

**LAPORAN PRAKTIKUM**  
**MATA KULIAH PRAKTIKUM DASAR PEMROGRAMAN**

Dosen Pengampu : Triana Fatmawati, S.T, M.T

**PERTEMUAN 6 : PEMILIHAN 2**



Nama : Yonanda Mayla Rusdiaty

NIM : 2341760184

Prodi : D-IV Sistem Informasi Bisnis

**JURUSAN TEKNOLOGI INFORMASI**  
**POLITEKNIK NEGERI MALANG**  
**2023**

## 2.1 Percobaan 1 : Menentukan Tahun Kabisat Atau Bukan

Terlebih dahulu membuka aplikasi text editor dan membuat sebuah file yang diberi nama **Pemilihan2Percobaan129.java**. Kemudian tuliskan struktur dasar bahasa java dan import library scanner.

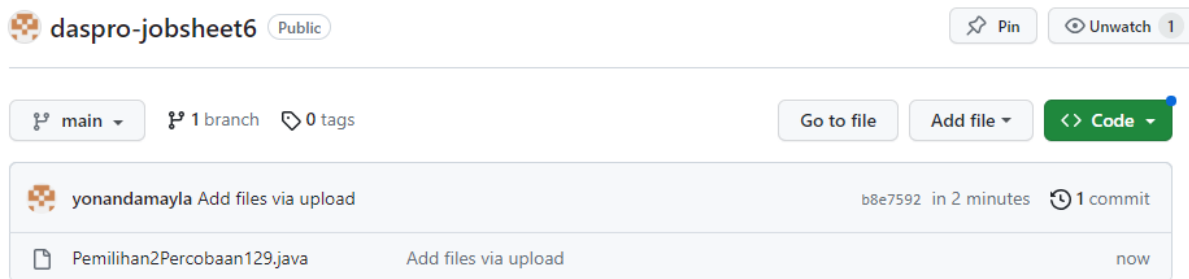
Kode Program :

```
1  import java.util.Scanner;
2
3  public class Pemilihan2Percobaan129 {
4      public static void main(String[] args) {
5          Scanner input29 = new Scanner(System.in);
6          System.out.print("Masukkan Tahun : ");
7          int tahun = input29.nextInt();
8
9          if ((tahun % 4 ) == 0) {
10             if ((tahun % 100) != 0)
11                 System.out.println("Tahun Kabisat");
12             } else {
13                 System.out.println("Bukan Tahun Kabisat");
14             }
15         }
16     }
```

Output :

```
Masukkan Tahun : 2004
Tahun Kabisat
PS D:\DASPRO\daspro-jobsheet6>
```

Push dan commit ke repository github :



### Pertanyaan

1. Bagaimana outputnya ketika diberikan input tahun 2100 ? Jelaskan! Bagaimana agar output sesuai dengan ketentuan (Tahun 2100 bukan tahun kabisat)
2. Modifikasi program sesuai jawaban no 1 !
3. Push dan commit hasil modifikasi anda ke repository !
4. 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)
5. Push dan commit hasil modifikasi anda ke repository !

### Jawaban :

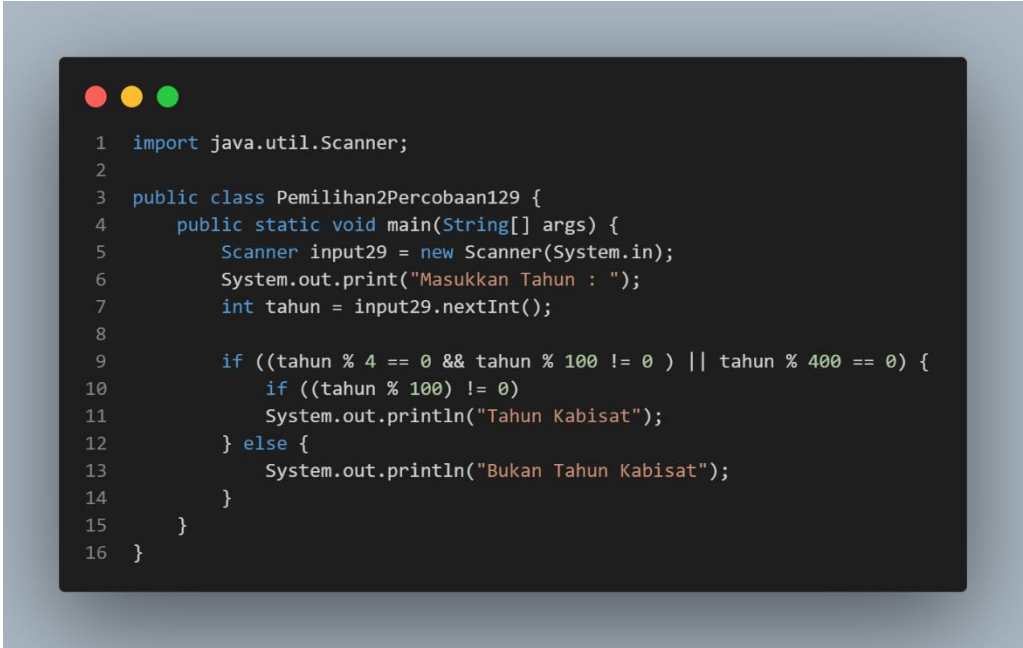
1. Tampilan output jika diberi input 2100

```
Masukkan Tahun : 2100
PS D:\DASPRO\daspro-jobsheet6> |
```

