

Nama : Azzahra Aulia Rahman

NIM : 254107020227

JOBSHEET 11

Fungsi 1

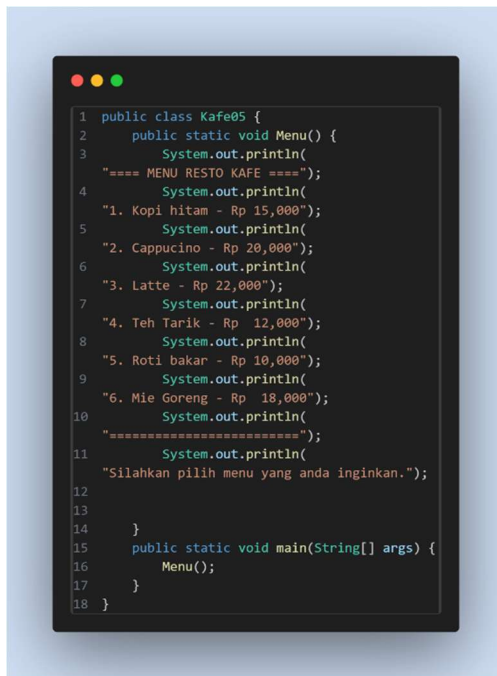
1. Tujuan

1. Mahasiswa mampu memahami penggunaan fungsi static pada Java dengan parameter dan mengembalikan nilai.
2. Mahasiswa mampu membuat program menggunakan fungsi static dan mengeksekusi fungsi tersebut.

2. Praktikum

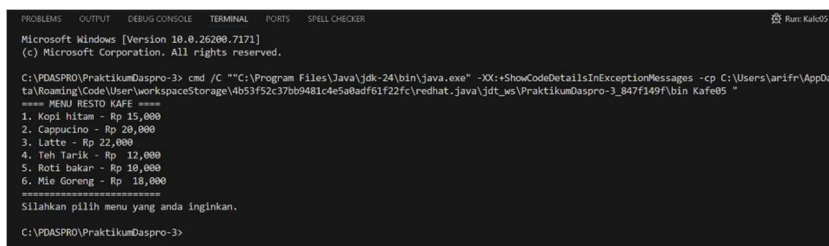
2.1 Percobaan 1: Membuat Fungsi Tanpa Parameter

- Kode Java



```
1 public class Kafe05 {
2     public static void Menu() {
3         System.out.println(
4             "==== MENU RESTO KAFE ===");
5         System.out.println(
6             "1. Kopi hitam - Rp 15,000");
7         System.out.println(
8             "2. Cappuccino - Rp 20,000");
9         System.out.println(
10            "3. Latte - Rp 22,000");
11        System.out.println(
12            "4. Teh Tarik - Rp 12,000");
13        System.out.println(
14            "5. Roti bakar - Rp 10,000");
15        System.out.println(
16            "6. Mie Goreng - Rp 18,000");
17        System.out.println(
18            "=====");
19        System.out.println(
20            "Silahkan pilih menu yang anda inginkan.");
21    }
22    public static void main(String[] args) {
23        Menu();
24    }
25 }
```

