

Tugas 2

Makassar, 07 Maret 2025

**PRAKTIKUM
PEMROGRAMAN
BERORIENTASI OBJEK**



Nama : Aqilah Jamellya Wulandari Putrisyah

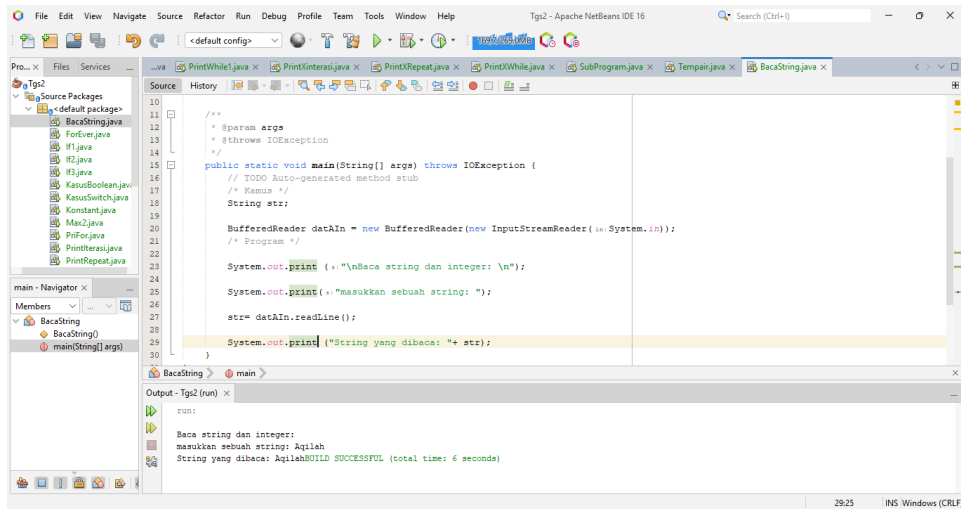
Nim : 13020230155

Frekuensi : TI_PBO-12(B2)

Dosen : Mardiyah Hasnawi, S.Kom., M.T.

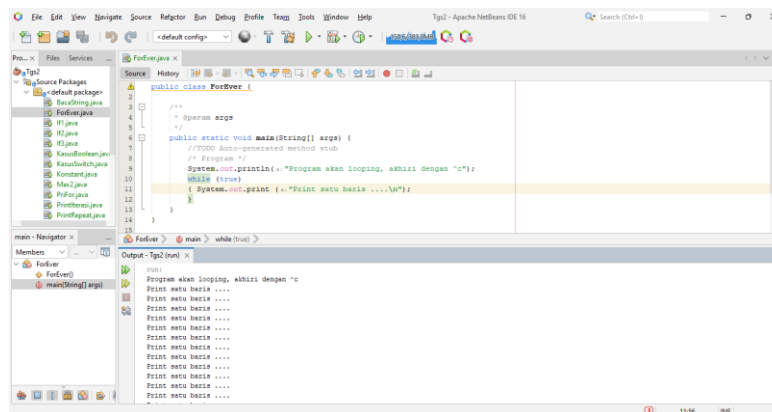
**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS MUSLIM INDONESIA
MAKASSAR
2025**

1.



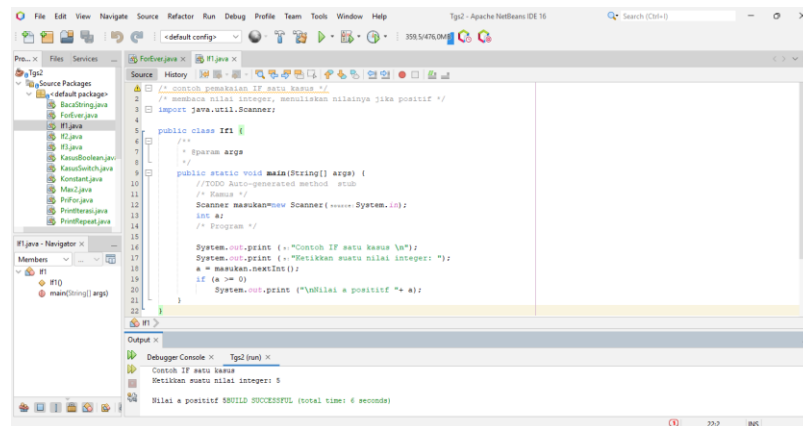
Penjelasannya: Program ini dirancang untuk membaca input string dari User dan menampilkannya kembali. Pertama-tama, program mengimpor library yang diperlukan untuk operasi input/output, terutama `BufferedReader` yang memungkinkan pembacaan input dari standard input. Di dalam metode `main`, program mendeklarasikan variabel `str` untuk menyimpan string input. Kemudian, objek `BufferedReader` dibuat untuk membaca input dari User. Program menampilkan pesan "masukkan sebuah string: " ke layar, meminta user untuk memasukkan string. Setelah pengguna memasukkan string dan menekan Enter, program membaca string tersebut menggunakan `dataIn.readLine()` dan menyimpannya di variabel `str`. Terakhir, program menampilkan pesan "String yang dibaca: " diikuti dengan nilai variabel `str`, yaitu string yang dimasukkan oleh user, ke layar. Program ini berfungsi untuk mengambil input string dari user dan menampilkannya kembali.

