**LAPORAN PRAKTIKUM**

**PRAKTIKUM DASAR PEMROGRAMAN**

**JOBSHEET 6**

**PEMILIHAN 2**

**Oleh:**

**Yefta Octavianus Santo            2341720110**

****

**PROGRAM STUDI D-IV TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG**

**OKTOBER 2023**

**2.1 Percobaan 1**

Menentukan tahun kabisat atau bukan, dengan ketentuan tahun kabisat adalah tahun kelipatan 4 dan bukan kelipatan 100.

1. Bagaimana outputnya ketika diberikan input tahun 2100 ? Jelaskan! Bagaimana agar output sesuai dengan ketentuan (Tahun 2100 bukan tahun kabisat)

* Ketika diberi input tahun 2100 tidak muncul apa-apa, agar muncul output maka kode dimodifikasi sedikit sebagai berikut:

1. Tahun 2000 adalah kelipatan 4 dan kelipatan 100, tetapi tahun 2000 merupakan tahun kabisat, ketentuan tambahan (pengecualian) adalah ketika tahun kelipatan 100 **dan juga** kelipatan 400 maka tahun tersebut merupakan tahun kabisat. Modifikasi program untuk menyesuaikan ketentuan tersebut ! (selesaikan tanpa menggunakan operator logika)
2. import java.util.Scanner;
3. public class Pemilihan2Percobaan129 {
4. public static void main(String[] args){
5. Scanner input29 = new Scanner(System.in);
6. int tahun;
7. System.out.print("Masukkan Tahun : ");
8. tahun = input29.nextInt();
9. if(tahun % 4 == 0){
10. if(tahun % 100 == 0){
11. if(tahun % 400 == 0){
12. System.out.println("Tahun " + tahun + " adalah tahun kabisat");
13. }else{
14. System.out.println("Tahun " + tahun + " bukan tahun kabisat");
15. }
16. }else{
17. System.out.println("Tahun " + tahun + " adalah tahun kabisat");
18. }
19. } else {
20. System.out.println("Tahun " + tahun + " bukan tahun kabisat");
21. }
22. }
23. }

**2.2 Percobaan 2**

1. Jenis bangun datar segitiga selain segitiga siku-siku terdapat segitiga sama sisi, segitiga sama kaki dan segitiga sembarang. Modifikasi program tersebut untuk dapat mengeluarkan output jenis segitiga yang lain (selain segitiga siku-siku). Silahkan menggunakan operator logika untuk menghubungkan multi kondisi.

import java.util.Scanner;

public class Pemilihan2Percobaan229 {

    public static void main(String[] args) {

        System.out.println("Menentukan jenis bidang datar segitiga dari masukan ketiga sudutnya");

        Scanner input29 = new Scanner(System.in);

        float sudut1, sudut2, sudut3, totalSudut;

        System.out.print("Masukkan sudut 1 : ");

        sudut1 = input29.nextFloat();

        System.out.print("Masukkan sudut 2 : ");

        sudut2 = input29.nextFloat();

        System.out.print("Masukkan sudut 3 : ");

        sudut3 = input29.nextFloat();

        totalSudut = (sudut1 + sudut2 + sudut3);

        System.out.println("Total sudut : " + totalSudut);

        if (totalSudut == 180 && sudut1 > 0 && sudut2 > 0 && sudut3 > 0) {

            if (sudut1==90||sudut2==90||sudut3==90) {

                System.out.println("Segitiga siku-siku");

            } //segitiga sama sisi

            else if (sudut1 == sudut2 && sudut2 == sudut3) {

                System.out.println("Segitiga sama sisi");

            }

            //segitiga sama kaki

            else if (sudut1 == sudut2 || sudut2 == sudut3 || sudut1 == sudut3) {

                System.out.println("Segitiga sama kaki");

            }

            else {

                System.out.println("Segitiga sembarang / Bukan segitiga siku-siku");

            }

        } else {

            System.out.println("Bukan segitiga");