**Penjelasan :** Input 2100 tidak akan menghasilkan output yang benar karena kondisi pada kode program hanya memeriksa apakah nilai dari variabel tahun tersebut habis dibagi oleh 4 dan tidak habis dibagi oleh 100

Agar output sesuai ketentuan tahun 2100 bukan tahun kabisat (meskipun habis dibagi oleh 4) maka perlu memodifikasi kode program kondisi di dalam if menjadi ((tahun % 4 == 0 && tahun % 100 != 0) || tahun % 400 == 0). Kondisi ini memiliki arti tahun 2100 jika dibagi 4 akan habis dan tidak habis jika dibagi 100 namun habis jika dibagi 400.

2. Modifikasi kode program dari no 1 menjadi seperti berikut ini :

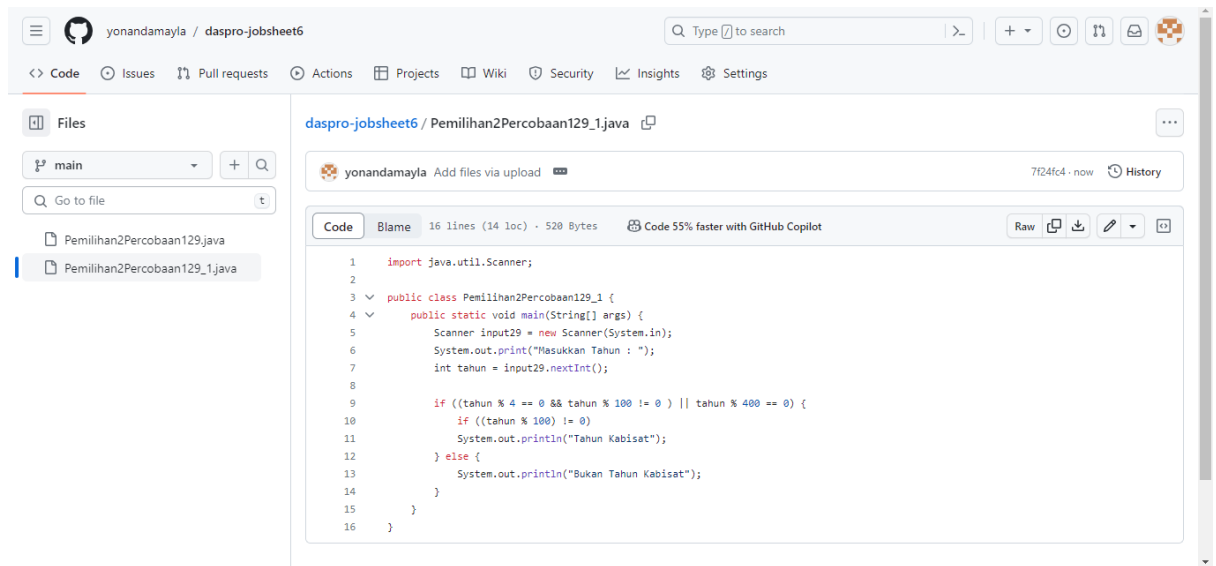


```
1  import java.util.Scanner;
2
3  public class Pemilihan2Percobaan129 {
4      public static void main(String[] args) {
5          Scanner input29 = new Scanner(System.in);
6          System.out.print("Masukkan Tahun : ");
7          int tahun = input29.nextInt();
8
9          if ((tahun % 4 == 0 && tahun % 100 != 0 ) || tahun % 400 == 0) {
10             if ((tahun % 100) != 0)
11                 System.out.println("Tahun Kabisat");
12             } else {
13                 System.out.println("Bukan Tahun Kabisat");
14             }
15         }
16     }
```

