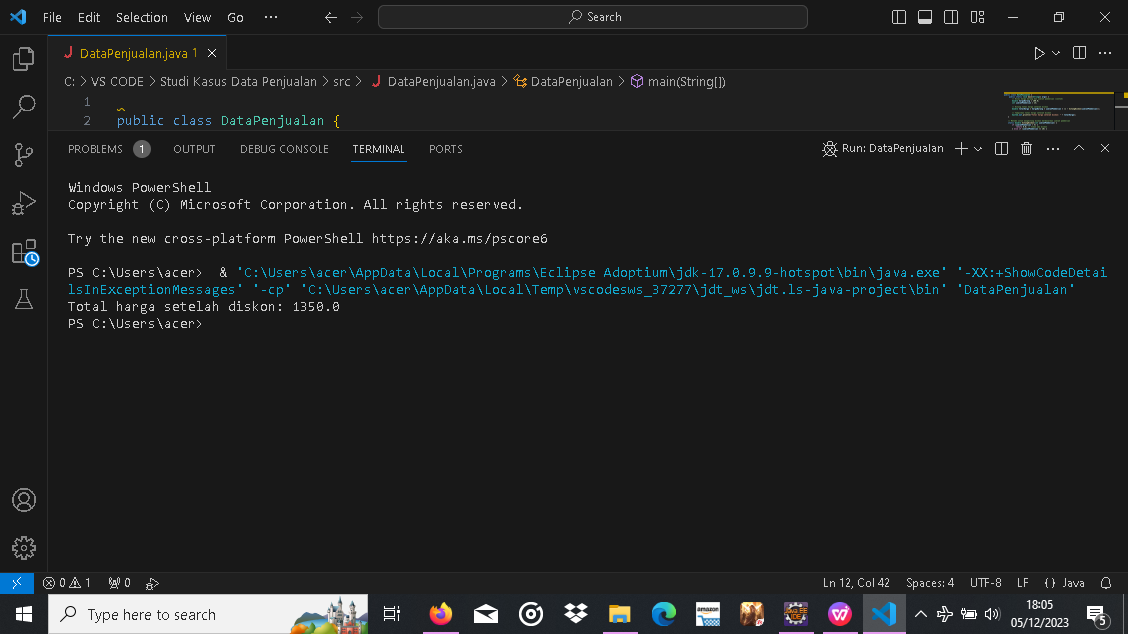
        } else {

            return 0.2; // Diskon 20%

        }

    }

}



2.

public class DataPenjualan {

public static void main(String[] args) {

        // username dan password asli

        String usernameAsli = "user124";

        String passwordAsli = "pass123";

        // username dan password  yang dimasukkan

        String usernameInput = "user123";

        String passwordInput = "pass123";

        // Lakukan autentikasi

        if (autentikasi(usernameInput, passwordInput, usernameAsli, passwordAsli)) {

            System.out.println("Autentikasi Berhasil");

        } else {

            System.out.println("Autentikasi Gagal");

