

PEMILIHAN

DASAR PEMROGRAMAN - PRAKTEK



Juan Madhy

SIB 1C


Prodi Sistem Informasi Bisnis
Jurusan Teknologi Informasi
Politeknik Negeri Malang

Pertanyaan

1. Mengapa kondisi pada struktur IF tersebut tidak memerlukan kondisi dengan operator relasional?
2. Saat program dijalankan, kemudian Anda mengisi nilai false, bagaimana hasilnya?
3. Sistem perlu memberikan informasi apabila pengguna memasukkan nilai false, maka terdapat keluaran "Registrasi ditolak. Silakan lunasi UKT terlebih dahulu". Modifikasi program tersebut dengan menambahkan struktur ELSE!
4. Commit dan push hasil modifikasi Anda ke Github dengan pesan "Modifikasi Percobaan 1"
5. Modifikasi program tersebut menggunakan ternary operator
6. Commit dan push hasil modifikasi ke Github dengan pesan "Percobaan 1 Ternary"

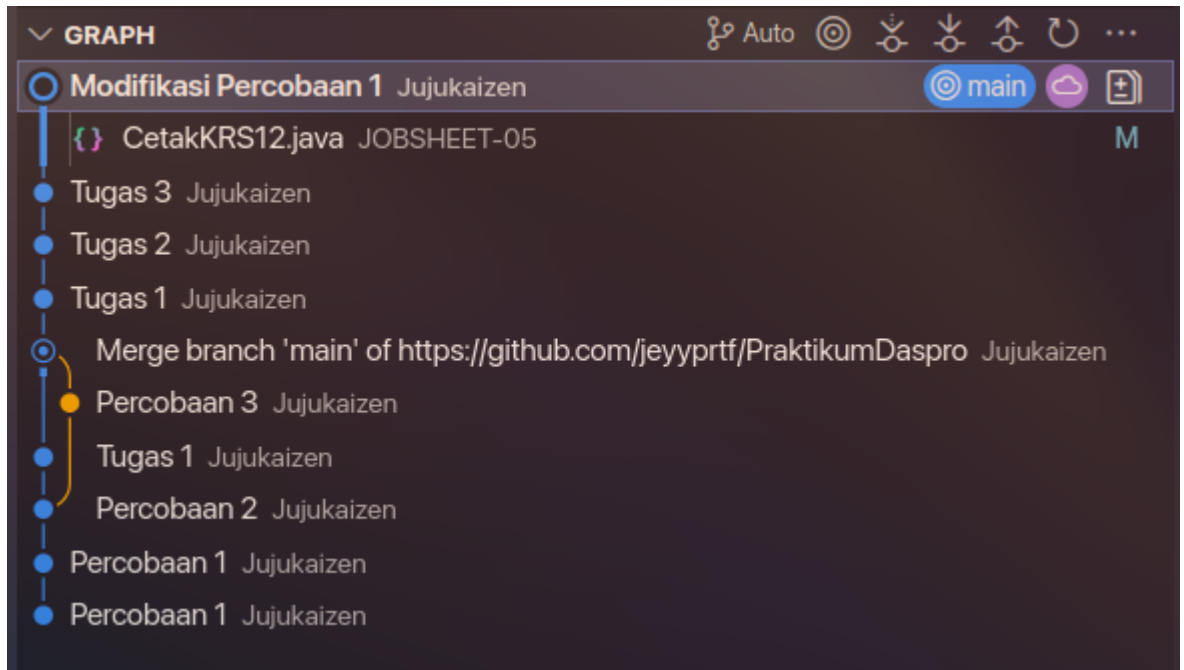
Jawaban

1. Karena pada struktur IF tersebut menggunakan variabel uktLunas, yang dimana variabel tersebut sudah bertipe data boolean. Boolean sendiri hanya memiliki 2 nilai, yaitu true dan false, jadi saat struktur IF tersebut sudah memanggil variabel bertipe data boolean, maka tidak lagi memerlukan operator relasional.
2. Jika nilai yang diisikan adalah false, maka program akan selesai, karena blok IF tersebut hanya dijalankan jika variabel bernilai true, dan setelah blok IF, tidak ada perintah lain yang dijalankan, sehingga program langsung berakhir.



```
1 import java.util.Scanner;
2 public class CetakKRS12 {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         System.out.println("--- Cetak KRS ---");
6         System.out.print("Apakah UKT sudah dibayar? (true/false): ");
7         boolean uktLunas = sc.nextBoolean();
8         if (uktLunas) {
9             System.out.println("Pembayaran UKT terverifikasi!");
10            System.out.println("Silahkan cetak KRS dan minta tanda tangan DPA");
11        } else {
12            System.out.println("Registrasi ditolak. Silakan lunasi UKT terlebih dahulu");
13        }
14        sc.close();
15    }
16 }
```

3.