Output :

```
Masukkan Tahun : 2100
Bukan Tahun Kabisat
PS D:\DASPRO\daspro-jobsheet6> |
```

### 3. Push dan commit hasil modifikasi ke repository github :



The screenshot displays the GitHub interface for the repository 'yonandamayla / daspro-jobsheet6'. The file 'Pemilihan2Percobaan129\_1.java' is selected, showing 16 lines of Java code. The code is as follows:

```
1 import java.util.Scanner;
2
3 public class Pemilihan2Percobaan129_1 {
4     public static void main(String[] args) {
5         Scanner input29 = new Scanner(System.in);
6         System.out.print("Masukkan Tahun : ");
7         int tahun = input29.nextInt();
8
9         if ((tahun % 4 == 0 && tahun % 100 != 0) || tahun % 400 == 0) {
10             if ((tahun % 100) != 0)
11                 System.out.println("Tahun Kabisat");
12             else {
13                 System.out.println("Bukan Tahun Kabisat");
14             }
15         }
16     }
17 }
```

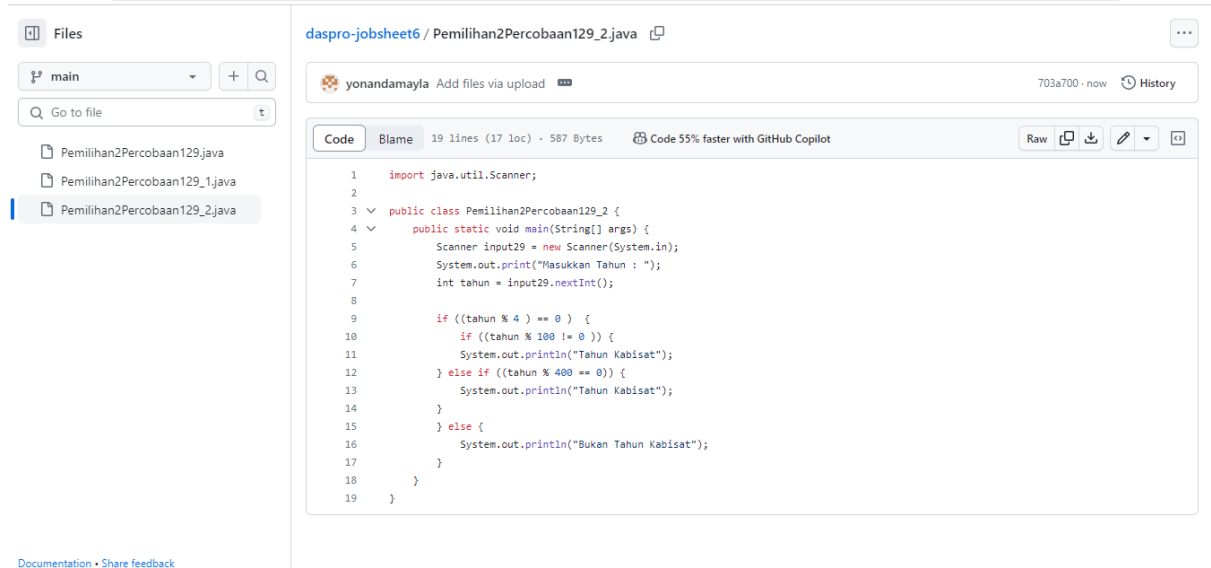
4. Modifikasi program tanpa menggunakan operator logika menjadi seperti ini :

```
1  import java.util.Scanner;
2
3  public class Pemilihan2Percobaan129_2 {
4      public static void main(String[] args) {
5          Scanner input29 = new Scanner(System.in);
6          System.out.print("Masukkan Tahun : ");
7          int tahun = input29.nextInt();
8
9          if ((tahun % 4 ) == 0 ) {
10             if ((tahun % 100 != 0 )) {
11                 System.out.println("Tahun Kabisat");
12             } else if ((tahun % 400 == 0)) {
13                 System.out.println("Tahun Kabisat");
14             }
15             } else {
16                 System.out.println("Bukan Tahun Kabisat");
17             }
18         }
19     }
```

Output :

```
Masukkan Tahun : 2000
Tahun Kabisat
PS D:\DASPRO\daspro-jobsheet6> 
```

## 5. Push dan commit hasil modifikasi ke repository github :



The screenshot displays a GitHub repository interface for the repository named `daspro-jobsheet6`. The file `Pemilihan2Percobaan129_2.java` is selected, showing its contents in a code editor. The file is located in the `main` branch. The interface includes a sidebar with a file explorer, a top bar with repository information, and a main area for the code editor. The code is a Java program that checks if a given year is a leap year (Kabisat) or not.

