**LAPORAN HASIL PRAKTIKUM**

**JOBSHEET 12**

**POLINEMA**

**MOCHAMMAD RIJAL DZAKI RIFKI AFIFUDIN**

**244107020240**

**1H**

**PRODI D-IV TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

Percobaan 1

Kode program:

public class percobaan1 {

    static int faktorialRekursif(int n){

        if (n==0) {

            return (1);

        }else {

            return (n\*faktorialRekursif(n-1));

        }

    }

    static int faktorialIteratif(int n){

        int faktor = 1;

        for (int i = n; i >= 1; i--) {

            faktor = faktor \*i;

        }

        return faktor;

    }

    public static void main(String[] args) {

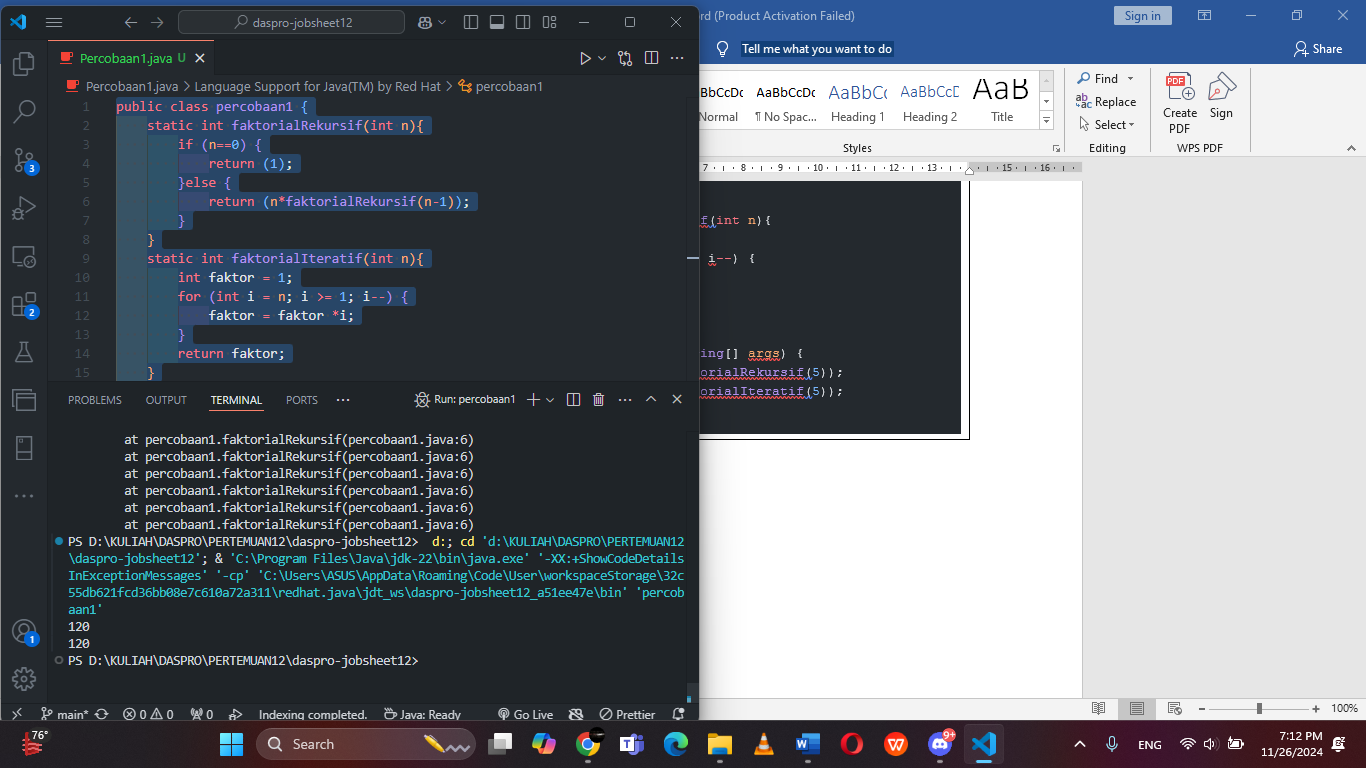
        System.out.println(faktorialRekursif(5));