2.



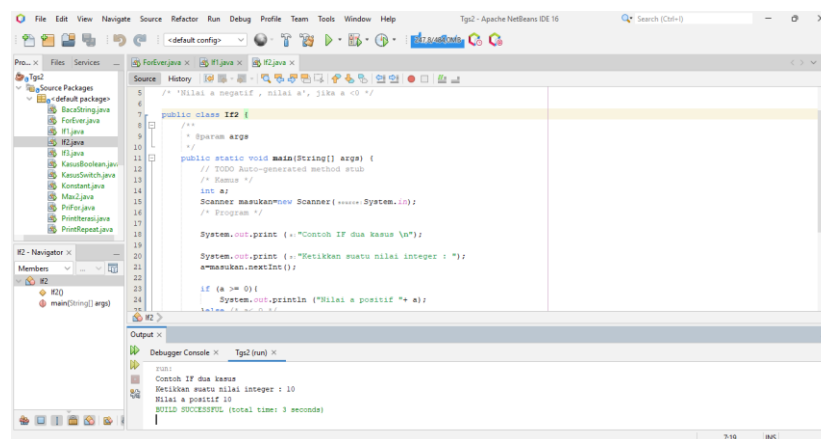
Penjelasannya: Program ini akan terus menerus mencetak teks "Print satu baris ..." ke layar tanpa berhenti. Program memiliki perulangan while (true) yang nanti akan terus berjalan sampai dihentikan secara paksa dengan menekan Ctrl+C atau ^c. Setiap kali perulangan berjalan, program mencetak teks tersebut ke layar dan pindah ke baris baru.

3.



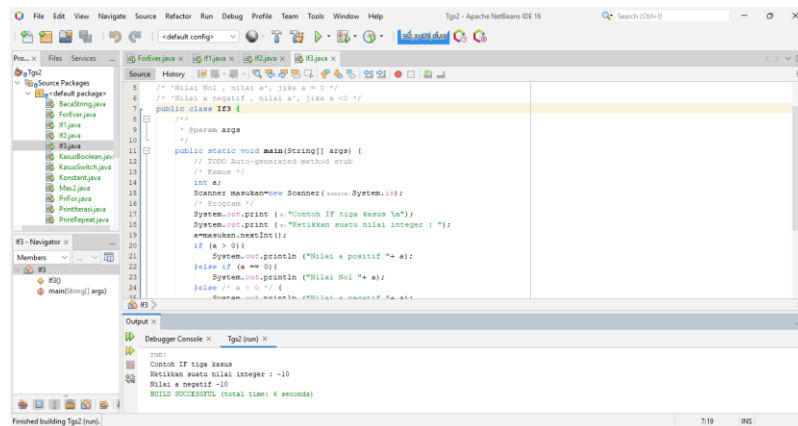
Penjelasannya: Program ini meminta user untuk memasukkan sebuah bilangan bulat. Jika bilangan tersebut positif atau nol, program akan menampilkan bilangan tersebut beserta pesan "Nilai a positif". Jika bilangan tersebut negatif, program tidak akan menampilkan apa pun.

4.