        }

    }

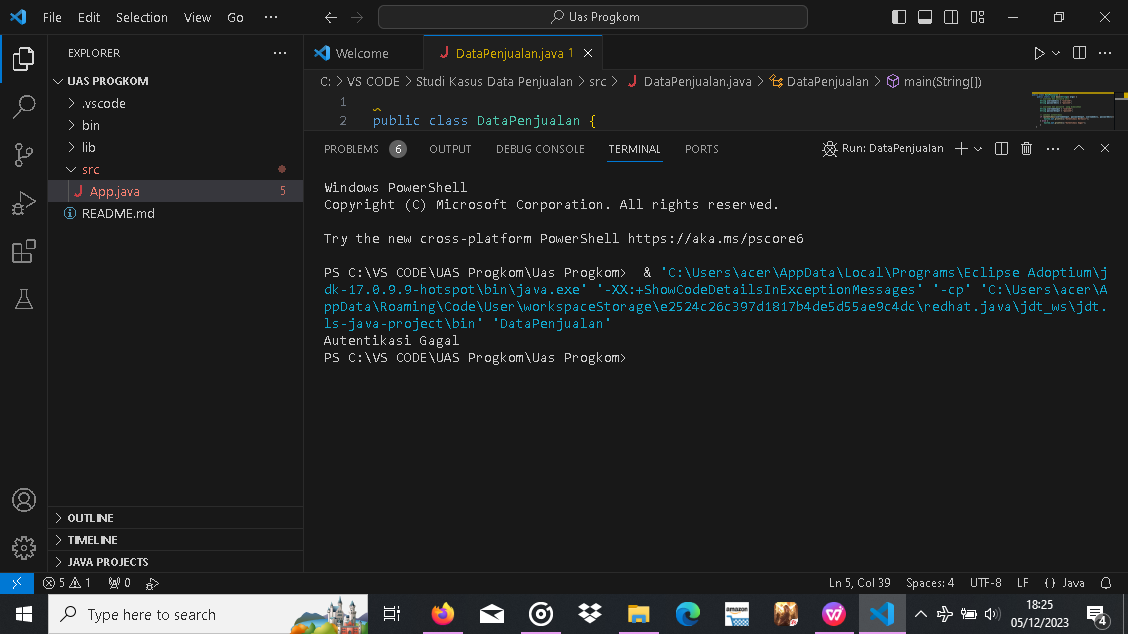
    // Metode untuk melakukan autentikasi

        static boolean autentikasi(String usernameInput, String passwordInput, String usernameValid, String passwordValid) {

        return usernameInput.equals(usernameValid) && passwordInput.equals(passwordValid);

    }

}



3.

public class DataPenjualan {

public static void main(String[] args) {

      int n = 12;

        // Tampilkan deret Fibonacci hingga suku ke-n

        System.out.println("Deret Fibonacci hingga suku ke-" + n + ":");

        for (int i = 1; i <= n; i++) {

            System.out.print(fibonacci(i) + " ");

        }

    }

    // Metode untuk menghitung suku ke-n dalam deret Fibonacci

    static int fibonacci(int n) {

        if (n <= 1) {

            return n;