- Hasil Run



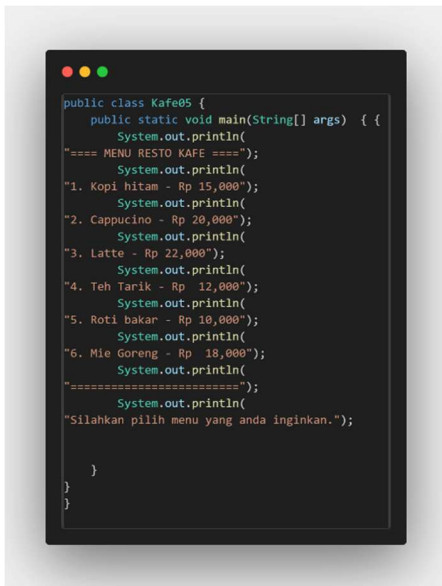
```
Microsoft Windows [Version 10.0.26200.7171]
(c) Microsoft Corporation. All rights reserved.

C:\PDASPRO\PraktikumDaspro-3> cmd /C ""C:\Program Files\Java\jdk-24\bin\java.exe" -XX:+ShowCodeDetailsInExceptionMessages -cp C:\Users\arifn\AppData\Local\Roaming\Code\User\workspaceStorage\4b53f52c37bb9481c4e5a0adf61f22fc\redhat.java\jdt_ws\PraktikumDaspro-3_847f149f\bin Kafe05 "
==== MENU RESTO KAFE ====
1. Kopi hitam - Rp 15,000
2. Cappuccino - Rp 20,000
3. Latte - Rp 22,000
4. Teh Tarik - Rp 12,000
5. Roti bakar - Rp 10,000
6. Mie Goreng - Rp 18,000
=====
Silahkan pilih menu yang anda inginkan.

C:\PDASPRO\PraktikumDaspro-3>
```

Pertanyaan

1. Apakah fungsi tanpa parameter selalu harus bertipe void?
 - Tidak
2. Apakah daftar menu pada program kafe dapat ditampilkan tanpa menggunakan fungsi Menu()? Modifikasi kode program tersebut untuk dapat menampilkan daftar menu tanpa menggunakan fungsi!
 - Ya bisa.



```
public class Kafe05 {  
    public static void main(String[] args) { {  
        System.out.println(  
            "==== MENU RESTO KAFE ====");  
        System.out.println(  
            "1. Kopi hitam - Rp 15,000");  
        System.out.println(  
            "2. Cappuccino - Rp 20,000");  
        System.out.println(  
            "3. Latte - Rp 22,000");  
        System.out.println(  
            "4. Teh Tarik - Rp 12,000");  
        System.out.println(  
            "5. Roti bakar - Rp 10,000");  
        System.out.println(  
            "6. Mie Goreng - Rp 18,000");  
        System.out.println(  
            "=====");  
        System.out.println(  
            "Silahkan pilih menu yang anda inginkan.");  
    }  
}
```

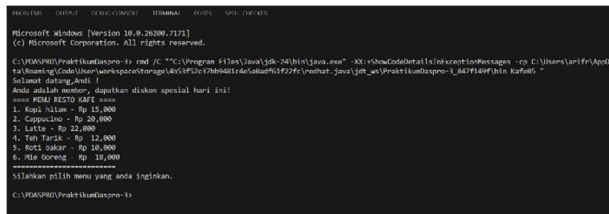
3. Jelaskan keuntungan menggunakan fungsi Menu() dibandingkan menulis semua perintah penampilan menu langsung di dalam fungsi main.
 - Dapat digunakan Kembali, fungsi main menjadi lebih singkat dll...
4. Uraikan secara singkat alur eksekusi program ketika fungsi Menu() dipanggil dari main (mulai dari program dijalankan sampai daftar menu tampil di layar).
 - 1. Program mulai
 - 2. Pemanggilan fungsi
 - 3. Transfer control
 - 4. Eksekusi isi fungsi
 - 5. Kembali ke Main
 - 6. Selesai

2.2 Percobaan 2: Membuat Fungsi Dengan Parameter

- **Kode Java**



- **Hasil Run**



Pertanyaan

1. Apakah kegunaan parameter di dalam fungsi?
 - Fungsi menjadi dinamis, karena bisa memproses data yang berbeda-beda setiap kali dipanggil tanpa mengubah kode didalam fungsi.
2. Jelaskan mengapa pada percobaan ini fungsi Menu() menggunakan parameter namaPelanggan dan isMember?
 - Menggunakan parameter namaPelanggan agar program bisa memberi tanda nama pelanggan dengan spesifik.
 - Menggunakan isMember agar program bisa membedakan mana yang member mana yang bukan.
3. Apakah parameter sama dengan variabel? Jelaskan.
 - Ya, parameter adalah jenis variabel khusus
4. Jelaskan bagaimana cara kerja parameter isMember pada fungsi Menu(). Apa perbedaan output ketika isMember bernilai true dan ketika false?
 - True : Program akan masuk ke dalam blok if dan mencetak kalimat “Anda adalah member, dapatkan diskon 10% untuk setiap pembelian.”