**Files**

- Pemilihan2Percobaan129.java
- Pemilihan2Percobaan129\_1.java
- Pemilihan2Percobaan129\_2.java**

**daspro-jobsheet6 / Pemilihan2Percobaan129\_2.java**

yonandamayla Add files via upload 703a700 · now History

**Code** Blame 19 lines (17 loc) · 587 Bytes Code 55% faster with GitHub Copilot Raw Download Edit

```
1  import java.util.Scanner;
2
3  public class Pemilihan2Percobaan129_2 {
4  public static void main(String[] args) {
5      Scanner input29 = new Scanner(System.in);
6      System.out.print("Masukkan Tahun : ");
7      int tahun = input29.nextInt();
8
9      if ((tahun % 4) == 0) {
10         if ((tahun % 100 != 0)) {
11             System.out.println("Tahun Kabisat");
12         } else if ((tahun % 400 == 0)) {
13             System.out.println("Tahun Kabisat");
14         }
15     } else {
16         System.out.println("Bukan Tahun Kabisat");
17     }
18 }
19 }
```

[Documentation](#) • [Share feedback](#)

## 2.2 Percobaan 2 : Menentukan Jenis Bidang Datar Segitiga dari Masukan Ketiga Sudutnya

Kode program :

```
1  import java.util.Scanner;
2
3  public class Pemilihan2Percobaan229 {
4      public static void main(String[] args) {
5          Scanner input29 = new Scanner(System.in);
6          float sudut1, sudut2, sudut3, totalSudut;
7
8
9          System.out.print("Masukkan sudut 1 : ");
10         sudut1 = input29.nextFloat();
11         System.out.print("Masukkan sudut 2 : ");
12         sudut2 = input29.nextFloat();
13         System.out.print("Masukkan sudut 3 : ");
14         sudut3 = input29.nextFloat();
15
16         totalSudut = sudut1 + sudut2 + sudut3;
17
18         if (totalSudut == 180) {
19             if ((sudut1 == 90) || (sudut2 == 90) || (sudut3 == 90))
20                 System.out.println("Segitiga tersebut adalah segitiga siku-siku");
21             else {
22                 System.out.println("Segitiga tersebut bukan segitiga siku-siku");
23             }
24         } else {
25             System.out.println("Bukan segitiga");
26         }
27     }
28 }
```



Output :

```
Masukkan sudut 1 : 90
Masukkan sudut 2 : 30
Masukkan sudut 3 : 60
Segitiga tersebut adalah segitiga siku-siku
PS D:\DASPRO\daspro-jobsheet6>
```

Push dan commit kode program ke repository Github :



The screenshot displays a GitHub repository interface. On the left, the 'Files' sidebar shows a list of files: 'Pemilihan2Percobaan129.java', 'Pemilihan2Percobaan129\_1.java', 'Pemilihan2Percobaan129\_2.java', and 'Pemilihan2Percobaan229.java'. The main area shows the code for 'Pemilihan2Percobaan229.java'. The code is a Java program that prompts the user to input three angles and then checks if they form a right-angled triangle (segitiga siku-siku).

```
1 import java.util.Scanner;
2
3 public class Pemilihan2Percobaan229 {
4     public static void main(String[] args) {
5         Scanner input29 = new Scanner(System.in);
6         float sudut1, sudut2, sudut3, totalSudut;
7
8
9         System.out.print("Masukkan sudut 1 : ");
10        sudut1 = input29.nextFloat();
11        System.out.print("Masukkan sudut 2 : ");
12        sudut2 = input29.nextFloat();
13        System.out.print("Masukkan sudut 3 : ");
14        sudut3 = input29.nextFloat();
15
16        totalSudut = sudut1 + sudut2 + sudut3;
17
18        if (totalSudut == 180) {
19            if ((sudut1 == 90) || (sudut2 == 90) || (sudut3 == 90))
20                System.out.println("Segitiga tersebut adalah segitiga siku-siku");
21            else {
22                System.out.println("Segitiga tersebut bukan segitiga siku-siku");
23            }
24        } else {
25            System.out.println("Bukan segitiga");
26        }
27    }
28 }
```

## Pertanyaan

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.
2. Push dan commit hasil modifikasi anda ke repository