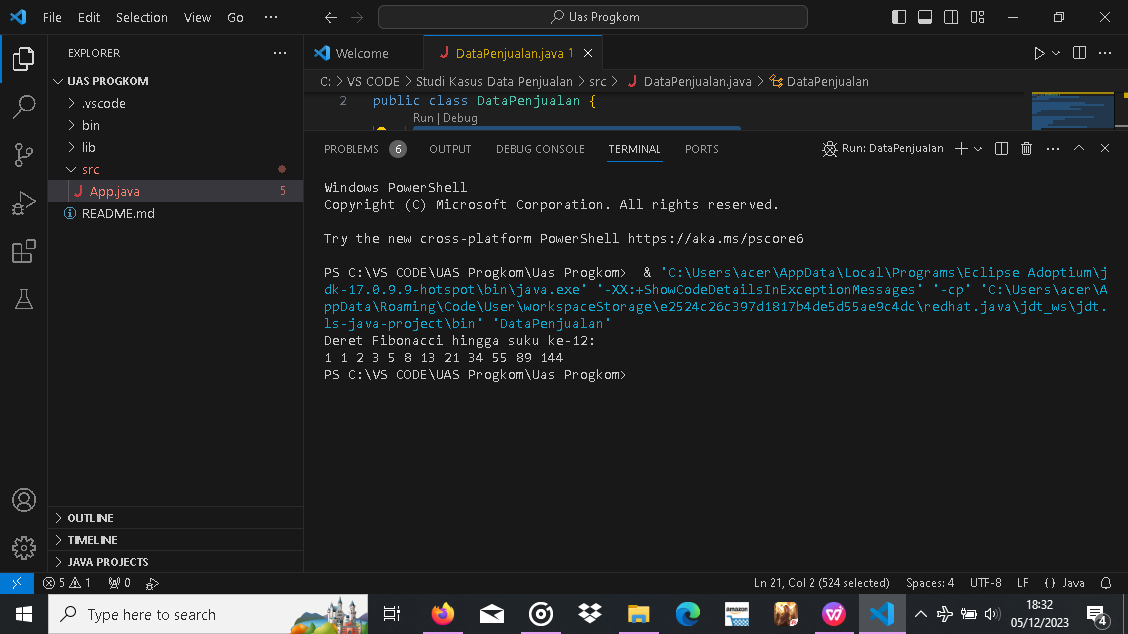
        } else {

            return fibonacci(n - 1) + fibonacci(n - 2);

        }

    }

}



4.

public class DataPenjualan {

public static void main(String[] args) {

        int bilangan = 12; // Ganti dengan bilangan yang diinginkan

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

        faktorisasi(bilangan);

    }

    // Metode untuk memfaktorisasi suatu bilangan

    static void faktorisasi(int bilangan) {

        for (int i = 2; i <= bilangan; i++) {

            while (bilangan % i == 0) {

                System.out.print(i);

                bilangan /= i;

                if (bilangan > 1) {

                    System.out.print(" \* ");

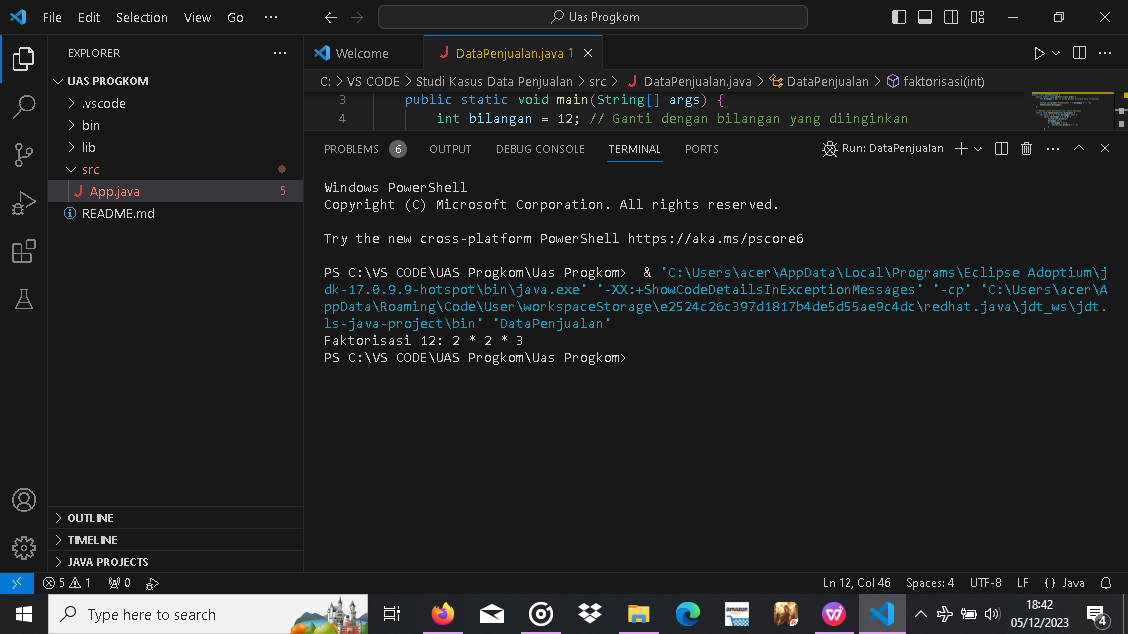
                }

            }

        }

    }

}



5.

public static void main(String[] args) {

        // Masukkan angka-angka yang ingin dioperasikan

        double angka1 = 10.0;

        double angka2 = 5.0;

        // Operasi penjumlahan

        penjumlahan(angka1, angka2);

