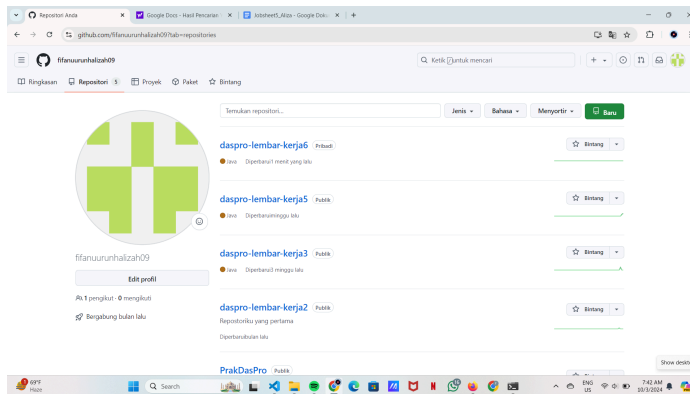


**NAMA : FIFA NUURUN HALIZAH**  
**NIM : 244107020019**  
**NO. ABSEN : 09**  
**KELAS : 1D**

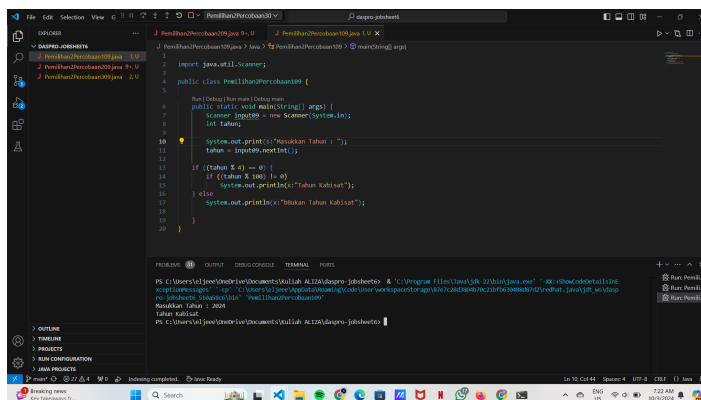
## JOBSHEET 6

### Percobaan 1

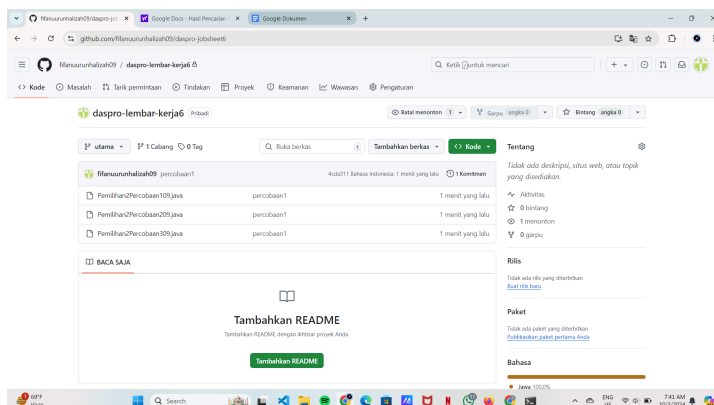
#### 1. Buat repository beri nama daspro-jobsheet6



#### 2. Masukkan kode program yang sudah ditentukan



#### 3. Commit dan push kode program ke Github



## Pertanyaan!

1. Outputnya tidak ada dikarenakan 2100 habis dibagi 4 dan juga habis dibagi 100
2. Hasil modifikasi

The screenshot displays an IDE with a Java file named `PemilihanPerobaan109.java`. The code is a menu-driven program that prompts the user to enter a number (1, 2, or 3) to select an option. It uses a `Scanner` class for input and a `switch` statement for logic. The code is as follows:



```
1  import java.util.Scanner;
2
3  public class PemilihanPerobaan109 {
4
5      Run | Debug | Run main | Debug main
6      public static void main(String[] args) {
7          Scanner inputMS = new Scanner(System.in);
8          int tahun;
9
10         System.out.println("Masukkan Tahun :");
11         tahun = inputMS.nextInt();
12
13         if ((tahun > 0) && (tahun < 100)) {
14             if ((tahun % 2) == 0) {
15                 System.out.println("Tahun Kabisat");
16             } else {
17                 System.out.println("Bukan Tahun Kabisat");
18             }
19         } else {
20             System.out.println("Bukan Tahun Kabisat");
21         }
22     }
23 }
24
```