### Jawaban :

1. Modifikasi kode program untuk dapat mengeluarkan output jenis segitiga yang lain (selain segitiga siku-siku), disini saya menambahkan segitiga sama sisi, segitiga sama kaki, dan segitiga sembarang. Kode program :

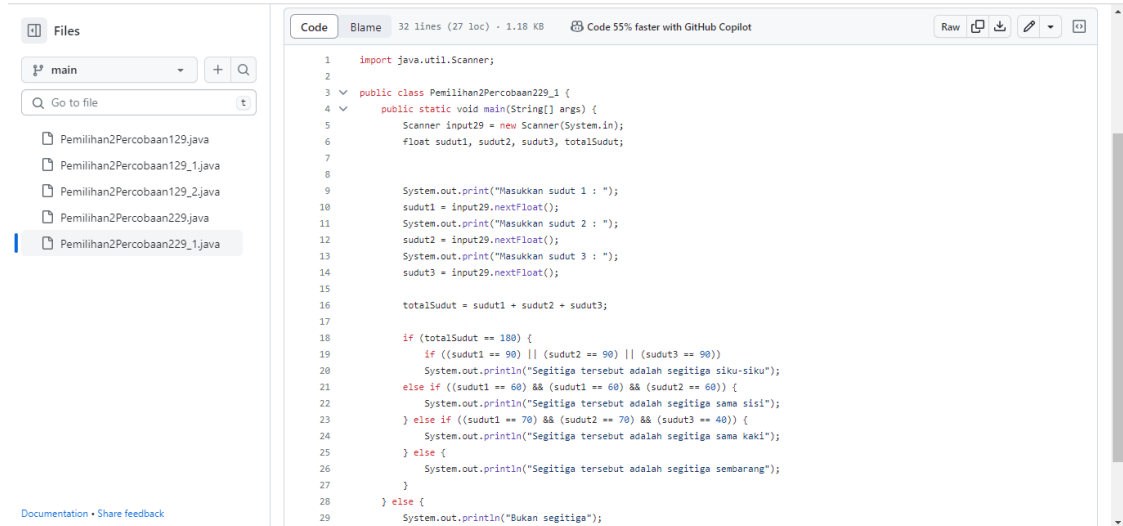
```
1  import java.util.Scanner;
2
3  public class Pemilihan2Percobaan229_1 {
4      public static void main(String[] args) {
5          Scanner input29 = new Scanner(System.in);
6          float sudut1, sudut2, sudut3, totalSudut;
7
8
9          System.out.print("Masukkan sudut 1 : ");
10         sudut1 = input29.nextFloat();
11         System.out.print("Masukkan sudut 2 : ");
12         sudut2 = input29.nextFloat();
13         System.out.print("Masukkan sudut 3 : ");
14         sudut3 = input29.nextFloat();
15
16         totalSudut = sudut1 + sudut2 + sudut3;
17
18         if (totalSudut == 180) {
19             if ((sudut1 == 90) || (sudut2 == 90) || (sudut3 == 90))
20                 System.out.println("Segitiga tersebut adalah segitiga siku-siku");
21             else if ((sudut1 == 60) && (sudut1 == 60) && (sudut2 == 60)) {
22                 System.out.println("Segitiga tersebut adalah segitiga sama sisi");
23             } else if ((sudut1 == 70) && (sudut2 == 70) && (sudut3 == 40)) {
24                 System.out.println("Segitiga tersebut adalah segitiga sama kaki");
25             } else {
26                 System.out.println("Segitiga tersebut adalah segitiga sembarang");
27             }
28         } else {
29             System.out.println("Bukan segitiga");
30         }
31     }
32 }
```

Output :

```
Masukkan sudut 1 : 60
Masukkan sudut 2 : 60
Masukkan sudut 3 : 60
Segitiga tersebut adalah segitiga sama sisi
PS D:\DASPRO\daspro-jobsheet6>
```

```
Masukkan sudut 1 : 70
Masukkan sudut 2 : 70
Masukkan sudut 3 : 40
Segitiga tersebut adalah segitiga sama kaki
PS D:\DASPRO\daspro-jobsheet6>
```