        // Operasi pengurangan

        pengurangan(angka1, angka2);

        // Operasi perkalian

        perkalian(angka1, angka2);

        // Operasi pembagian

        pembagian(angka1, angka2);

    }

    // Metode untuk operasi penjumlahan

    static void penjumlahan(double a, double b) {

        double hasil = a + b;

        System.out.println("Hasil Penjumlahan: " + hasil);

    }

    // Metode untuk operasi pengurangan

    static void pengurangan(double a, double b) {

        double hasil = a - b;

        System.out.println("Hasil Pengurangan: " + hasil);

    }

    // Metode untuk operasi perkalian

    static void perkalian(double a, double b) {

        double hasil = a \* b;

        System.out.println("Hasil Perkalian: " + hasil);

    }

    // Metode untuk operasi pembagian

    static void pembagian(double a, double b) {

        if (b != 0) {

            double hasil = a / b;

            System.out.println("Hasil Pembagian: " + hasil);

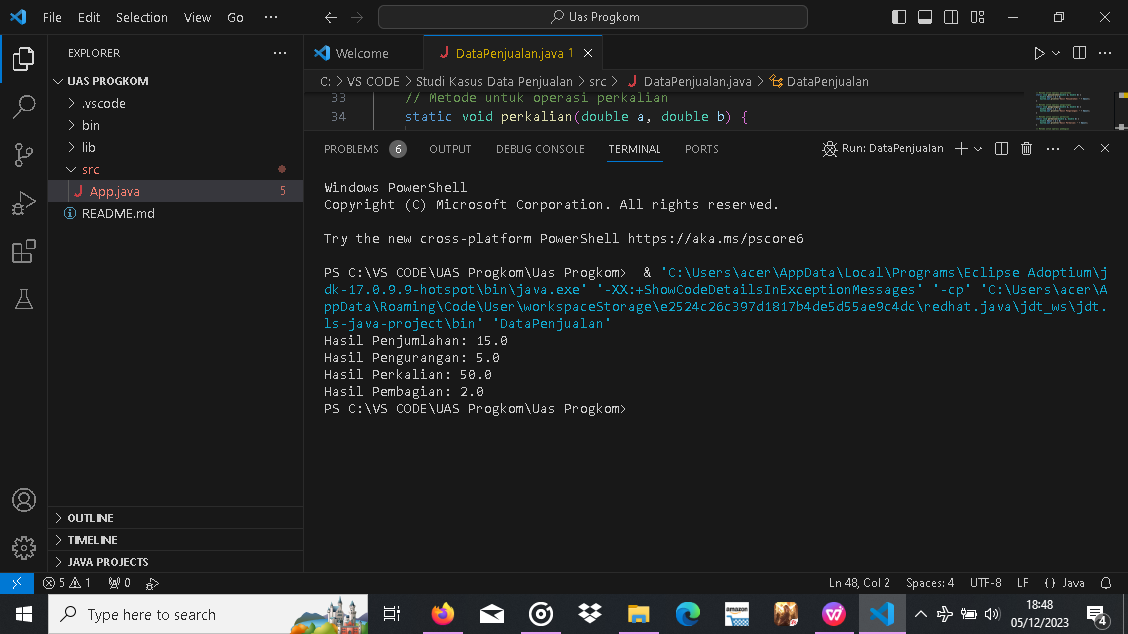
        } else {

            System.out.println("Pembagian oleh nol tidak diperbolehkan.");

        }

    }

}



6

public class DataPenjualan {

  public static void main(String[] args) {

        String input = "Coding"; // Ganti dengan kata atau frase yang ingin diuji

        if (isPalindrom(input)) {

            System.out.println("Kata atau frase tersebut adalah palindrom.");

        } else {

            System.out.println("Kata atau frase tersebut bukan palindrom.");