- False : Tidak ada respon
5. Apa yang akan terjadi jika memanggil fungsi Menu() tanpa menyertakan parameter namaPelanggan dan isMember?
 - Akan terjadi error.
 6. Modifikasi kode di atas dengan menambahkan parameter baru kodePromo (String). Jika kodePromo adalah "DISKON50", tampilkan berikan diskon 50%. Jika kodePromo adalah "DISKON30", tampilkan berikan diskon 30%. Jika tidak ada kode promo yang berlaku, tampilkan kode invalid.

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
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

```

7. Berdasarkan fungsi Menu() di atas, jika nama pelanggan adalah "Budi", pelanggan tersebut member, dan menggunakan kode promo "DISKON30", tuliskan satu baris perintah pemanggilan fungsi menu yang benar.

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
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

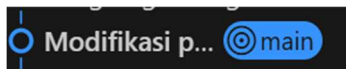
```

8. Menurut Anda, apakah penggunaan parameter namaPelanggan dan isMember pada fungsi Menu() membuat program lebih mudah dibaca dan dikembangkan dibandingkan jika nilai-nilai tersebut ditulis langsung di dalam fungsi tanpa parameter? Jelaskan alasan Anda.

- Ya penggunaan parameter membuat jauh lebih mudah dibaca dan dikembangkan.

Karena, jika ada parameter akan bersifat dinamis yaitu satu fungsi yang sama bisa digunakan untuk ribuan pelanggan, lalu dapat digunakan Kembali, bisa memanggil fungsi berulang – ulang dengan data yang berbeda – beda.

9. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 2”



2.3 Percobaan 3: Membuat Fungsi dengan Nilai Kembalian

- Kode Java

```
1 import java.util.Scanner;
2 public class Lab05 {
3     public static void Menu() {
4         System.out.println("==== MENU RESIJO KOMB ====");
5     }
6     System.out.println("1. Kopi Hitam - Rp 15,000");
7     System.out.println("2. Cappuccino - Rp 20,000");
8     System.out.println("3. Latte - Rp 22,000");
9     System.out.println("4. Teh Tarik - Rp 12,000");
10    System.out.println("5. Buih Buih - Rp 10,000");
11    System.out.println("6. Mie Goreng - Rp 18,000");
12    System.out.println("====");
13    System.out.println("Silahkan pilih menu yang anda inginkan.");
14    }
15 }
16
17 public static int hitungTotalHarga05 (int
18 pilihMenu, int banyakItem) {
19     int harga = 0;
20     switch (pilihMenu) {
21         case 1: harga = 15000; break;
22         case 2: harga = 20000; break;
23         case 3: harga = 22000; break;
24         case 4: harga = 12000; break;
25         case 5: harga = 10000; break;
26         case 6: harga = 18000; break;
27         default:
28             System.out.println(
29 "Menu tidak tersedia.");
30             return 0;
31     }
32     return harga * banyakItem;
33 }
34
35 public static void main(String[] args) {
36     Scanner sc = new Scanner(System.in);
37     Menu();
38     System.out.print(
39 "Masukkan nomor menu yang ingin Anda pesan: ");
40     int pilihMenu = sc.nextInt();
41     System.out.print(
42 "Masukkan banyak item yang ingin dipesan: ");
43     int banyakItem = sc.nextInt();
44     int totalBayar = hitungTotalHarga05(pilihMenu
45 , banyakItem);
46     System.out.println(
47 "Total yang harus dibayar: Rp " + totalBayar);
48     sc.close();
49 }
50 }
51 }
52 }
53 }
54 }
55 }
```