2. Push dan commit hasil modifikasi program ke repository Github :



```
1  import java.util.Scanner;
2
3  public class Pemilihan2Percobaan229_1 {
4      public static void main(String[] args) {
5          Scanner input29 = new Scanner(System.in);
6          float sudut1, sudut2, sudut3, totalSudut;
7
8
9          System.out.print("Masukkan sudut 1 : ");
10         sudut1 = input29.nextFloat();
11         System.out.print("Masukkan sudut 2 : ");
12         sudut2 = input29.nextFloat();
13         System.out.print("Masukkan sudut 3 : ");
14         sudut3 = input29.nextFloat();
15
16         totalSudut = sudut1 + sudut2 + sudut3;
17
18         if (totalSudut == 180) {
19             if ((sudut1 == 90) || (sudut2 == 90) || (sudut3 == 90))
20                 System.out.println("Segitiga tersebut adalah segitiga siku-siku");
21             else if ((sudut1 == 60) && (sudut1 == 60) && (sudut2 == 60)) {
22                 System.out.println("Segitiga tersebut adalah segitiga sama sisi");
23             } else if ((sudut1 == 70) && (sudut2 == 70) && (sudut3 == 40)) {
24                 System.out.println("Segitiga tersebut adalah segitiga sama kaki");
25             } else {
26                 System.out.println("Segitiga tersebut adalah segitiga sembarang");
27             }
28         } else {
29             System.out.println("Bukan segitiga");
```

## 2.3 Percobaan 3 : Menghitung gaji bersih seseorang setelah dipotong pajak sesuai dengan kategori jabatan

Kode program :

```
1  import java.util.Scanner;
2
3  public class Pemilihan2Percobaan329 {
4      public static void main(String[] args) {
5          Scanner input29 = new Scanner(System.in);
6          String kategori;
7          int penghasilan, gajiBersih;
8          double pajak = 0;
9
10         System.out.print("Masukkan kategori : ");
11         kategori = input29.nextLine();
12         System.out.print("Masukkan besarnya penghasilan : ");
13         penghasilan = input29.nextInt();
14
15         if (kategori.equalsIgnoreCase ("pekerja")) {
16             if (penghasilan <= 2000000) {
17                 pajak = 0.1;
18             } else if (penghasilan <= 3000000) {
19                 pajak = 0.15;
20             } else {
21                 pajak = 0.2;
22             }
23             gajiBersih = (int) (penghasilan - (pajak * penghasilan));
24             System.out.println("Penghasilan bersih : " + gajiBersih);
25         } else if (kategori.equalsIgnoreCase("pebisnis")) {
26             if (penghasilan <= 2500000) {
27                 pajak = 0.15;
28             } else if (penghasilan <= 3500000) {
29                 pajak = 0.2;
30             } else {
31                 pajak = 0.25;
32             }
33             gajiBersih = (int) (penghasilan - (pajak * penghasilan));
34             System.out.println("Penghasilan bersih : " + gajiBersih );
35         } else {
36             System.out.println("Masukkan kategori salah");
37         }
38     }
39 }
40
41 }
42
```

Output :

```
Masukkan kategori : pebisnis
Masukkan besarnya penghasilan : 2000000
Penghasilan bersih : 1700000
PS D:\DASPRO\daspro-jobsheet6>
```

Push dan commit hasil program ke repository Github :



The screenshot shows a GitHub web interface for a repository named 'daspro-jobsheet6'. The file 'Pemilihan2Percobaan329.java' is selected, showing its code. The code is a Java program that prompts the user for a category and income, then calculates the net income based on the category and income. The code is as follows:

```
10      System.out.print("Masukkan kategori : ");
11      kategori = input29.nextLine();
12      System.out.print("Masukkan besarnya penghasilan : ");
13      penghasilan = input29.nextInt();
14
15      if (kategori.equalsIgnoreCase("pekerja")) {
16          if (penghasilan <= 2000000) {
17              pajak = 0.1;
18          } else if (penghasilan <= 3000000) {
19              pajak = 0.15;
20          } else {
21              pajak = 0.2;
22          }
23      }
24      gajiBersih = (int) (penghasilan - (pajak * penghasilan));
25      System.out.println("Penghasilan bersih : " + gajiBersih);
26      } else if (kategori.equalsIgnoreCase("pebisnis")) {
27          if (penghasilan <= 2500000) {
28              pajak = 0.15;
29          } else if (penghasilan <= 3500000) {
30              pajak = 0.2;
31          } else {
32              pajak = 0.25;
33          }
34      }
35      gajiBersih = (int) (penghasilan - (pajak * penghasilan));
36      System.out.println("Penghasilan bersih : " + gajiBersih);
37      } else {
38          System.out.println("Masukkan kategori salah");
39      }
40  }
```

## Pertanyaan

1. Jelaskan fungsi dari (int) pada sintaks gajiBersih = (int) (penghasilan - (penghasilan \* pajak));
2. Jalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Amati apa yang terjadi! Apa kegunaan dari equalsIgnoreCase?
3. 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?

