

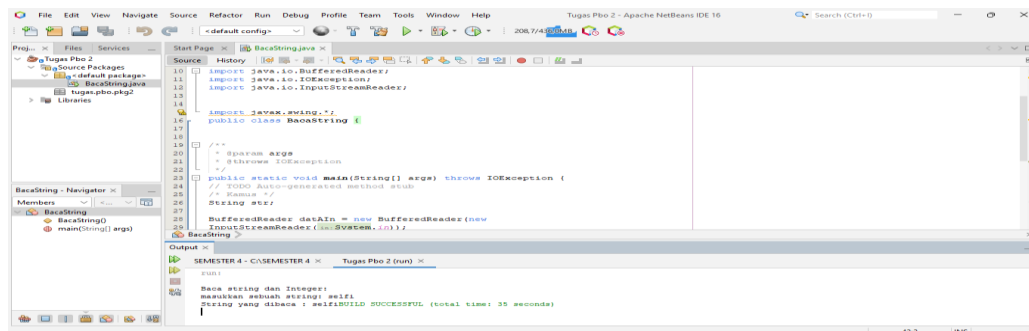
**PRAKTIKUM PEMROGRAMAN
BERIONTASI OBJEK**



Nama : Sarwana
Stambuk : 1302023017
Dosen : Mardiyah Hasnawi, S.Kom., M.T.

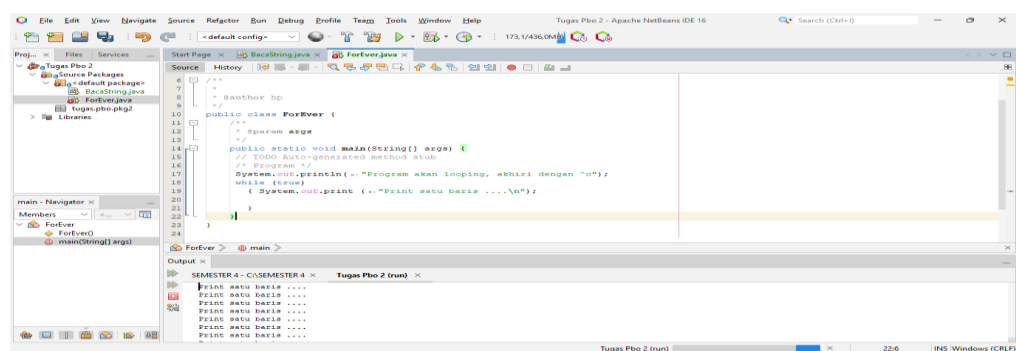
**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MUSLIM INDONESIA
MAKASSAR
2025**

1.



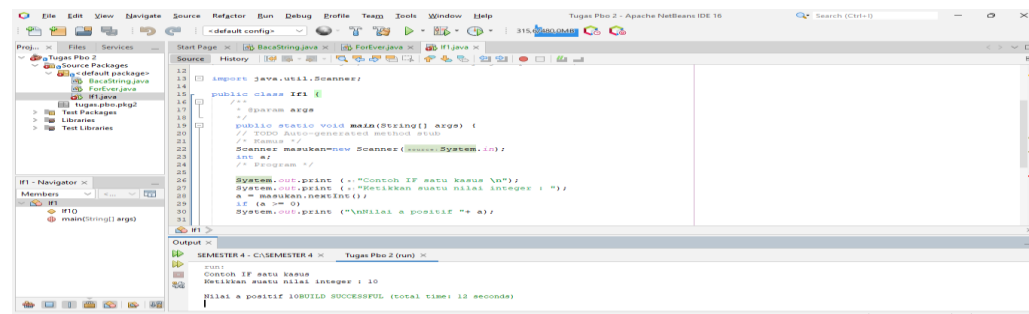
- Import : Mengimpor kelas untuk operasi input-output (BufferedReader, InputStreamReader)
- Kelas Bactracking: Mendeklarasikan kelas utama
- Metode main: Titik masuk eksekusi program
- Membaca Input: Menggunakan BufferedReader untuk membaca input dari pengguna, seperti semester dan nilai integer
- Menampilkan Output: Menghasilkan output yang menunjukkan semester, nilai, dan string hasil yang telah ditentukan.