        }

    }

    // Metode untuk mengecek apakah suatu kata atau frase adalah palindrom

    static boolean isPalindrom(String str) {

        // Menghapus spasi dan mengubah ke huruf kecil untuk pengecekan yang lebih baik

        String cleanedStr = str.replaceAll("\\s", "").toLowerCase();

        // Mengecek palindrom

        int left = 0;

        int right = cleanedStr.length() - 1;

        while (left < right) {

            if (cleanedStr.charAt(left) != cleanedStr.charAt(right)) {

                return false;

            }

            left++;

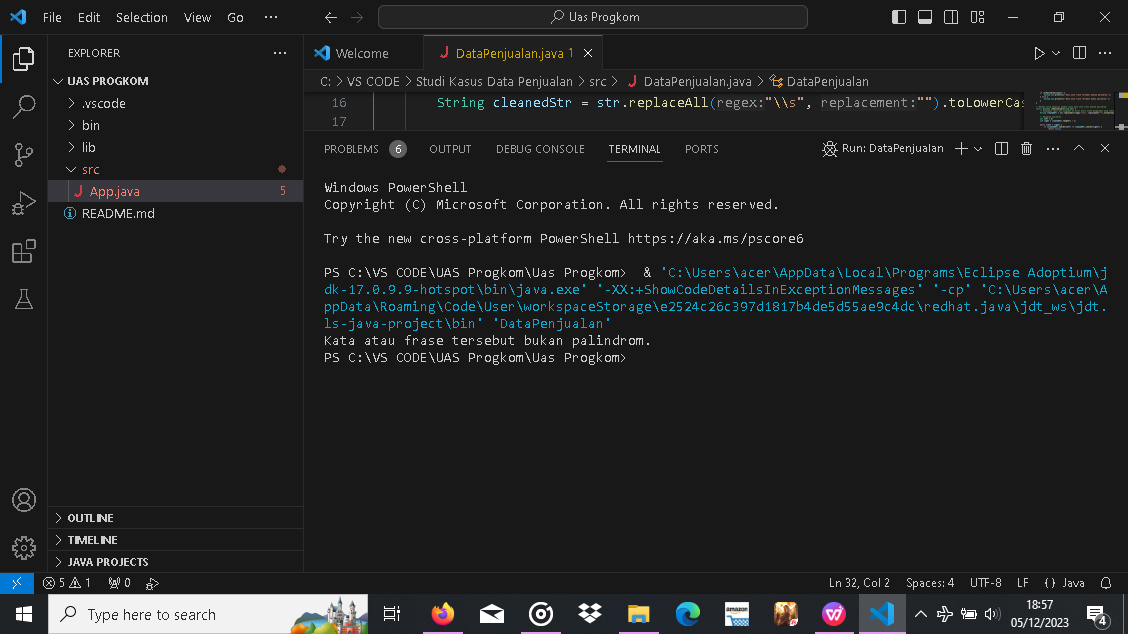
            right--;

        }

        return true;

    }

}



7.

import java.util.Scanner;

public class DataPenjualan {

        public static void tampilkanInformasiBuku(String[] judulBuku, String[] penulis, int[] tahunTerbit, String[] status) {

        System.out.println("Daftar Buku:");

        for (int i = 0; i < judulBuku.length; i++) {

            System.out.println("Judul: " + judulBuku[i]);

            System.out.println("Penulis: " + penulis[i]);

            System.out.println("Tahun Terbit: " + tahunTerbit[i]);

            System.out.println("Status: " + status[i]);

            System.out.println();

        }

    }

    // Metode untuk meminjam buku

    public static void pinjamBuku(String[] judulBuku, String[] status, String judulDicari) {

        for (int i = 0; i < judulBuku.length; i++) {

            if (judulBuku[i].equalsIgnoreCase(judulDicari)) {

                if (status[i].equalsIgnoreCase("Tersedia")) {

                    status[i] = "Dipinjam";

                    System.out.println("Buku " + judulDicari + " berhasil dipinjam.");