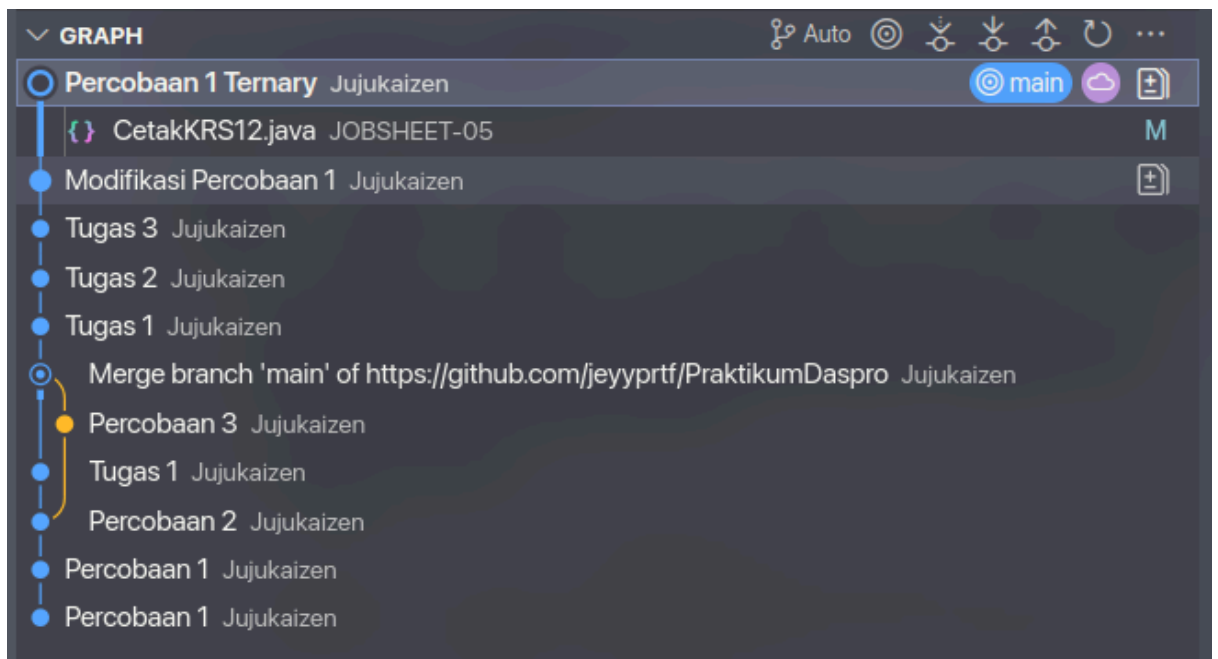
4.

```

1 import java.util.Scanner;
2 public class CetakKRS12 {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         System.out.println("---- Cetak KRS ----");
6         System.out.print("Apakah UKT sudah dibayar? (true/false): ");
7         boolean uktLunas = sc.nextBoolean();
8         String message = uktLunas ? "Pembayaran UKT terverifikasi!\nSilahkan cetak KRS dan minta tanda tangan DPA" : "Registrasi ditolak. Silakan lunasi UKT terlebih dahulu!";
9         System.out.println(message);
10        sc.close();
11    }
12 }

```

5.



6.

Pertanyaan

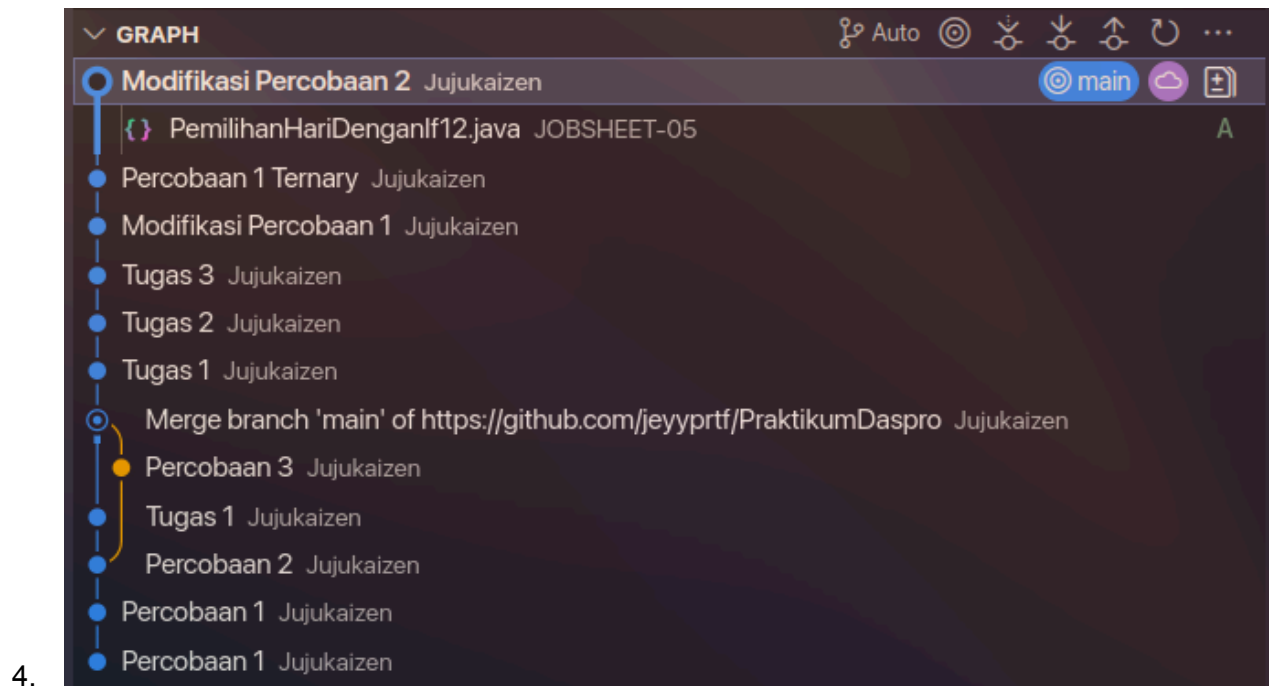
1. Apa fungsi dari break;?
2. Apa kegunaan dari toLowerCase()? Apa efeknya jika program tidak menggunakan toLowerCase()?
3. Buat program baru dengan nama PemilihanHariDenganIf<NoAbsen>.java. Program menerima input bilangan bulat. Jika angka 1, 2, 3, 4, 5 maka output yang ditampilkan adalah "Weekday" sedangkan untuk 6 dan 7 adalah "Weekend". Selain angka tersebut, tampilkan "Invalid Number". Gunakan struktur IF-ELSE IF-ELSE.
4. Commit dan push file tersebut ke github dengan comment "Modifikasi percobaan 2"