2.



- Kelas Forevor: Mendeklarasikan kelas dengan nama Forever
- Metode main: Titik awal eksekusi program
- Output: Menggunakan System.out.println untuk mencetak string ke konsol. Dalam hal ini, program mencetak "Program akan looping, selalu hidup hingga -99"

3.



- Import Statement: Mengimpor kelas Scanner untuk membaca input dari pengguna
- Kelas RT: Mendeklarasikan kelas dengan nama RT
- Metode Main: Titik awal eksekusi program, di mana input diambil dari pengguna

- Output: Program mencetak pesan ke konsol dan hasil berdasarkan input yang diberikan

4.

```

10 // TODO Auto-generated method stub
11
12 public static void main(String[] args) {
13     // TODO Auto-generated method stub
14     Scanner masukan=new Scanner(System.in);
15     // Program
16
17     System.out.print ("Contoh IF dua kasus \n");
18     System.out.print ("Masukkan suatu nilai integer :");
19     masukan.nextInt();
20
21     if (a >= 0) {
22         System.out.println ("Nilai a positif "+ a);
23     }
24     else /* a < 0 */ {
25         System.out.println ("Nilai a negatif "+ a);
26     }
27 }

```

Output - Tugas Pbo 2 (run) x

```

Contoh IF dua kasus
Masukkan suatu nilai integer : 3
Nilai a positif 3
BUILD SUCCESSFUL (total time: 40 seconds)

```

- Import dan Kelas: Tidak ada import yang ditampilkan, tetapi program mendeklarasikan kelas (nama kelas tidak terlihat)
- Metode Main: Merupakan titik awal eksekusi program
- Membaca Input: Menggunakan Scanner untuk membaca input dari pengguna (tidak terlihat dalam gambar)
- Output: Menampilkan pesan yang menunjukkan apakah nilai tersebut positif atau tidak

5.

```

10 // TODO Auto-generated method stub
11
12 public static void main(String[] args) {
13     // TODO Auto-generated method stub
14     Scanner masukan=new Scanner(System.in);
15     // Program
16
17     System.out.print ("Contoh IF tiga kasus \n");
18     System.out.print ("Masukkan suatu nilai integer :");
19     masukan.nextInt();
20
21     if (a > 0) {
22         System.out.println ("Nilai a positif "+ a);
23     }
24     else if (a == 0) {
25         System.out.println ("Nilai Nol "+ a);
26     }
27     else /* a < 0 */ {
28         System.out.println ("Nilai a negatif "+ a);
29     }
30 }

```

Output - Tugas Pbo 2 (run) x

```

Contoh IF tiga kasus
Masukkan suatu nilai integer : 14
Nilai a positif 14
BUILD SUCCESSFUL (total time: 23 seconds)

```

- Input Pengguna: Program meminta pengguna untuk memasukkan sebuah angka bulat
- Pemeriksaan angka
 - Jika angka lebih besar dari nol, program akan menampilkan "Bilangan positif"
 - Jika angka kurang dari nol, program akan menampilkan "Bilangan negatif"
 - Jika angka sama dengan nol, tidak ada pesan yang ditampilkan

6.

```

10 // TODO Auto-generated method stub
11
12 public class KasusBoolean {
13     // TODO Auto-generated method stub
14     boolean bool;
15
16     // Algoritma
17     bool = true;
18
19     if (bool) {
20         System.out.print ("true\n");
21     }
22     else {
23         System.out.print ("false\n");
24     }
25     if (!bool) {
26         System.out.print ("salah\n");
27     }
28     else {
29         System.out.print ("benar\n");
30     }
31 }

```

Output - Tugas Pbo 2 (run) x

```

true
false
BUILD SUCCESSFUL (total time: 1 second)

```

- Deklarasi Kelas: Program ini berada dalam kelas bernama KasusBoolean
- Variabel Boolean: Terdapat variabel bool yang diset ke true
- Pernyataan IF: Program memeriksa nilai dari bool
 - Jika bool bernilai true, maka program mencetak "true" dan "benar"
 - Tidak ada tindakan jika bool bernilai false
- Output: Hasil yang ditampilkan adalah "true" dan "benar" karena bool diset ke true