                } else {

                    System.out.println("Maaf, buku " + judulDicari + " sedang dipinjam.");

                }

                return;

            }

        }

        System.out.println("Buku " + judulDicari + " tidak ditemukan dalam daftar.");

    }

    public static void main(String[] args) {

        // List Buku dengan Author, Tahun, dan Statusnya

        String[] judulBuku = {"Java Programming", "Data Structures and Algorithms", "Introduction to AI", "Database Management", "Web Development"};

        String[] penulis = {"John Smith", "Alice Johnson", "Robert Williams", "Emily Davis", "Michael Brown"};

        int[] tahunTerbit = {2020, 2019, 2022, 2018, 2021};

        String[] status = {"Tersedia", "Tidak Tersedia", "Tersedia", "Tersedia", "Tersedia"};

        // Menampilkan List Buku

        tampilkanInformasiBuku(judulBuku, penulis, tahunTerbit, status);

        // Melakukan pencarian judul buku

        Scanner scanner = new Scanner(System.in);

        System.out.print("Masukkan judul buku yang ingin dipinjam: ");

        String judulDicari = scanner.nextLine();

        // Meminjam buku

        pinjamBuku(judulBuku, status, judulDicari);

        // Menampilkan informasi buku setelah peminjaman

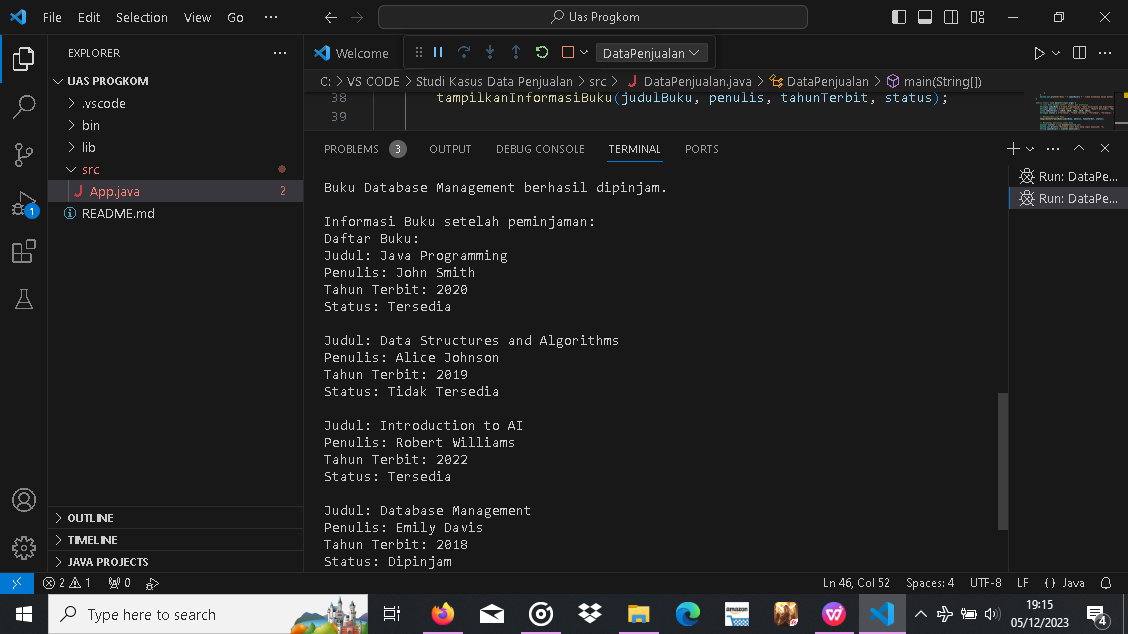
        System.out.println("\nInformasi Buku setelah peminjaman:");

        tampilkanInformasiBuku(judulBuku, penulis, tahunTerbit, status);

        scanner.close();