The IDE shows the program's execution output in the `OUTPUT` tab:

```
Masukkan Tahun : 2000
PS C:\Users\user\OneDrive\Documents\Wallah ALIZA\daoprog_jobsheet5>
PS C:\Users\user\OneDrive\Documents\Wallah ALIZA\daoprog_jobsheet5>
PS C:\Users\user\OneDrive\Documents\Wallah ALIZA\daoprog_jobsheet5> cd "C:\Users\user\OneDrive\Documents\Wallah ALIZA\daoprog_jobsheet5" & "C:\Program Files\Java\jdk-22\bin\java.exe" -Xms256m -Xmx512m -cp .\daoprog_jobsheet5.jar -Djava.class.path=. daoprog_jobsheet5.jar
Masukkan Tahun : 2000
Tahun Kabisat
PS C:\Users\user\OneDrive\Documents\Wallah ALIZA\daoprog_jobsheet5>
```

The bottom status bar indicates the program is running on `Java 17` with `UTF-8` encoding.

- ### 3. Push dan commit hasil modifikasi ke github

 **fifanuurunhalizah09** modifikasiPercobaan1 58f5794 Bahasa Indonesia: 15 menit yang lalu 

 Pemilihan2Percobaan109.java modifikasiPercobaan1 15 menit yang lalu

- #### 4. Hasil modifikasi

The screenshot shows an IDE window titled "J -> FibonacciSequences01.java". The code defines a class `FibonacciSequences01` with a `main` method that takes command-line arguments. It uses `Scanner` to read input, calculates the Fibonacci sequence up to the specified number, and prints the result.

```
1 public class FibonacciSequences01 {
2     public static void main(String[] args) {
3         Scanner input00 = new Scanner(System.in);
4         int Tahun;
5
6         System.out.println("Masukkan Tahun : ");
7         Tahun = input00.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             } else {
15                 System.out.println("Bukan Tahun Kabisat");
16             }
17         } else {
18             System.out.println("Bukan Tahun Kabisat");
19         }
20     }
21 }
22
23
24
25 }
```

The bottom panel shows the terminal output:

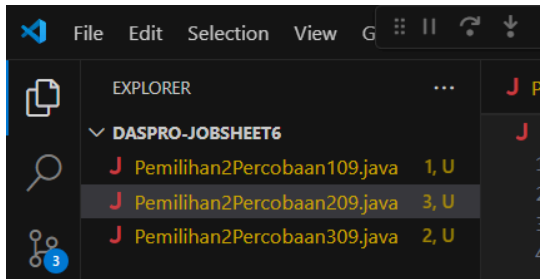
```
-o origin main
Enumerating objects: 5, done.
Counting objects: 100%, done.
Delta compression using up to 12 threads
Compressing objects: 100% (0/0), done.
Writing objects: 100% (100 / 360 bytes) | 360.00 KiB/s, done.
Total: 100% (delta 2), reused 0 (delta 0), compacted 0 (rew 0)
remote: Resolving deltas: 100% (1/2), completed with 2 local objects.
To https://github.com/Furamansalbi2009/dagpro-jobsheet.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\lab\source\repos\Wallah Aliza\dagpro-jobsheet>
J main C:\> Run Code Run Debug View Ready
```