7.

```

16 char c;
17 Scanner masukan=new Scanner(System.in);
18 // Program //
19 System.out.println ("Ketikkan sebuah huruf, akhiri dengan RETURN \n");
20 c=masukan.next().charAt(0);
21 switch (c) {
22     case 'a': {
23         System.out.println (": " Yang anda ketik adalah a \n");
24         break;
25     }
26     case 'u': {
27         System.out.println (": " Yang anda ketik adalah u\n");
28         break;
29     }
30     case 'e': {
31         System.out.println (": " Yang anda ketik adalah e\n");
32         break;
33     }
34     case 'i': {
35         System.out.println (": " Yang anda ketik adalah i\n");
36         break;
37     }
38     default: {
39         System.out.println (": " Yang anda ketik adalah lain\n");
40     }
41 }
42 }
  
```

Output - Tugas Pbo 2 (run) >

```

run:
Ketikkan sebuah huruf, akhiri dengan RETURN
a
Yang anda ketik adalah a
BUILD SUCCESSFUL (total time: 1 minute 4 seconds)
  
```

- Input Pengguna: Program meminta pengguna untuk memasukkan sebuah huruf
- Scanner: digunakan untuk membaca input dari pengguna melalui konsol
- Pernyataan Output: Program mengeluarkan pesan yang meminta pengguna untuk memasukkan huruf, diakhiri dengan kata "RETURN"
- Output: Hasil yang ditampilkan adalah instruksi untuk memasukkan huruf

8.

```

7 //
8 // @author hp
9 //
10 public class Konstant {
11
12     public static void main(String[] args) {
13         // TODO Auto-generated method stub
14         // Kasus //
15         final float PHI = 3.1415f;
16         float r;
17         Scanner masukan=new Scanner(System.in);
18         // program // // baca data //
19
20         System.out.println (": "Jari-jari lingkaran =");
21         r = masukan.nextFloat();
22         // Hitung dan tulis hasil //
23         System.out.println ("Luas lingkaran = "+ (PHI * r * r)+"\n");
24         System.out.println (": "Akhir program \n");
25     }
26 }
  
```

Output - Tugas Pbo 2 (run) >

```

run:
Jari-jari lingkaran =15
Luas lingkaran = 706.8375
Akhir program
BUILD SUCCESSFUL (total time: 7 seconds)
  
```

- Deklarasi Kelas: Program ini berada dalam kelas bernama Konstant
- Konstanta: Dideklarasikan konstanta PHI yang bernilai 3.14157, digunakan untuk perhitungan lingkaran
- Penghitungan: Program mencetak nilai jari-jari dan menghitung panjang lingkaran menggunakan rumus
- Output: Program mencetak hasil jari-jari dan panjang lingkaran

9.

```

11 public class Max2 {
12
13     public static void main(String[] args) {
14         // TODO Auto-generated method stub
15         // Kasus //
16         int a, b;
17         Scanner masukan=new Scanner(System.in);
18
19         System.out.println (": "Maksimum dua bilangan : \n");
20         System.out.println (": "Ketikkan dua bilangan, pisahkan dg RETURN \n");
21         a=masukan.nextInt();
22         b=masukan.nextInt();
23         System.out.println ("Ke dua bilangan : a = "+ a+ " b = "+b);
24         if (a > b) {
25             System.out.println ("Nilai a yang maksimum "+ a);
26         } else {
27             System.out.println ("Nilai b yang maksimum "+ b);
28         }
29     }
30 }
  