    }

}



8. public class DataPenjualan {

    private static String username = "user123";

    private static String password = "password123";

    private static boolean aktif = false;

    public static void main(String[] args) {

        // Tampilkan informasi akun sebelum perubahan status

        System.out.println("Informasi Akun:\n" + tampilkanInformasi());

        // Aktifkan akun

        System.out.println("\n" + aktifkanAkun());

        // Tampilkan informasi setelah perubahan status

        System.out.println("\nInformasi Akun setelah diaktifkan:\n" + tampilkanInformasi());

        // Nonaktifkan akun

        System.out.println("\n" + nonaktifkanAkun());

        // Tampilkan informasi setelah perubahan status

        System.out.println("\nInformasi Akun setelah dinonaktifkan:\n" + tampilkanInformasi());

    }

    // Metode untuk menampilkan informasi akun

    private static String tampilkanInformasi() {

        return "Username: " + username + "\nPassword: " + password + "\nStatus Aktif: " + (aktif ? "Aktif" : "Nonaktif");

    }

    // Metode untuk mengaktifkan akun

    private static String aktifkanAkun() {

        if (!aktif) {

            aktif = true;

            return "Akun berhasil diaktifkan.";

        } else {

            return "Akun sudah aktif.";

        }

    }

    // Metode untuk menonaktifkan akun

    private static String nonaktifkanAkun() {

        if (aktif) {

            aktif = false;

            return "Akun berhasil dinonaktifkan.";

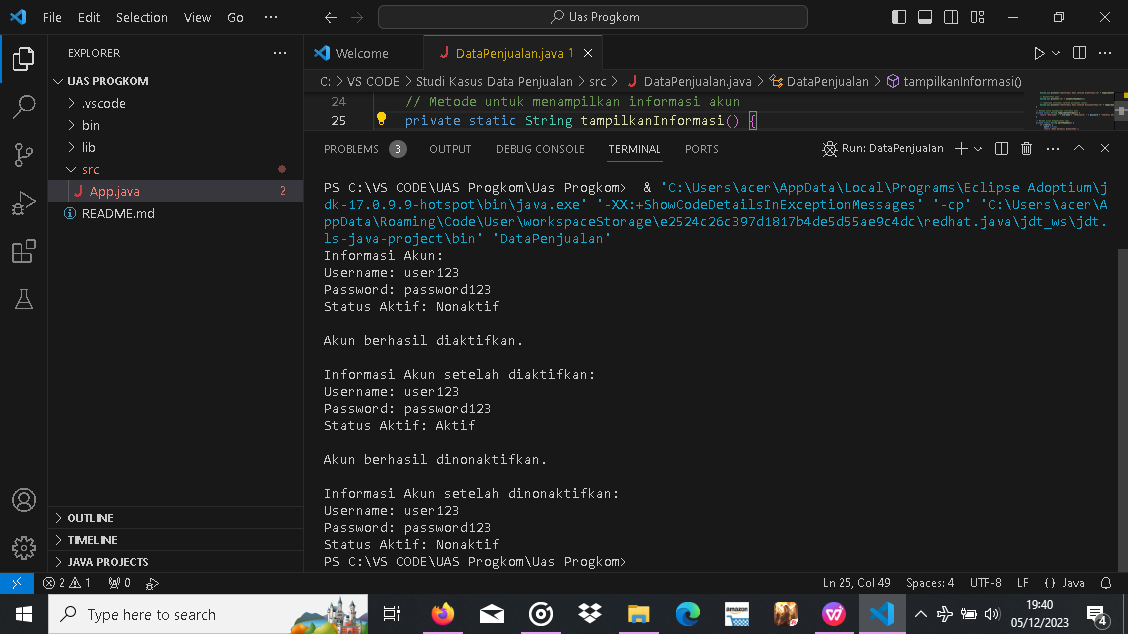
        } else {

            return "Akun sudah nonaktif.";

        }

    }

}



9.