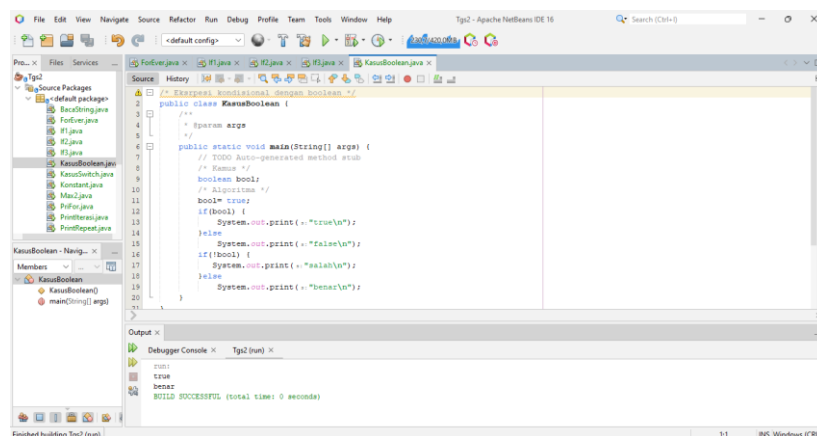
Penjelasannya: Program ini meminta user untuk memasukkan sebuah bilangan bulat. Program kemudian akan memeriksa apakah bilangan tersebut positif atau negatif. Jika positif atau nol, program akan menampilkan bilangan tersebut beserta pesan "Nilai a positif". Jika negatif, program akan menampilkan bilangan tersebut beserta pesan "Nilai a negatif".

5.



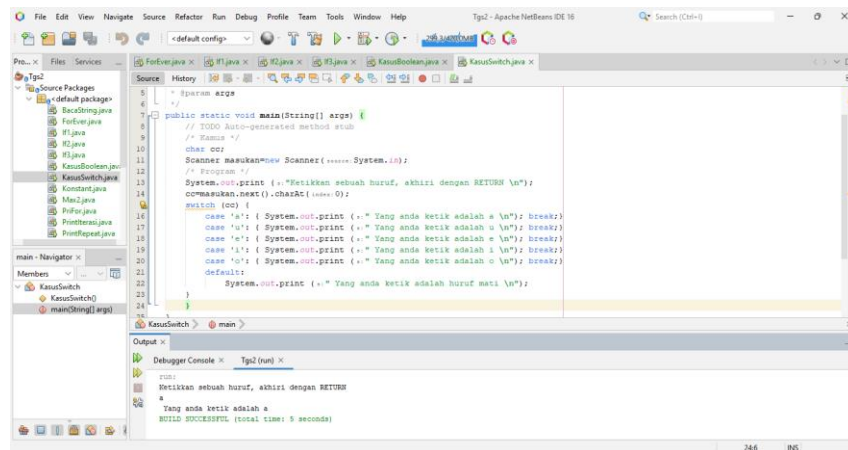
Penjelasannya: Program ini meminta user untuk memasukkan sebuah bilangan bulat. Program kemudian memeriksa bilangan tersebut dan menampilkan pesan yang sesuai berdasarkan nilai bilangan tersebut, Jika bilangan tersebut positif, program akan menampilkan pesan "Nilai a positif" diikuti dengan nilai bilangan tersebut. Namun, Jika bilangan tersebut nol, program akan menampilkan pesan "Nilai Nol" diikuti dengan nilai bilangan tersebut dan Jika bilangan tersebut negatif, program akan menampilkan pesan "Nilai a negatif" diikuti dengan nilai bilangan tersebut. Program ini mengklasifikasi bilangan bulat yang dimasukkan pengguna ke dalam tiga kategori: positif, nol, atau negatif, dan menampilkan pesan yang sesuai kategorinya.

6.



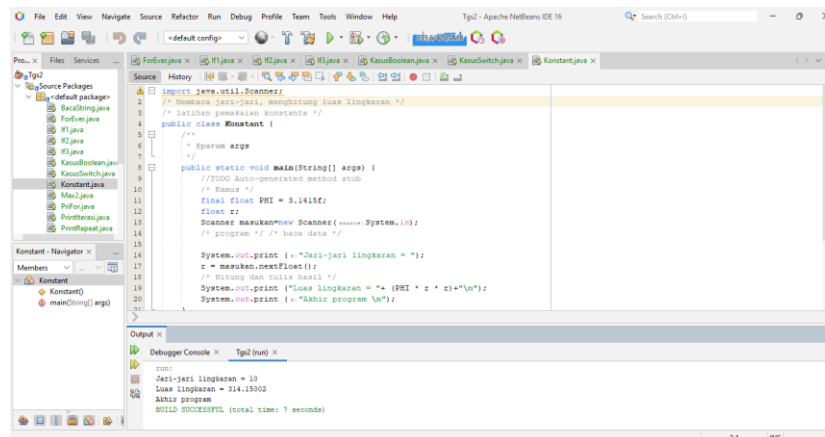
Penjelasannya: Program ini mendemonstrasikan penggunaan variabel boolean dan ekspresi kondisional if. Pertama-tama, program akan mendeklarasikan variabel boolean bernama bool dan menginisialisasinya dengan nilai true. Kemudian, program memiliki dua blok if-else: Blok pertama, Program memeriksa apakah bool bernilai true. Karena bool memang true, program mencetak "true" ke layar. Jika bool bernilai false, program akan mencetak "false". Blok kedua, Program memeriksa apakah bool bernilai false dengan menggunakan operator ! (not). Karena bool bernilai true, maka !bool bernilai false. Oleh karena itu, bagian else dieksekusi, dan program mencetak "benar" ke layar. Jika !bool bernilai true, program akan mencetak "salah".