- ## 5. Push dan commit hasil modifikasi ke github

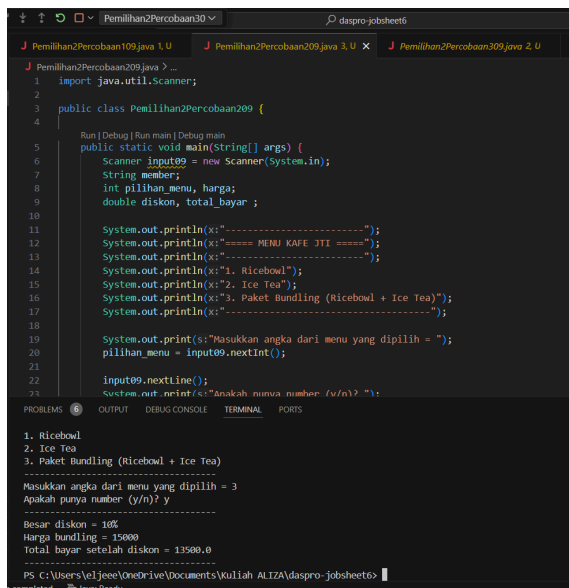
fifanuurunhalizah09 modifikasikKe2 66fe728 Bahasa Indonesia: sekarang

## Percobaan 2

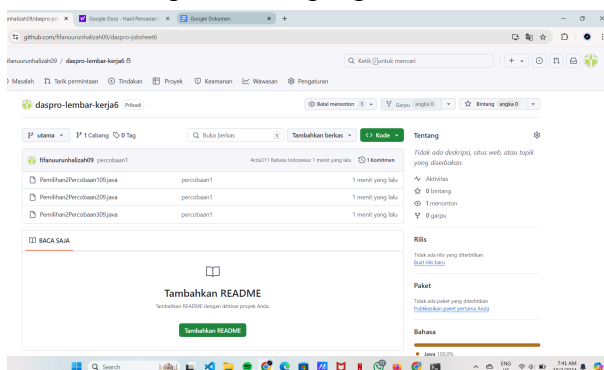
1. Buat file baru dan beri nama Pemilihan2Percobaan209.java



2. Masukkan kode program yang sudah ditentukan



3. Commit dan push kode program ke Github



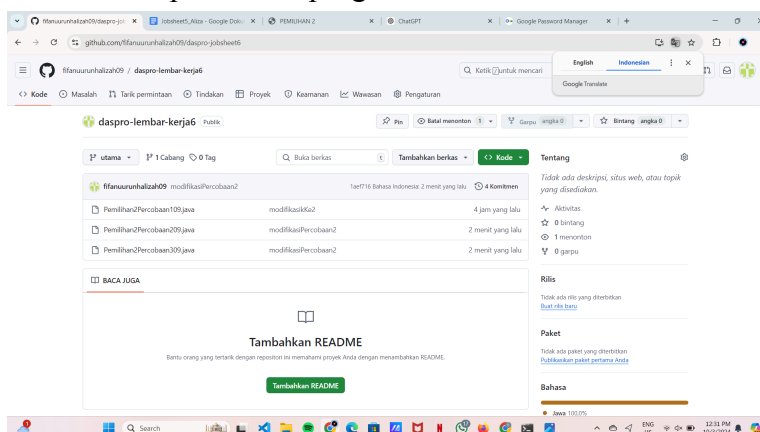
## Pertanyaan!

1. `method.equalsIgnoreCase()` digunakan untuk membandingkan dua string tanpa memperhatikan perbedaan huruf besar dan kecil. Dalam seleksi kondisi member metode ini digunakan untuk mengecek apakah nilai variable member sama dengan “y” atau “n” yang mewakili apakah pembeli memiliki member atau tidak, tanpa perlu mempertimbangkan apakah hurufnya kecil atau besar.