import java.util.Stack;

public class DataPenjualan {

    public static boolean cekUrutanKurung(String ekspresi) {

        Stack<Character> stack = new Stack<>();

        for (int i = 0; i < ekspresi.length(); i++) {

            char karakter = ekspresi.charAt(i);

            if (karakter == '(' || karakter == '[' || karakter == '{') {

                stack.push(karakter);

            } else if (karakter == ')' || karakter == ']' || karakter == '}') {

                if (stack.isEmpty()) {

                    return false; // Tidak ada kurung buka yang sesuai

                }

                char kurungBuka = stack.pop();

                // Periksa apakah kurung tutup sesuai dengan kurung buka yang ada di stack

                if ((karakter == ')' && kurungBuka != '(') ||

                    (karakter == ']' && kurungBuka != '[') ||

                    (karakter == '}' && kurungBuka != '{')) {

                    return false;

                }

            }

        }

        // Pastikan tidak ada kurung buka yang belum tertutup

        return stack.isEmpty();

    }

    public static void main(String[] args) {

        String ekspresi1 = "((a + b) \* (c - d))";

        String ekspresi2 = "{[a + b] \* (c - d)}";

        String ekspresi3 = "((a + b) \* (c - d)";

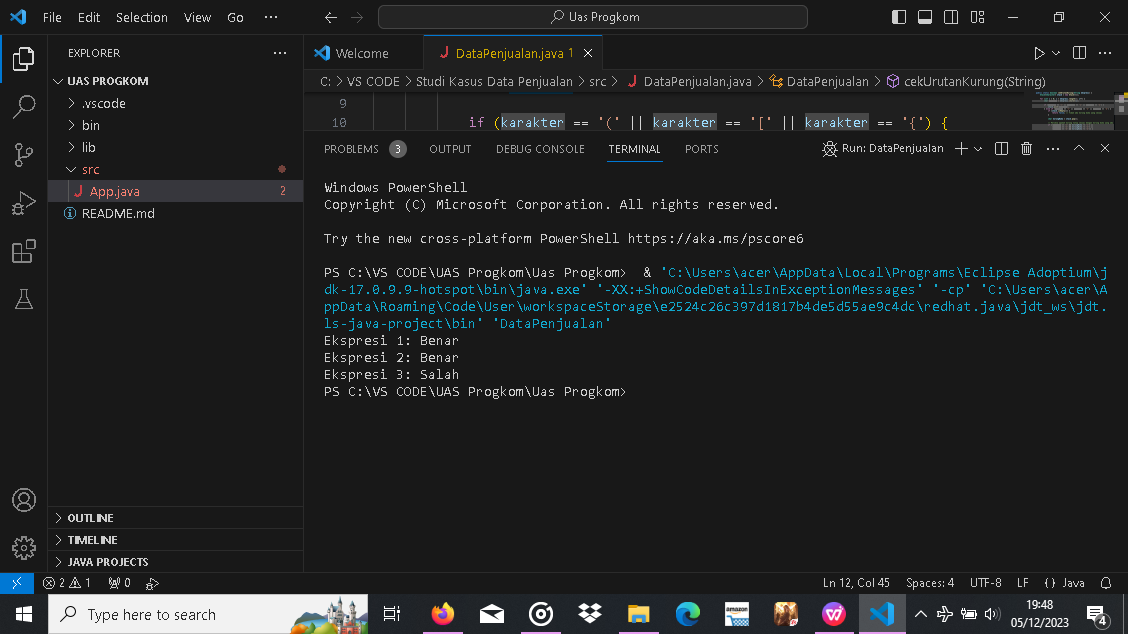
        System.out.println("Ekspresi 1: " + (cekUrutanKurung(ekspresi1) ? "Benar" : "Salah"));

        System.out.println("Ekspresi 2: " + (cekUrutanKurung(ekspresi2) ? "Benar" : "Salah"));

        System.out.println("Ekspresi 3: " + (cekUrutanKurung(ekspresi3) ? "Benar" : "Salah"));

    }

}



10.