7.



Penjelasannya: Program ini akan meminta user untuk memasukkan sebuah karakter (huruf). Kemudian program akan membaca karakter pertama yang dimasukkan user dan menggunakan struktur switch untuk menentukan tindakan yang akan diambil. Jika karakter yang dimasukkan adalah salah satu dari huruf vokal ('a', 'u', 'e', 'i', 'o'), program akan menampilkan pesan yang sesuai, misalnya "Yang anda ketik adalah a". Jika karakter yang dimasukkan bukan huruf vokal, program akan menampilkan pesan "Yang anda ketik adalah huruf mati". Dengan kata lain, program ini mengklasifikasikan karakter yang dimasukkan pengguna sebagai huruf vokal atau huruf mati dan menampilkannya.

8.



Penjelasannya: Program ini untuk menghitung luas lingkaran. Pertama, program akan mendeklarasikan konstanta PHI dengan nilai 3.1415f. Kemudian, program akan meminta user untuk memasukkan jari-jari lingkaran. Setelah pengguna memasukkan jari-jari, program menghitung luas lingkaran menggunakan rumus $\text{PHI} \times r \times r$ dan menampilkan hasilnya ke layar. Terakhir, program akan menampilkan pesan "Akhir program".

9.


```

1 public static void main(String[] args) {
2     // TODO Auto-generated method stub
3     /* Nama : */
4     int N;
5     int i;
6     Scanner masukan = new Scanner(System.in);
7     /* Program */
8     System.out.print("Nilai N > 0 = "); /* Inisialisasi */
9     N = masukan.nextInt();
10
11     i = 1; /* First Elmt */
12     System.out.print("Print i dengan ITERATE : \n");
13     for (i = 1; i <= N; i++) { /* Proses */
14         System.out.println(i); /* Break */
15         i++; /* Next Elmt */
16     } /* (i == N) */
17 }

```

Output Console:

```

1 Nilai N > 0 = 4
2 Print i dengan ITERATE :
3 1
4 2
5 3
6 4

```

Penjelasannya: Program ini meminta user untuk memasukkan sebuah bilangan bulat N. Kemudian, program akan mencetak angka dari 1 hingga N ke layar menggunakan perulangan for. Program menginisialisasi variabel i dengan nilai 1. Di dalam perulangan, program mencetak nilai i ke layar dan kemudian menambahkan 1 ke nilai i. Perulangan akan terus berlanjut hingga i sama dengan N. Setelah perulangan selesai, program akan berhenti.

12.

```

1 public static void main(String[] args) {
2     // TODO Auto-generated method stub
3     /* Nama : */
4     int N;
5     int i;
6     Scanner masukan = new Scanner(System.in);
7     /* Program */
8     System.out.print("Nilai N > 0 = ");
9     /* Inisialisasi */
10    N = masukan.nextInt();
11
12    i = 1; /* First Elmt */
13    System.out.print("Print i dengan REPEAT : \n");
14    do {
15        System.out.println(i); /* Proses */
16        i++; /* Next Elmt */
17    } while (i <= N);
18 }

```

Output Console:

```

1 Nilai N > 0 = 3
2 Print i dengan REPEAT:
3 1
4 2
5 3

```

Penjelasannya: Program ini akan meminta pengguna untuk memasukkan sebuah bilangan bulat N. Kemudian, program akan menggunakan perulangan do-while untuk mencetak angka dari 1 hingga N ke layar. Program menginisialisasi variabel i dengan nilai 1. Di dalam perulangan, program mencetak nilai i ke layar dan kemudian menambahkan 1 ke nilai i. Perulangan akan terus berlanjut hingga i lebih besar dari N. Setelah perulangan selesai, program akan berhenti.