```

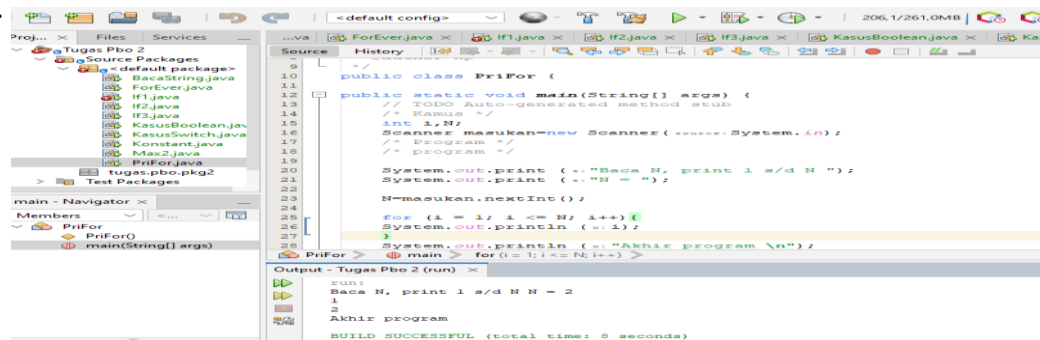
Output - Tugas Pbo 2 (run) >

```

run:
Maksimum dua bilangan :
Ketikkan dua bilangan, pisahkan dg RETURN :
5
10
Ke dua bilangan : a = 5 b = 10
Nilai a yang maksimum 10
BUILD SUCCESSFUL (total time: 2 minutes 19 seconds)
  
```

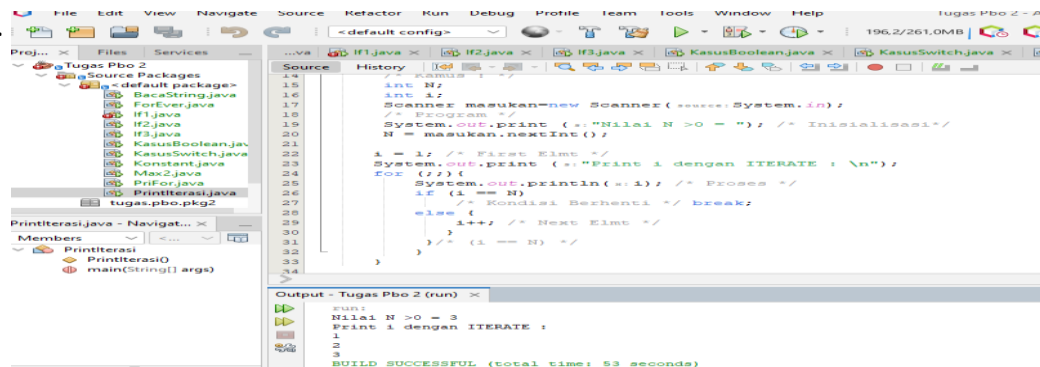
- Input Pengguna: Program meminta pengguna untuk memasukkan dua bilangan
- Pernyataan IF: Program membandingkan kedua bilangan:
 - Jika bilangan pertama lebih besar, program mencetak "Bilangan 1 adalah yang maksimum"
 - Jika bilangan kedua lebih besar, program mencetak "Bilangan 2 adalah yang maksimum"
 - Jika kedua bilangan sama, program mencetak "Kedua bilangan sama"
- Output: Menampilkan hasil perbandingan berdasarkan input yang diberikan oleh pengguna

10.



- Input Pengguna: Program meminta pengguna untuk memasukkan nilai n
- Looping: Menggunakan loop for untuk mencetak angka dari 1 hingga n:
 - Loop dimulai dari i = 1 hingga i <= n
- Setelah mencetak angka, program menampilkan pesan "Akhir program"

11.



- Input Pengguna: Program meminta pengguna untuk memasukkan nilai m
- Looping: Menggunakan loop (yang tidak sepenuhnya terlihat) untuk menjalankan iterasi berdasarkan nilai m
- Pernyataan IF: Program memeriksa kondisi tertentu (tidak terlihat sepenuhnya) dan dapat menggunakan pernyataan break untuk keluar dari loop jika kondisi terpenuhi
- Output: Menampilkan pesan "Print 1 dengan ITERATE" sebelum melakukan iterasi