- Perbedaan utama antara `method.equals()` dan `.equalsIgnoreCase()` yaitu Metode `equals()` membandingkan string secara case-sensitive, sehingga perbedaan dalam huruf besar dan kecil akan mengakibatkan pengembalian nilai false, sementara `.equalsIgnoreCase()` membandingkan string secara case-insensitive, mengabaikan perbedaan tersebut dan mengembalikan true jika isi stringnya sama.
- Kode tersebut digunakan untuk memastikan bahwa program membaca input secara akurat dan menghindari kesalahan yang timbul dikarenakan perilaku `input09.nextInt()`;
- Hasil output

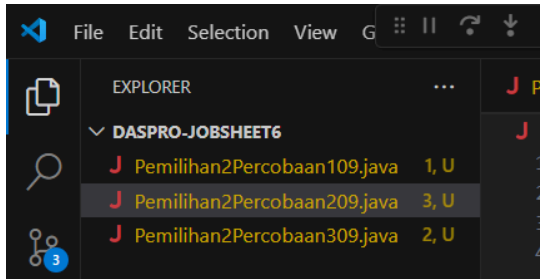
- Hasil modifikasi

- Commit dan push code program ke Github

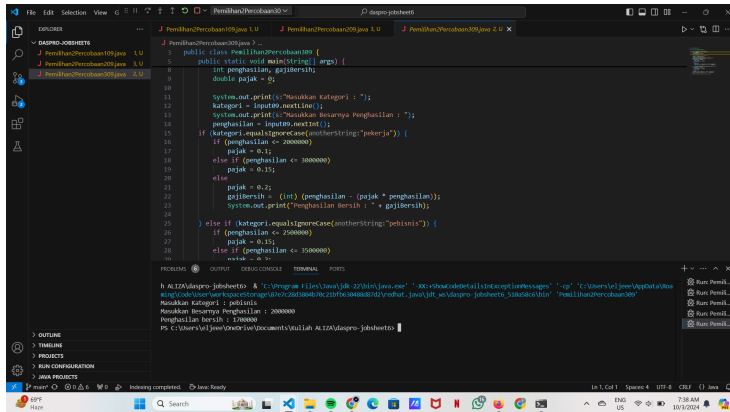


## Percobaan 3

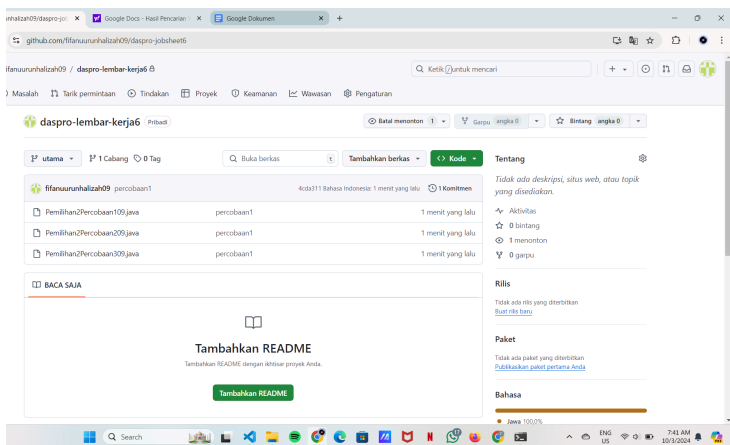
- Buat file baru dan beri nama 'Pemilihan2Percobaan309.java'