        System.out.println(faktorialIteratif(5));

    }

}

Hasil :



Jawaban pertanyaan percobaan 1

1. Rekursif adalah suatu konsep yang dapat memanggil fungsi itu sendiri. Tujuan memanggil dirinya sendiri adalah untuk menghasilkan proses perulangan dengan pola tertentu dan batasan tertentu.
2. Penggunaan rekursi sendiri bisa di terapkan pada kasus pencarian data berskala besar, sebagai contohnya menampilkan data mahasiswa kelas TI-1H berdasarkan nomor absen.
3. Hasilnya sama yaitu 120, Alur interatif :

* Inisialisasi variabel hasil dengan 1
* Melakukan perulangan dari 1 hingga n
* Mengalikan hasil dengan nilai i saat ini
* Setelah loop selesai, mengembalikan hasil akhir
* Tidak ada pemanggilan fungsi di dalam fungsi itu sendiri

Alur Rekursif :

 Memeriksa base case terlebih dahulu

 Setiap pemanggilan disimpan

 Proses berlanjut hingga mencapai base case

 Kemudian stack akan mulai diurai dengan mengembalikan hasil

Percobaan 2

Kode program :

import java.util.Scanner;

public class percobaan2 {

    static int hitungPangkat(int x, int y){

        if (y==0){

            return (1);

        }else {

            return (x\*hitungPangkat(x, y-1));

        }

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Bilangan yang dihitung:");

        int bilangan = sc.nextInt();

        System.out.print("Pangkat:");

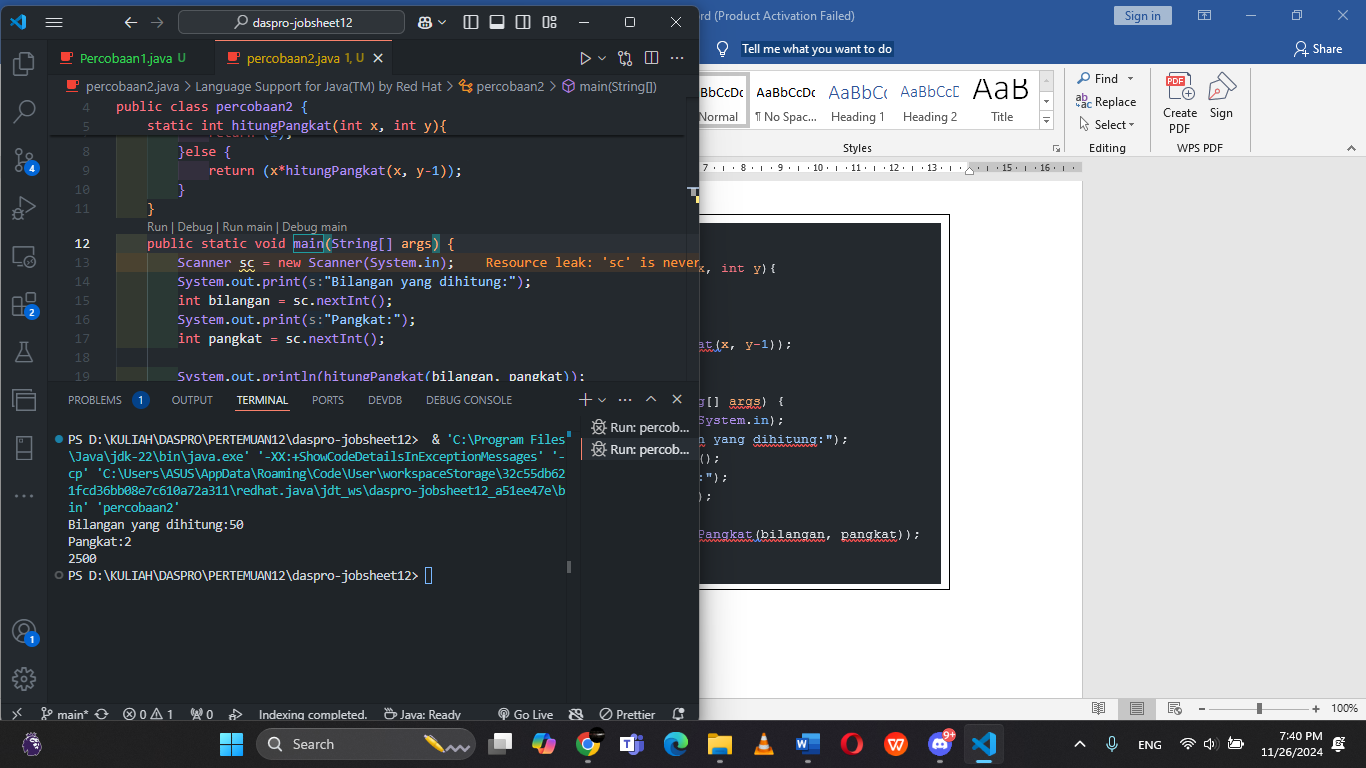
        int pangkat = sc.nextInt();