12.

```

10 public class PrintRepeat {
11     public static void main(String[] args) {
12         int N;
13         int i;
14         Scanner masukan=new Scanner(System.in);
15         System.out.print (".Nilai N >0 = ");
16         N = masukan.nextInt();
17         i = 1;
18         System.out.print (".Print i dengan REPEAT: \n");
19         do{
20             System.out.print (i+"\n"); /* Proses */
21             i++;
22         }while(i <= N);
23     }
24 }

```

Output:

```

Tugas Pbo 2 (run)  Tugas Pbo 2 (run) #2
1
1
1
1
1
1

```

- Deklarasi Kelas: Program ini berada dalam kelas bernama PrintRepeat
- Input Pengguna: Program meminta pengguna untuk memasukkan nilai N
- Inisialisasi: Variabel i diinisialisasi dengan nilai 1
- Looping: Menggunakan loop do-while untuk mencetak angka dari 1 hingga N -Setelah mencetak, nilai i akan ditingkatkan
- Output: Program mencetak nilai i hingga mencapai batas yang ditentukan oleh pengguna

13.

```

10 public class PrintWhile {
11     public static void main(String[] args) {
12         int N;
13         int i;
14         Scanner masukan=new Scanner(System.in);
15         System.out.print (".Nilai N >0 = "); /* Inisialisasi */
16         N = masukan.nextInt();
17         i = 1; /* First Elmt */
18         System.out.print (".Print i dengan WHILE: \n");
19         while (i <= N) /* Kondisi pengulangan */
20         {
21             System.out.println (i); /* Proses */
22             i++; /* Next Elmt */
23         }
24         /* (i > N) */
25     }
26 }

```

Output:

```

Tugas Pbo 2 (run)  Tugas Pbo 2 (run) #2  Tugas Pbo 2 (run) #3
RUN:
Nilai N >0 = 4
Print i dengan WHILE:
1
2
3
4
BUILD SUCCESSFUL (total time: 9 seconds)

```

- Inisialisasi: Variabel i diinisialisasi dengan nilai 1
- Looping: Menggunakan loop while untuk mencetak angka dari 1 hingga N -Loop akan terus berjalan selama kondisi tertentu terpenuhi (misalnya, i kurang dari atau sama dengan N)
- Output: Program mencetak nilai i dan kemungkinan menginformasikan pengguna tentang proses

14.

```

7 //
8 // Author hp
9 //
10 public class PrintWhile1 {
11     public static void main(String[] args) {
12         int N;
13         int i = 1;
14         Scanner masukan=new Scanner(System.in);
15         System.out.print (".Nilai N >0 = ");
16         N = masukan.nextInt();
17         System.out.print (".Print i dengan WHILE (ringkas): \n");
18         while (i <= N) {
19             System.out.println (i);
20             i++;
21         }
22         /* (i > N) */
23     }
24 }

```

Output:

```

Tugas Pbo 2 (run)  Tugas Pbo 2 (run) #2  Tugas Pbo 2 (run) #3
RUN:
Nilai N >0 = 5
Print i dengan WHILE (ringkas):
1
2
3
4
5
BUILD SUCCESSFUL (total time: 13 seconds)

```

- Input Pengguna: Program meminta pengguna untuk memasukkan nilai N (meskipun tidak terlihat secara langsung dalam gambar)
- Looping: Menggunakan loop while untuk mencetak angka:
 - Program akan mencetak "Print 1 dengan WHILE" dan kemudian melanjutkan hingga kondisi tertentu tidak terpenuhi lagi.
- Output: Program mencetak hasil dan menunjukkan bahwa loop berfungsi sesuai harapan

15.

```

12 public static void main(String[] args) {
13     int Sum=0;
14     int x;
15     Scanner masukan=new Scanner(System.in);
16     System.out.print("Masukkan nilai x (int), akhiri dg 999 :");
17     x = masukan.nextInt(); /* First Elmt */
18     if (x == 999){
19         System.out.print("Kasus kosong \n");
20     }else { /* Minimal ada satu data yang dijumlahkan
21         Sum = x; /* Inisialisasi Inisialisasi
22         for (i=1; i<=x; i++){
23             System.out.print("Masukkan nilai x (int),akhiri dg 999 :");
24             x = masukan.nextInt(); /* Next Elmt */
25             if(x==999)
26                 break;
27             Sum = Sum + x; /* Proses */
28         }
29     }
30 }
  