13.

The screenshot shows the Eclipse IDE with a Java project named 'Tipe2'. The source editor displays a file named 'PrintHello.java' with the following code:

```
1 // Tipe2.java
2
3 // TODO Auto-generated method stub
4 // Penulis :
5
6 public static void main(String[] args) {
7
8     // TODO Auto-generated method stub
9     // Penulis :
10
11     int N;
12     int i;
13     Scanner masukan = new Scanner(System.in);
14     // Program
15
16     System.out.print ("Silakan N > 0 = "); // Input
17     N = masukan.nextInt();
18     i = 1; // Print Elmt
19
20     System.out.print ("Print 1 dengan WHILE: \n");
21     while (i <= N) { // Print dengan WHILE
22         System.out.println (i); // Print
23         i++; // Next Elmt
24     } // (i > N)
25 }
```

The Output console shows the execution results:

```
PrintHello
main
Silakan N > 0 = 5
Print 1 dengan WHILE:
1
2
3
4
5
```

Penjelasannya: Program ini akan meminta user untuk memasukkan sebuah bilangan bulat N. Kemudian, program akan menggunakan perulangan while untuk mencetak angka dari 1 hingga N ke layar. Program akan menginisialisasi variabel i dengan nilai 1. Di dalam perulangan, program akan mencetak nilai i ke layar, kemudian menambahkan 1 ke nilai i. Perulangan akan terus berlanjut hingga i lebih besar dari N. Setelah perulangan selesai, program akan berhenti.

14.

The screenshot shows the Eclipse IDE with a Java project named 'Tipe2'. The source editor displays a file named 'PrintWhile.java' with the following code:

```
1 // Tipe2.java
2
3 // TODO Auto-generated method stub
4 // Penulis :
5
6 public class PrintWhile {
7
8     // TODO Auto-generated method stub
9     // Penulis :
10
11     int N;
12     int i = 1;
13     Scanner masukan = new Scanner(System.in);
14     // Program
15
16     System.out.print ("Silakan N > 0 = ");
17     N = masukan.nextInt();
18     System.out.print ("Print 1 dengan WHILE (ringkas): \n");
19     while (i <= N) {
20         System.out.println (i);
21     } // (i > N)
22 }
```

The Output console shows the execution results:

```
PrintWhile
main
Silakan N > 0 = 5
Print 1 dengan WHILE (ringkas):
1
2
3
4
5
BUILD SUCCESSFUL (total time: 12 seconds)
```

Penjelasannya: Sama seperti program sebelumnya, program ini akan meminta user untuk memasukkan sebuah bilangan bulat N. Kemudian, program akan menggunakan perulangan while untuk mencetak angka dari 1 hingga N ke layar. Program akan menginisialisasi variabel i dengan nilai 1. Di dalam perulangan, program mencetak nilai i ke layar dan kemudian menambahkan 1 ke nilai i. Perulangan akan terus berlanjut hingga i lebih besar dari N. Setelah perulangan selesai, program akan berhenti.

15.


```

13  int x;
14  Scanner masukan=new Scanner(System.in);
15  /* Program */
16
17  System.out.print ("Masukkan nilai x (inti), akhiri dg 999 : ");
18
19  x = masukan.nextInt(); /* First Elmt */
20
21  if (x == 999) {
22      System.out.print ("\nKasus kosong \n");
23      return; /* Minimal ada satu data yang dijumlahkan */
24  }
25  Sum = 0; /* Inisialisasi inisialisasi */
26
27  do {
28      Sum = Sum + x; /* Proses */
29      System.out.print ("Masukkan nilai x (inti), akhiri dg 999 : ");
30      x = masukan.nextInt(); /* Next Elmt */
31  } while (x != 999); /* Kondisi pengulangan */
32  System.out.println ("Hasil penjumlahan = "+Sum);
33  /* Terminasi */

```

Output - Tpg2 (run) #2

```

run:
Masukkan nilai x (inti), akhiri dg 999 : 2
Masukkan nilai x (inti), akhiri dg 999 : 5
Masukkan nilai x (inti), akhiri dg 999 : 4 999
Masukkan nilai x (inti), akhiri dg 999 : Hasil penjumlahan = 11
BUILD SUCCESSFUL (total time: 0 minutes 0 seconds)