        System.out.println(hitungPangkat(bilangan, pangkat));

    }

}

Hasil :



Jawaban pertanyaan percobaan 2

1. Sampai y==0 (pangkat mencapai 0)
2. Modifikasi program :

Hasil :

Percobaan 3

Kode program :

import java.util.Scanner;

public class percobaan3 {

    static double hitungLaba(double saldo, int tahun){

        if (tahun == 0){

            return (saldo);

        }else {

            return (1.11 \* hitungLaba(saldo, tahun-1));

        }

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Jumlah saldo awal :");

        int saldoAwal = sc.nextInt();

        System.out.print("Lamanya investasi (tahun) :");

        int tahun = sc.nextInt();

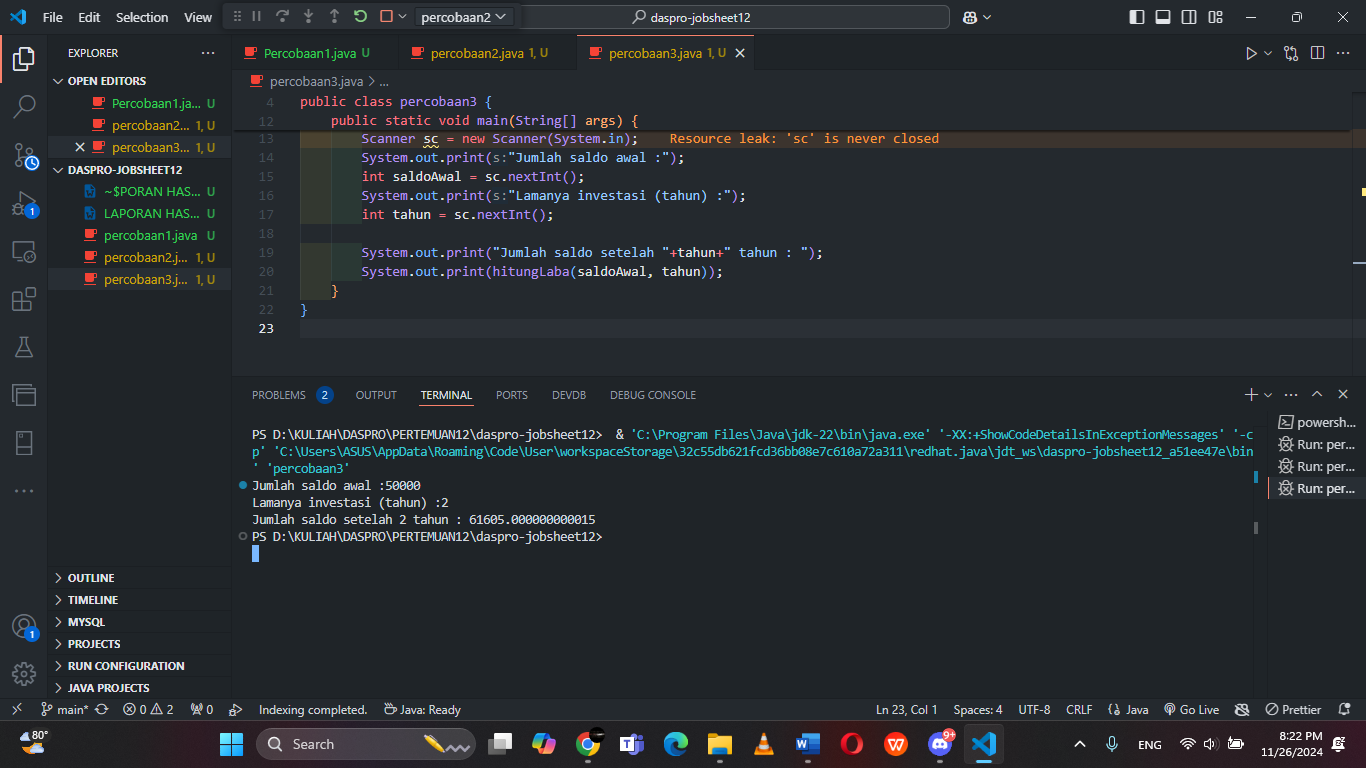
        System.out.print("Jumlah saldo setelah "+tahun+" tahun : ");