```

Output:

```

Tugas Pbo 2 (run) x3
Sum:
Masukkan nilai x (int), akhiri dg 999 :1
Masukkan nilai x (int),akhiri dg 999 :20
Masukkan nilai x (int),akhiri dg 999 :999
Hasil penjumlahan = 20
BUILD SUCCESSFUL (total time: 1 minute 17 seconds)
  
```

- Deklarasi Kelas: Program ini berada dalam kelas bernama PrintIterasi
- Input Pengguna: Program meminta pengguna untuk memasukkan angka, dengan instruksi bahwa pengguna harus memasukkan -999 untuk menghentikan input
- Inisialisasi: Variabel sum diinisialisasi untuk menyimpan total jumlah
- Looping: Program menggunakan loop untuk terus meminta input hingga pengguna memasukkan -999
 - Angka yang dimasukkan akan ditambahkan ke dalam sum.
- Output: Setelah pengguna selesai memasukkan angka, program akan mencetak total jumlah yang telah dihitung

16.

```

12 public static void main(String[] args) {
13     int Sum;
14     int x;
15     Scanner masukan=new Scanner(System.in);
16     System.out.print("Masukkan nilai x (int), akhiri dg 999 : ");
17     x = masukan.nextInt(); /* First Elmt */
18     if (x == 999){
19         System.out.print("Kasus kosong \n");
20     }else {
21         Sum = 0;
22         do {
23             Sum = Sum + x; /* Proses */
24             System.out.print("Masukkan nilai x (int), akhiri dg 999 : ");
25             x = masukan.nextInt(); /* Next Elmt */
26             while (x != 999); /* Kondisi pengulangan */
27         } while (x != 999);
28     }
29     System.out.println("Hasil penjumlahan = "+Sum);
30 }
  
```

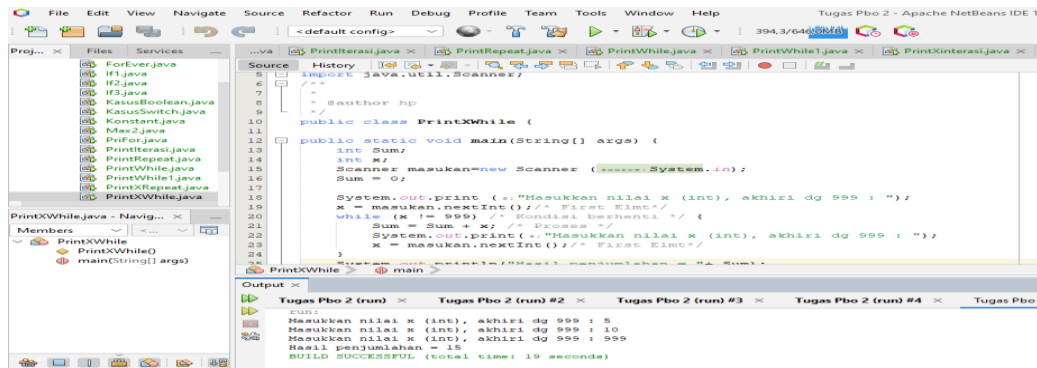
Output:

```

Tugas Pbo 2 (run) x3
Sum:
Masukkan nilai x (int), akhiri dg 999 : 10
Masukkan nilai x (int), akhiri dg 999 : 20
Masukkan nilai x (int), akhiri dg 999 : 999
Hasil penjumlahan = 30
BUILD SUCCESSFUL (total time: 14 seconds)
  