```

Penjelasannya: Program ini akan meminta user untuk memasukkan serangkaian bilangan bulat. Kemudian, program akan menjumlahkan semua bilangan bulat yang dimasukkan oleh user hingga user memasukkan angka 999. Jika user langsung memasukkan 999 di awal, program akan menampilkan pesan "Kasus kosong". Jika tidak, program akan terus meminta input bilangan bulat dan menjumlahkannya hingga user memasukkan 999. Setelah user memasukkan 999, program akan menampilkan hasil penjumlahan semua bilangan bulat yang dimasukkan.

16.

```

4  public class PrintWhile {
5      /**
6       * @param args
7       */
8
9      public static void main(String[] args) {
10         // TODO Auto-generated method stub
11         // Fungsi : */
12         int Sum;
13         int x;
14         Scanner masukan=new Scanner(System.in);
15         /* Program */
16
17         Sum = 0; /* Inisialisasi */
18
19         System.out.print ("Masukkan nilai x (inti), akhiri dg 999 : ");
20         x = masukan.nextInt(); /* First Elmt */
21         while (x != 999) /* Kondisi berhenti */ {
22             Sum = Sum + x; /* Proses */
23             System.out.print ("Masukkan nilai x (inti), akhiri dg 999 : ");
24             x = masukan.nextInt(); /* Next Elmt */
25         }
26         System.out.println ("Hasil penjumlahan = "+ Sum); /* Terminasi */
27     }
28 }

```

Output - Tpg2 (run) #3

```

run:
Masukkan nilai x (inti), akhiri dg 999 : 3
Masukkan nilai x (inti), akhiri dg 999 : 5 999
Masukkan nilai x (inti), akhiri dg 999 : Hasil penjumlahan = 8
BUILD SUCCESSFUL (total time: 0 minutes 0 seconds)

```

Penjelasannya: Program ini akan menjumlahkan serangkaian bilangan bulat yang dimasukkan oleh user. Program akan terus meminta input bilangan bulat dan menambahkannya ke total jumlah hingga user memasukkan angka 999. Setelah pengguna memasukkan 999, program akan berhenti meminta input dan menampilkan hasil penjumlahan semua bilangan bulat yang dimasukkan pengguna.

17.

```

1  //
2  // Param args
3  //
4  public static void main(String[] args) {
5      // TODO Auto-generated method stub
6      /* Kasus : */
7      int Sum=0;
8      int x;
9      Scanner masukan=new Scanner(System.in);
10     /* Program */
11     System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");
12     x = masukan.nextInt(); /* First Elmt */
13     if (x == 999){
14         System.out.print ("Kasus Kosong \n");
15     }else { /* Minimal ada satu data yang dijumlahkan */
16         Sum = x; /* Inisialisasi invariant */
17         for (;;) {
18             System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");
19             x = masukan.nextInt(); /* Next Elmt */
20             if(x==999)
21                 break;
22             else{
23                 Sum = Sum + x; /* Success */
24             }
25         }
26     }
27 }
28
29 Output - Tpg2 (run) #3
30
31 run
32 Masukan nilai x (int), akhiri dg 999 : 10
33 Masukan nilai x (int), akhiri dg 999 : 100
34 Masukan nilai x (int), akhiri dg 999 : 999 Hasil penjumlahan = 11
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Penjelasannya: Program ini akan menjumlahkan serangkaian bilangan bulat yang dimasukkan oleh user. Program akan terus membaca bilangan bulat hingga user memasukkan angka 999. Jika user langsung memasukkan 999 di awal, program akan menampilkan pesan "Kasus kosong". Jika tidak, program akan menjumlahkan setiap bilangan bulat yang dimasukkan user. Setelah user memasukkan 999, program akan menampilkan hasil penjumlahan semua bilangan bulat yang dimasukkan.

18.

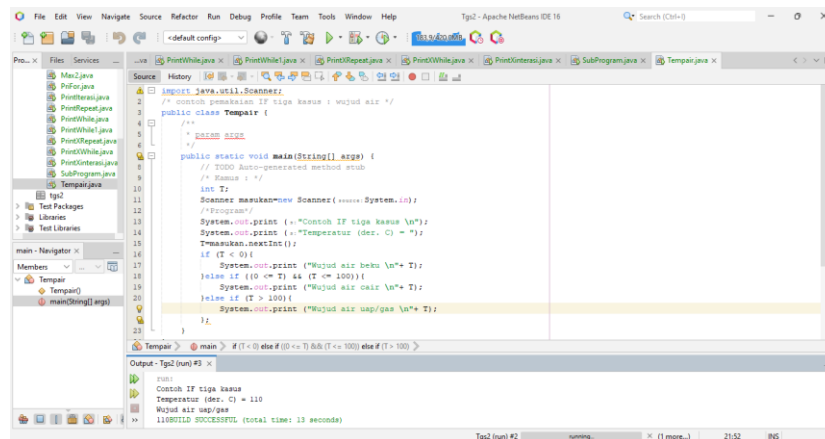
```