**Jawaban :**

1. (int) pada sintaks gajiBersih = (int) (penghasilan - (penghasilan \* pajak)); merupakan operasi type casting . Dalam konteks ini (**int**) digunakan untuk mengubah hasil perhitungan dari tipe data yang lebih besar (misalnya double atau float) menjadi tipe data yang lebih kecil yaitu int
2. Output dari input kategori PEBISNIS dan penghasilan 2000000 :

```
Masukkan kategori : pebisnis
Masukkan besarnya penghasilan : 2000000
Penghasilan bersih : 1700000
PS D:\DASPRO\daspro-jobsheet6>
```

```
Masukkan kategori : PEBISNIS
Masukkan besarnya penghasilan : 2000000
Penghasilan bersih : 1700000
PS D:\DASPRO\daspro-jobsheet6>
```

**Penjelasan :** program yang di outputkan akan menerima inputan dari PEBISNIS sebagai kategori dan 2000000 sebagai penghasilan. Karena penggunaan equals.IgnoreCase dalam sebuah kondisi, maka perbedaan huruf besar atau huruf kecil dalam masukan kategori tidak akan memengaruhi hasilnya. PEBISNIS akan sama saja dengan pebisnis tanpa memerhatikan besar kecilnya huruf.

3. Mengubah equalsIgnoreCase menjadi equals, kemudian menjalankan program dengan memasukkan kategori = PEBISNIS dan penghasilan = 2000000. Berikut adalah tampilannya :

```
Masukkan kategori : PEBISNIS
Masukkan besarnya penghasilan : 2000000
Masukkan kategori salah
PS D:\DASPRO\daspro-jobsheet6>
```

**Penjelasan :** Output yang dihasilkan adalah masukan kategori salah. Karena jika kita mengganti equalsIgnoreCase dengan equals kemudian menjalankan program dengan menginputkan kategori PEBISNIS dan

penghasilan 2000000, equals akan memeriksa String secara case-sensitive yang berarti huruf besar dan huruf kecil akan menjadi berbeda. Kategori PEBISNIS dengan pebisnis merupakan hal yang berbeda.

Push dan commit ke repository Github :



```
3 public class Pemilihan2Percobaan329_1 {
4     public static void main(String[] args) {
13         penghasilan = Input29.nextInt();
14
15         if (kategori.equalsIgnoreCase("pekerja")) {
16             if (penghasilan <= 2000000) {
17                 pajak = 0.1;
18             } else if (penghasilan <= 3000000) {
19                 pajak = 0.15;
20             } else {
21                 pajak = 0.2;
22             }
23             gajiBersih = (int) (penghasilan - (pajak * penghasilan));
24             System.out.println("Penghasilan bersih : " + gajiBersih);
25
26         } else if (kategori.equals("pebisnis")) {
27             if (penghasilan <= 2500000) {
28                 pajak = 0.15;
29             } else if (penghasilan <= 3500000) {
30                 pajak = 0.2;
31             } else {
32                 pajak = 0.25;
33             }
34             gajiBersih = (int) (penghasilan - (pajak * penghasilan));
35             System.out.println("Penghasilan bersih : " + gajiBersih);
36         } else {
37             System.out.println("Masukkan kategori salah");
38         }
39     }
40 }
```

## Latihan Individu 1 : Bilangan Terbesar

Kode program :