```

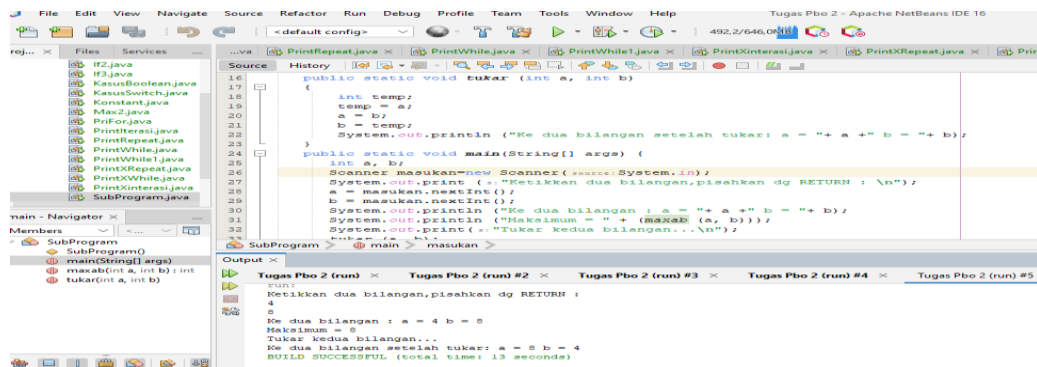
- Output: Setelah pengguna selesai memasukkan angka, program mencetak hasil jumlah total yang telah dihitung
- Deklarasi Kelas: Program ini berada dalam kelas bernama PrintRepeat.
- Input Pengguna: Program meminta pengguna untuk memasukkan angka, dengan instruksi bahwa pengguna harus memasukkan -999 untuk menghentikan input

17.



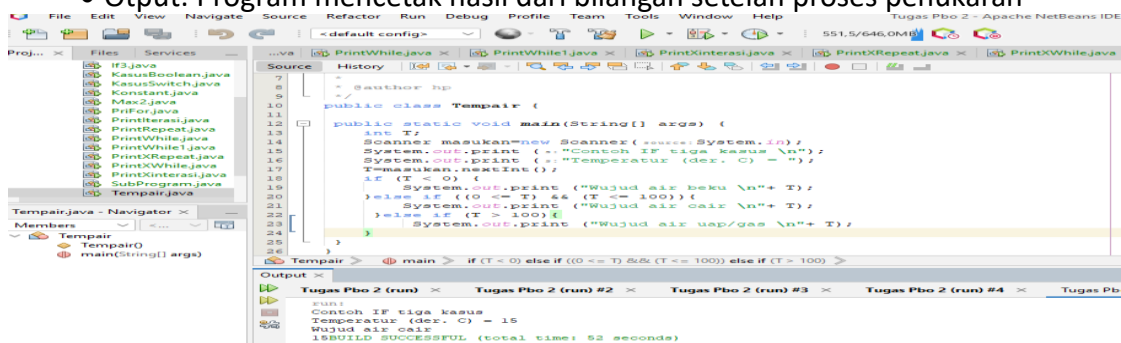
- Input Pengguna: Program meminta pengguna untuk memasukkan angka, dengan instruksi untuk mengakhiri input dengan -999
- Looping: Program menggunakan loop untuk meminta input terus menerus hingga pengguna memasukkan -999
 - Setiap angka yang dimasukkan ditambahkan ke dalam sum
- Output: Setelah selesai, program mencetak total jumlah yang telah dihitung

18.



- Deklarasi Kelas: Program ini berada dalam kelas yang tidak disebutkan namanya dalam gambar, tetapi berfungsi untuk menukar dua bilangan
- Deklarasi Variable: Variabel temp, a, dan b dideklarasikan untuk menyimpan nilai yang akan diproses
- Input Pengguna: Program meminta pengguna untuk memasukkan dua bilangan
- Proses Tukar: Menggunakan variabel sementara (temp) untuk menukar nilai antara a dan b
- Otput: Program mencetak hasil dari bilangan setelah proses penukaran

19.



- Deklarasi Kelas: Program ini berada dalam kelas bernama Tempar
- Input Pengguna: Program menggunakan Scanner untuk meminta pengguna memasukkan pilihan menu
- Output: Program menampilkan hasil berdasarkan pilihan yang dimasukkan oleh pengguna
- Menu Opsi: Terdapat opsi yang ditampilkan kepada pengguna untuk memilih, misalnya:
 - Menampilkan suhu dalam satuan tertentu (kemungkinan Celsius atau Fahrenheit)
 - Menampilkan pilihan lain yang berkaitan dengan suhu.