Jawaban

1. Fungsi dari break adalah untuk keluar dari blok switch setelah satu kasus selesai untuk mencegah eksekusi case selanjutnya secara otomatis.
2. Kegunaan dari toLowerCase() digunakan untuk membuat atau memformat nilai dari variabel yang berupa huruf menjadi huruf kecil atau non kapital. Jika program tidak menggunakan toLowerCase(), maka saat user memasukkan nilai ke dalam variabel dayName berupa atau mengandung huruf kapital, maka case yang ada pada blok switch tidak akan di eksekusi, karena trigger dari masing masing case menggunakan huruf kecil atau non kapital.

3.

```
1 import java.util.Scanner;
2 public class PemilihanHariDenganIf12 {
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5         String dayName, dayType;
6         System.out.print("Input day name: ");
7         dayName = sc.nextLine();
8         if (dayName.equalsIgnoreCase("monday") || dayName.equalsIgnoreCase("tuesday") || dayName.equalsIgnoreCase("wednesday") || dayName.equalsIgnoreCase("thursday") || dayName.equalsIgnoreCase("friday")) {
9             dayType = "Weekday";
10        } else if (dayName.equalsIgnoreCase("saturday") || dayName.equalsIgnoreCase("sunday")) {
11            dayType = "Weekend";
12        } else {
13            dayType = "Invalid day name";
14        }
15        System.out.println(dayName + " is a " + dayType);
16        sc.close();
17    }
18 }
19
```



Pertanyaan

1. Apa yang terjadi jika mahasiswa menjawab "No" pada pertanyaan bebas komponen?
2. Apa kegunaan dari `equalsIgnoreCase()`? Jelaskan perbedaannya dengan `toLowerCase()`!
3. Jelaskan maksud dari potongan kode berikut!
 - a. `if (bimbinganP1 >= 8 && bimbinganP2 >= 4) {`

Jawaban

1. Program akan tetap dijalankan dengan normal hingga blok `if else`, lalu akan menjalankan pesan = "Mahasiswa belum boleh ujian skripsi, masih ada tanggungan komponen.", karena blok `if` hanya mencocokkan nilai variabel dengan nilai "Ya". Jadi jika diinputkan nilai selain ya, maka akan eksekusi blok `else`.
2. Kegunaan dari `equalsIgnoreCase()` adalah untuk membandingkan nilai dari variabel `bebasKompen` dengan variabel yang ditentukan yaitu ("Ya") tanpa memedulikan huruf kapital atau non kapital. Sedangkan `toLowerCase()` sendiri hanya mengubah format variabel berupa huruf menjadi huruf kecil. Perbedaannya adalah, jika `EqualsIgnoreCase()` membandingkan nilai 2 variabel tanpa mempedulikan huruf besar atau kecil, `toLowerCase()` hanyalah fungsi untuk membuat atau memformat sebuah nilai variabel berupa teks menjadi huruf kecil.
3. Maksud dari potongan kode tersebut adalah pengkondisian jika nilai dari variabel `bimbinganP1` lebih dari atau sama dengan 8 dan `bimbinganP2` lebih dari atau sama dengan 4, maka menjalankan perintah pada blok `if` tersebut.