1  //
2  // Param args
3  //
4  public static void main(String[] args) {
5      // TODO Auto-generated method stub
6      /* Kasus : */
7      int Sum=0;
8      int x;
9      Scanner masukan=new Scanner(System.in);
10     /* Program */
11     System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");
12     x = masukan.nextInt(); /* First Elmt */
13     if (x == 999){
14         System.out.print ("Kasus Kosong \n");
15     }else { /* Minimal ada satu data yang dijumlahkan */
16         Sum = x; /* Inisialisasi invariant */
17         for (;;) {
18             System.out.print ("Masukkan nilai x (int), akhiri dg 999 : ");
19             x = masukan.nextInt(); /* Next Elmt */
20             if(x==999)
21                 break;
22             else{
23                 Sum = Sum + x; /* Success */
24             }
25         }
26     }
27 }
28
29 Output - Tpg2 (run) #3
30
31 run
32 Masukan nilai x (int), akhiri dg 999 : 10
33 Masukan nilai x (int), akhiri dg 999 : 100
34 Masukan nilai x (int), akhiri dg 999 : 999 Hasil penjumlahan = 11
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

Penjelasannya: Program ini berisi dua fungsi yaitu, maxab yang mengembalikan nilai maksimum dari dua bilangan bulat, dan tukar yang menukar nilai dua bilangan bulat. Program utama akan meminta user untuk memasukkan dua bilangan bulat, lalu program akan menampilkan nilai maksimum dari kedua bilangan tersebut menggunakan fungsi maxab, dan kemudian menukar nilai kedua bilangan tersebut menggunakan fungsi tukar dan menampilkan nilai yang telah ditukar.

19.



The screenshot shows the Apache NetBeans IDE interface. The main editor window displays a Java file named `Tempair.java`. The code defines a `Tempair` class with a `main` method. The `main` method prompts the user to enter a temperature, reads the input using `Scanner`, and then uses a series of `if-else` statements to determine the state of water based on the temperature value. The output window at the bottom shows the execution results, indicating that the program ran successfully and displayed the correct state of water for the input temperature.

```
1 import java.util.Scanner;
2 /* contoh penentuan IF tiga kasus : wujud air */
3 public class Tempair {
4     /**
5      * program suhu
6      */
7     public static void main(String[] args) {
8         // TODO: Auto-generated method stub
9         /* Nama : */
10        int T;
11        Scanner masukan = new Scanner(System.in);
12        /*Program*/
13        System.out.print ("Contoh IF tiga kasus \n");
14        System.out.print ("Temperatur (der. C) = ");
15        T = masukan.nextInt();
16        if (T < 0) {
17            System.out.print ("Wujud air beku \n" + T);
18        } else if ((0 <= T) && (T <= 100)) {
19            System.out.print ("Wujud air cair \n" + T);
20        } else if (T > 100) {
21            System.out.print ("Wujud air uap/gas \n" + T);
22        }
23    }
24 }
```

Output - Tpg2 (run) #3 x

```
run
Contoh IF tiga kasus
Temperatur (der. C) = 110
Wujud air uap/gas
110BUILD SUCCESSFUL (total time: 13 seconds)
```

Penjelasannya: Program ini akan meminta user untuk memasukkan nilai suhu dalam derajat Celsius. Kemudian, program menentukan wujud air berdasarkan nilai suhu yang dimasukkan oleh user. Jika suhu kurang dari 0, program akan menampilkan "Wujud air beku". Jika suhu antara 0 dan 100, program akan menampilkan "Wujud air cair". Jika suhu lebih dari 100, program akan menampilkan "Wujud air uap/gas".