## 2. Masukkan kode program yang sudah ditentukan



## 3. Commit dan push kode program ke Github

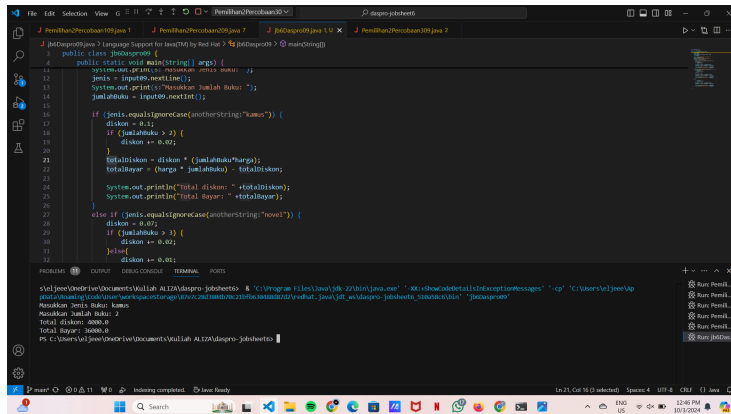


**Pertanyaan!**

1. Operasi (int) dalam sintaks (int) (penghasilan - (penghasilan \* pajak)) adalah proses konversi tipe data ke integer. Ini membulatkan nilai ke bawah ke bilangan bulat terdekat.
2. Kegunaan equalsIgnoreCase adalah metode yang membandingkan dua string tanpa memperhatikan huruf besar-kecil. Artinya, metode ini menganggap dua string dengan karakter yang sama, namun berbeda huruf besar-kecil, sebagai string yang sama.
3. Output yang dihasilkan “masukkan kategori salah” dikarenakan tidak memperhatikan huruf besar atau kecilnya. Equals adalah metode yang digunakan untuk memeriksa apakah dua objek atau nilai memiliki nilai yang sama. Pada string, metode ini membandingkan isi karakter dari dua string.

# TUGAS!

## 1. Masukkan kode program yang sudah ditentukan

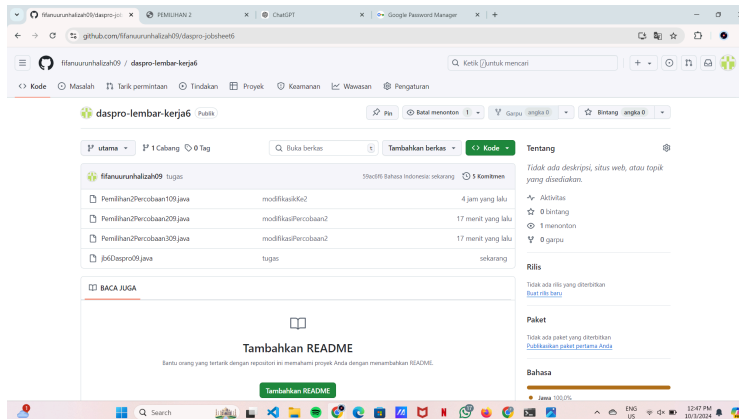


```
1 public class J04Despro09 {
2     public static void main(String[] args) {
3         Scanner input = new Scanner(System.in);
4         int jenis;
5         int jumlahBuku;
6         double diskon;
7         double totalBayar;
8         double totalDiskon;
9
10        System.out.println("Masukkan jenis buku: ");
11        jenis = input.nextInt();
12
13        System.out.println("Masukkan jumlah buku: ");
14        jumlahBuku = input.nextInt();
15
16        if (jenis == 1) {
17            diskon = 0.1;
18            if (jumlahBuku > 2) {
19                diskon += 0.02;
20            }
21            totalDiskon = diskon * jumlahBuku;
22            totalBayar = (jumlahBuku * 10000) - totalDiskon;
23            System.out.println("Total diskon: " + totalDiskon);
24            System.out.println("Total bayar: " + totalBayar);
25        } else if (jenis == 2) {
26            diskon = 0.05;
27            if (jumlahBuku > 3) {
28                diskon += 0.02;
29            }
30            totalDiskon = diskon * jumlahBuku;
31            totalBayar = (jumlahBuku * 10000) - totalDiskon;
32            System.out.println("Total diskon: " + totalDiskon);
33            System.out.println("Total bayar: " + totalBayar);
34        }
35    }
36 }
```