        }

    }

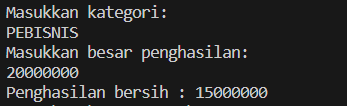
}

* 1. **Percobaan 3**

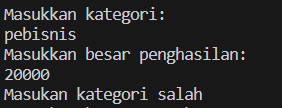
1. import java.util.Scanner;
2. public class Pemilihan2Percobaan329 {
4. public static void main(String[] args) {
5. Scanner input29 = new Scanner(System.in);
6. String category;
7. int inCome, netSalary;
8. double tax;
9. System.out.print("Enter a category : ");
10. category = input29.nextLine();
11. System.out.print("Enter inCome : ");
12. inCome = input29.nextInt();
14. if (category.equalsIgnoreCase("worker")) {
15. if (inCome <= 2000000) {
16. tax = 0.1;
17. } else if (inCome <= 3000000) {
18. tax = 0.15;
19. } else {
20. tax = 0.2;
21. }
22. netSalary = (int) (inCome - (inCome \* tax));
23. System.out.println("The net salary : " + netSalary);
24. } else if (category.equalsIgnoreCase("businessman")) {
25. if (inCome <= 250000000) {
26. tax = 0.15;
27. } else if (inCome <= 35000000) {
28. tax = 0.2;
29. } else {
30. tax = 0.25;
31. }
32. netSalary = (int) (inCome - (inCome \* tax));
33. System.out.println("The net salary : " + netSalary);
34. } else {
35. System.out.println("category pekerjaan tidak valid");;
36. }
37. }
38. }
39. Jelaskan fungsi dari (**int**) pada sintaks gajiBersih = (int) (penghasilan - (penghasilan \* pajak));

fungsi dari (int) pada sintaks tersebut adalah untuk proses casting, yaitu mengubah tipe data hasil perhitungan menjadi tipe data integer (bilangan bulat)

1. Jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Apa kegunaan dari **equalsIgnoreCase**?

* Yang terjadi adalah sebagai berikut : . Tidak berpengaruh apa-apa, karena **equalsIgnoreCase** digunakan untuk membandingkan dua string tanpa memperhatikan perbedaan huruf besar dan kecil (case-insensitive).

1. Ubah **equalsIgnoreCase** menjadi **equals**, kemudian jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Mengapa hasilnya demikian? Apa kegunaan dari **equals**?

* Yang terjadi adalah output masukan kategori salah, .

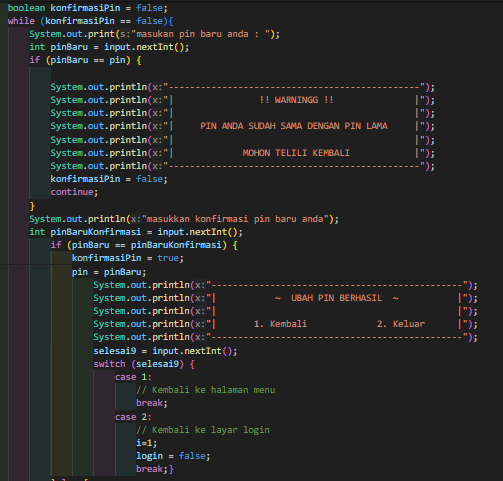
Masukan kategori salah, karena **equals** digunakan untuk membandingkan dua string secara tepat dan memerlukan bahwa kedua string harus identik dalam hal huruf besar-kecil (case-sensitive).

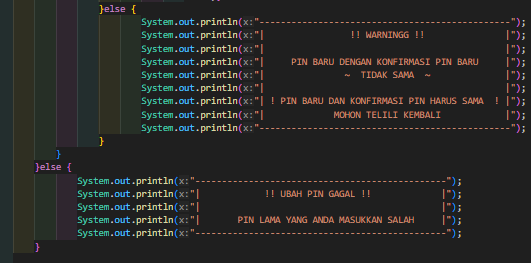
**Tugas**

Buatlah kode program berdasarkan flowchart yang telah dibuat pada Tugas pertemuan 6 Matakuliah Dasar Pemrograman terkait Project !