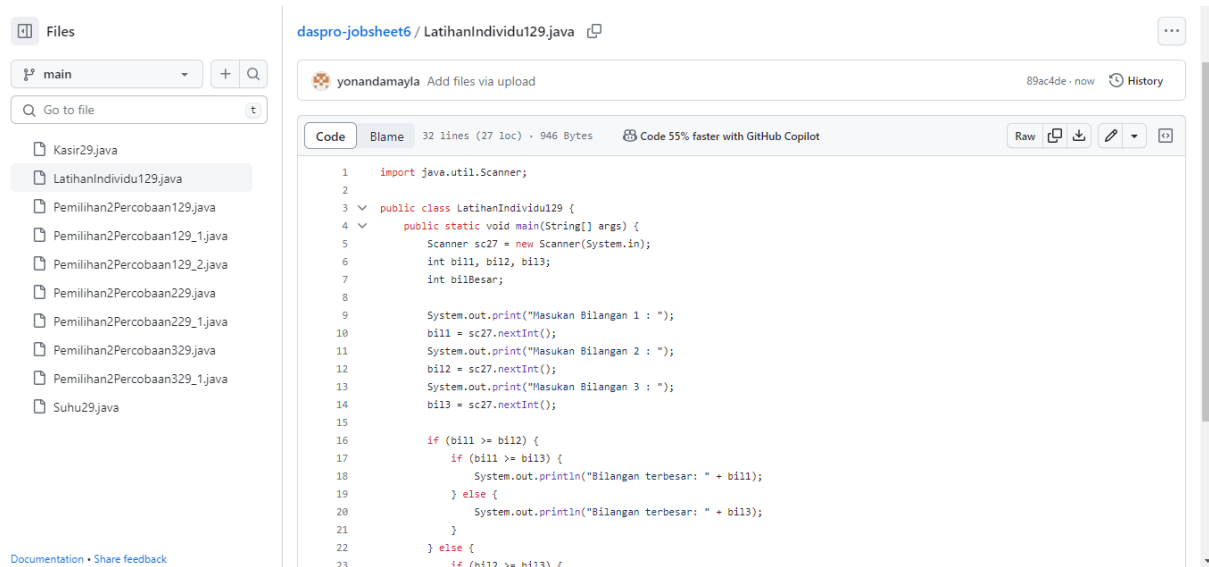
```
1  import java.util.Scanner;
2
3  public class LatihanIndividu129 {
4      public static void main(String[] args) {
5          Scanner sc27 = new Scanner(System.in);
6          int bil1, bil2, bil3;
7          int bilBesar;
8
9          System.out.print("Masukan Bilangan 1 : ");
10         bil1 = sc27.nextInt();
11         System.out.print("Masukan Bilangan 2 : ");
12         bil2 = sc27.nextInt();
13         System.out.print("Masukan Bilangan 3 : ");
14         bil3 = sc27.nextInt();
15
16         if (bil1 >= bil2) {
17             if (bil1 >= bil3) {
18                 System.out.println("Bilangan terbesar: " + bil1);
19             } else {
20                 System.out.println("Bilangan terbesar: " + bil3);
21             }
22         } else {
23             if (bil2 >= bil3) {
24                 System.out.println("Bilangan terbesar: " + bil2);
25             } else {
26                 System.out.println("Bilangan terbesar: " + bil3);
27             }
28         }
29
30     }
31
32 }
```



Output :

```
Masukan Bilangan 1 : 28
Masukan Bilangan 2 : 54
Masukan Bilangan 3 : 15
Bilangan terbesar: 54
PS D:\DASPRO\daspro-jobsheet6> █
```

Push dan commit ke repository Github :



The screenshot displays a GitHub repository interface. On the left, a file explorer shows a list of files: `Kasir29.java`, `LatihanIndividu129.java` (selected), `Pemilihan2Percobaan129.java`, `Pemilihan2Percobaan129_1.java`, `Pemilihan2Percobaan129_2.java`, `Pemilihan2Percobaan229.java`, `Pemilihan2Percobaan229_1.java`, `Pemilihan2Percobaan329.java`, `Pemilihan2Percobaan329_1.java`, and `Suhu29.java`. Below the file explorer, there are links for [Documentation](#) and [Share feedback](#).

The main area shows the code editor for `daspro-jobsheet6 / LatihanIndividu129.java`. The editor includes a header with the username `yonandamayla`, a button to `Add files via upload`, and a commit hash `89ac4de` with a `History` link. The code editor has tabs for `Code` and `Blame`, and a status bar indicating `32 lines (27 loc) · 946 Bytes` and `Code 55% faster with GitHub Copilot`. The code itself is as follows:

```
1  import java.util.Scanner;
2
3  public class LatihanIndividu129 {
4      public static void main(String[] args) {
5          Scanner sc27 = new Scanner(System.in);
6          int bi11, bi12, bi13;
7          int bi1Besar;
8
9          System.out.print("Masukan Bilangan 1 : ");
10         bi11 = sc27.nextInt();
11         System.out.print("Masukan Bilangan 2 : ");
12         bi12 = sc27.nextInt();
13         System.out.print("Masukan Bilangan 3 : ");
14         bi13 = sc27.nextInt();
15
16         if (bi11 >= bi12) {
17             if (bi11 >= bi13) {
18                 System.out.println("Bilangan terbesar: " + bi11);
19             } else {
20                 System.out.println("Bilangan terbesar: " + bi13);
21             }
22         } else {
23             if (bi12 >= bi13) {
```