Output:

```
PS C:\Users\alifah\Documents\Udah\ALIDA\despro-j04despro> java J04Despro09
Masukkan jenis buku: 1
Masukkan jumlah buku: 2
Total diskon: 2000.0
Total bayar: 80000.0
PS C:\Users\alifah\Documents\Udah\ALIDA\despro-j04despro>
```

## 2. Commit dan push kode program ke Github

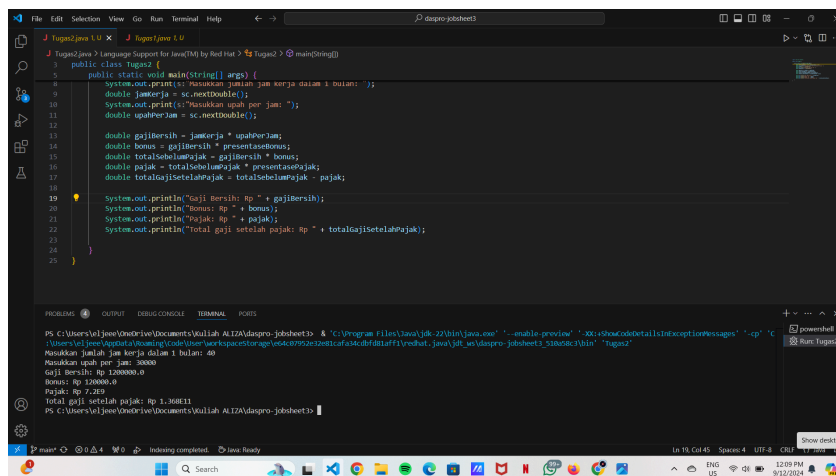


Input: Jumlah jam kerja karyawan per bulan dan upah per jam

Output: Gaji bersih setelah dipotong pajak

Algoritma:

1. Input jumlah jam kerja dan upah per jam
2. Hitung gaji pokok dari jumlah jam kerja dikalikan dengan upah per jam
3. Hitung bonus
4. Hitung total gaji sebelum pajak dengan menjumlahkan gaji pokok dengan bonus
5. Hitung pajak
6. Hitung gaji bersih dari total gaji sebelum terpotong dengan pajak-pajak
7. Dihasilkan gaji bersih yang diperoleh



```
1 public class Tugast2 {
2     public static void main(String[] args) {
3         System.out.println("Masukkan jumlah jam kerja dalam 1 bulan: ");
4         double jamKerja = sc.nextDouble();
5         System.out.print("Masukkan upah per jam: ");
6         double upahPerJam = sc.nextDouble();
7
8         double gajiBersih = jamKerja * upahPerJam;
9         double bonus = gajiBersih * 0.1;
10        double totalSebelumPajak = gajiBersih + bonus;
11        double pajak = totalSebelumPajak * 0.1;
12        double totalGajiSetelahPajak = totalSebelumPajak - pajak;
13
14        System.out.println("Gaji Bersih: Rp " + gajiBersih);
15        System.out.println("Bonus: Rp " + bonus);
16        System.out.println("Pajak: Rp " + pajak);
17        System.out.println("Total gaji setelah pajak: Rp " + totalGajiSetelahPajak);
18    }
19 }
20
21
22
23
24
25 }
```

PROBLEMA OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\eljee\OneDrive\Documents\Udah ALIZA\daspro-jobsheet> & "C:\Program Files\Java\jdk-22\bin\java.exe" "-enable-preview" "-XX:+ShowCodeDetailsInExceptionMessages" "-cp" "C:\Users\eljee\OneDrive\Documents\Udah ALIZA\daspro-jobsheet\src\Tugast2.jar" "Tugast2"
Masukkan jumlah jam kerja dalam 1 bulan: 40
Masukkan upah per jam: 30000
Gaji Bersih: Rp 1200000.0
Bonus: Rp 120000.0
Pajak: Rp 132000.0
Total gaji setelah pajak: Rp 1.068000
PS C:\Users\eljee\OneDrive\Documents\Udah ALIZA\daspro-jobsheet>
```