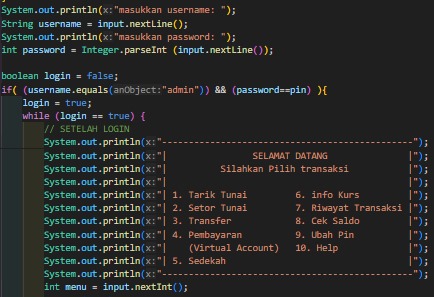
Push dan commit hasil kode program anda ke repository project Anda!

* **Ubah Pin**

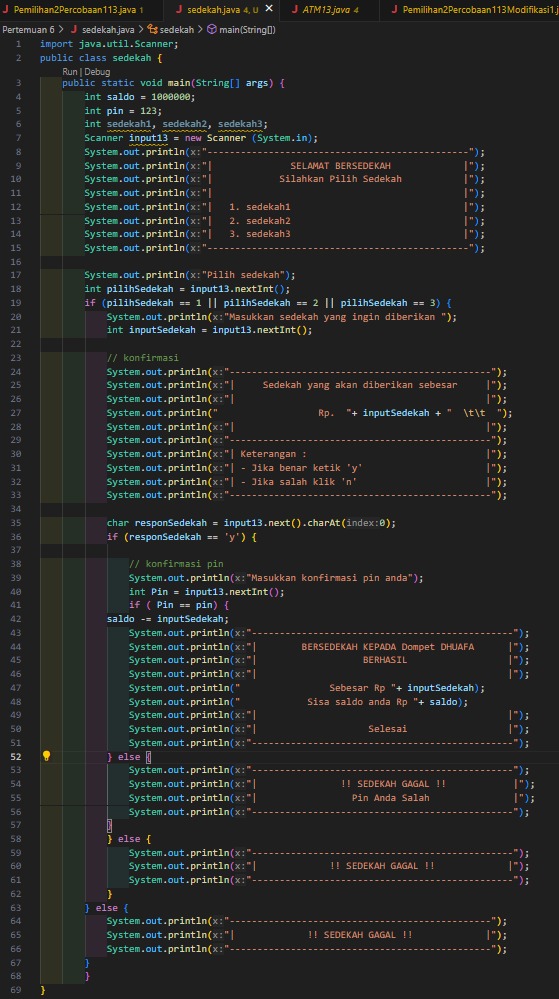
****

****

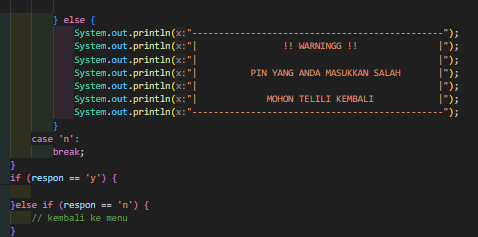
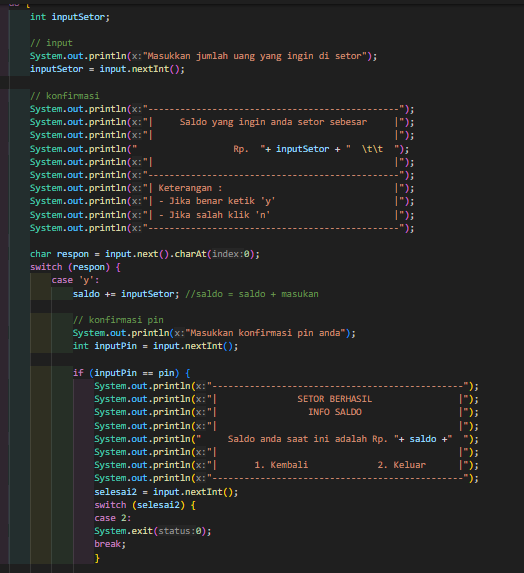
* **Login**



* **Sedekah**



* **Setor tunai**

****