        System.out.print(hitungLaba(saldoAwal, tahun));

    }

}

Hasil :



Jawaban pertanyaan percobaan 3

1. Base code :

if (tahun == 0){

            return (saldo);

        }

Recursion call :

else {

            return (1.11 \* hitungLaba(saldo, tahun-1));

        }

1. Fase ekspansi :

Panggilann 1 : hitungLaba(100000,3) -> Tahun !=0 maka memanggil 1.11 \* hitungLaba(100000, 2).

Panggilann 2 : hitungLaba(100000,2) -> Tahun !=0 maka memanggil 1.11 \* hitungLaba(100000, 1).

Panggilann 3 : hitungLaba(100000,1) -> Tahun !=0 maka memanggil 1.11 \* hitungLaba(100000, 0).

Panggilann 4 : hitungLaba(100000,0) -> Tahun ==0 maka mengembalikan saldo awal =100000

Fase subsitusi :

hitungLaba(100000,0) -> mengembalikan saldo awal = 100000

hitungLaba(100000,1) -> memanggil 1.11 \* hitungLaba(100000, 0) = 111000

hitungLaba(100000,2) -> memanggil 1.11 \* hitungLaba(100000, 1) = 123210

hitungLaba(100000,3) -> memanggil 1.11 \* hitungLaba(100000, 2) = 136643.10

Jawaban tugas jobsheet 13

1. Kode program :

import java.util.Scanner;

public class tugas1 {

    static int rekursif(int n){

        if (n<1) {

            return (0);

        }else {

            System.out.print(n+" ");

        }

        return (rekursif(n-1));

    }

    static int Iteratif(int n){

        for (int i = n; i >= 1; i--) {

            System.out.print(i+" ");

        }

        return 0;

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan Angka: ");

        int angka = sc.nextInt();

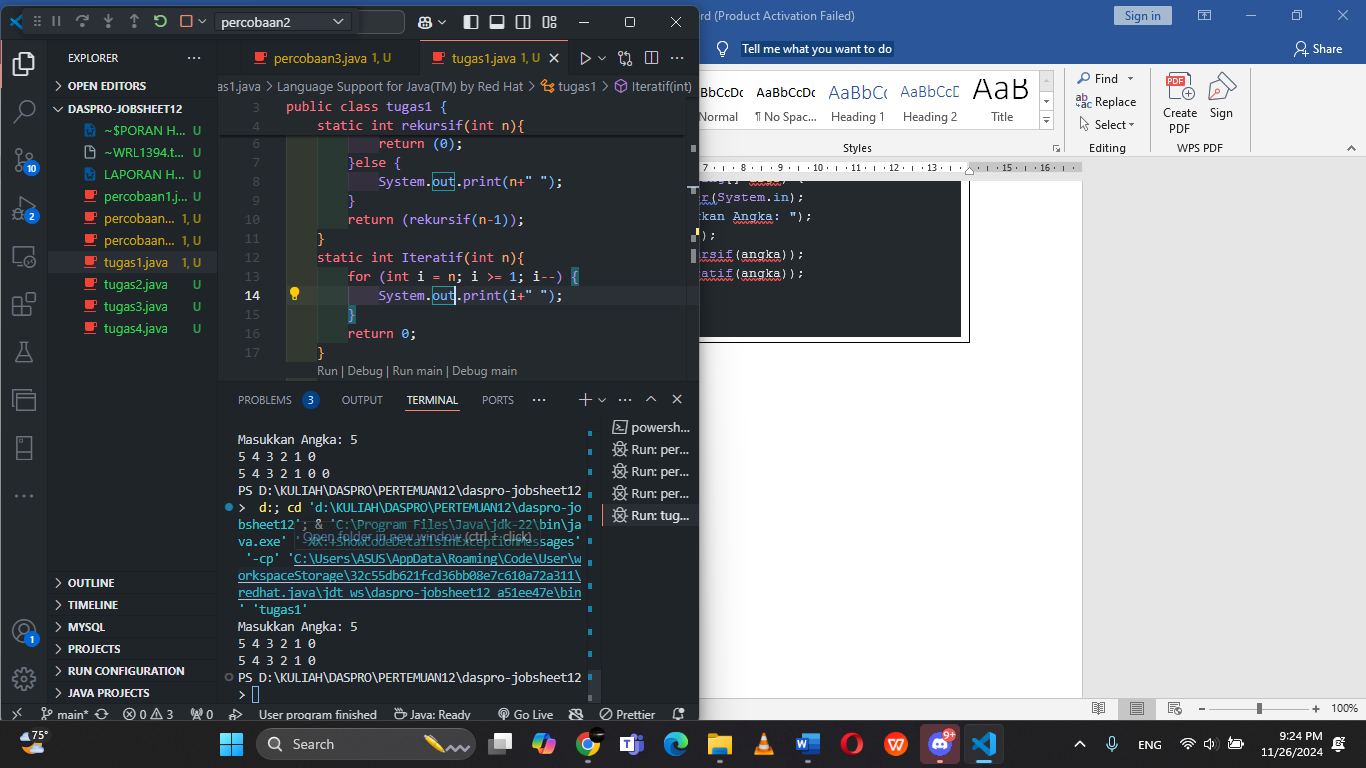
        System.out.println(rekursif(angka));

        System.out.println(Iteratif(angka));

    }

}

Hasil :



1. Kode program :

import java.util.Scanner;

public class tugas2 {

    static int rekursif(int n){

        if (n==1) {

            System.out.print("1");

            return 1;

        }

        System.out.print(n+" + ");

        return (n+rekursif(n-1));

    }

    static int urut(int nomer, int max) {

        if (nomer > max) {

            return 0;

        }

        if (nomer == max) {

            System.out.print(nomer);

            return nomer;

        }

        System.out.print(nomer + "+");

        return nomer + urut(nomer + 1, max);

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan Angka: ");

        int angka = sc.nextInt();

        int total = rekursif(angka);

        System.out.println(" = "+total);

    }

}

Hasil :