- **Hasil Run**

```
C:\PDASPRO\PraktikumDaspro-3>
C:\PDASPRO\PraktikumDaspro-3> c: && cd c:\PDASPRO\PraktikumDaspro-3 && cmd /C "C:\Program Files\Java\jdk-24\bin\java.exe" -XX:+ShowCodeDetailsInExceptionMessages
-
User: VerifAgData\Running\Code\User\workspace\storage\4b53f52c37a69481c4e5abaf61f22fc\redhat_java\jdk_ws\PraktikumDaspro-3_847f149f\bin_Kafe05
===== MENU RESTO KAFE =====
1. Kopi hitam - Rp 15,000
2. Cappuccino - Rp 20,000
3. Latte - Rp 22,000
4. Teh Tarik - Rp 12,000
5. Roti bakar - Rp 10,000
6. Mie Goreng - Rp 10,000
=====
Silahkan pilih menu yang anda inginkan.
Masukkan nomor menu yang ingin Anda pesan: █
```

Pertanyaan

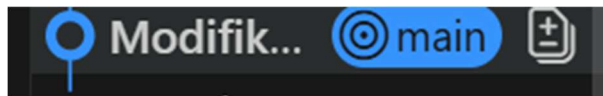
1. Jelaskan secara singkat kapan suatu fungsi membutuhkan nilai kembalian (return value) dan kapan fungsi tidak perlu mengembalikan nilai. Berikan minimal satu contoh dari program kafe pada Percobaan 3 untuk masing-masing kasus.
 - Suatu fungsi membutuhkan return value saat menghasilkan data yang harus diolah lagi atau disimpan oleh fungsi pemanggilnya. Ex : `hitungTotalHarga05`
 - Suatu fungsi yang tidak membutuhkan return(void) saat fungsi hanya menjalankan aksi selesai tidak ada pengembalian data.
2. Fungsi `hitungTotalHargaNoPresensi` saat ini mengembalikan total harga berdasarkan pilihanMenu dan jumlahPesanan. Sebutkan tipe data nilai kembalian dan dua buah parameter yang digunakan fungsi tersebut. Jelaskan arti masing-masing parameter dalam konteks program kafe.
 - `int pilihanMenu` : Parameter ini mempresentasikan nomor menu yang dipilih oleh user. Gunanya untuk menentukan harga satuan barang.
 - `Int banyakItem` : Variabel ini mempresentasikan jumlah porsi yang dibeli. Gunanya untuk dikalikan dengan harga satuan.
3. Modifikasi kode di atas sehingga fungsi `hitungTotalHargaNoPresensi` dapat menerima kodePromo. Jika kodePromo adalah "DISKON50", maka mendapat diskon 50% dari totalHarga dan tampilkan diskon. Jika kodePromo adalah "DISKON30", maka mendapat diskon 30% dari totalHarga dan tampilkan diskon. Jika tidak ada kode promo yang berlaku, tampilkan kode invalid dan tidak ada pengurangan total harga totalHarga.



4. Modifikasi kode di atas sehingga bisa memilih beberapa jenis menu berbeda serta menampilkan total keseluruhan pesanan. Bagaimana memodifikasi program sehingga pengguna dapat: memesan lebih dari satu jenis menu (misalnya menu 1 dan 3 sekaligus), dan menampilkan total keseluruhan pesanan (gabungan dari semua jenis menu)?

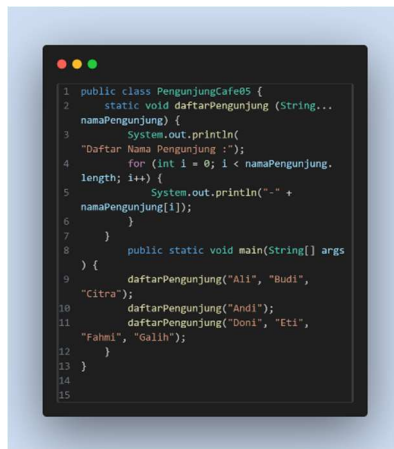


5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 3”

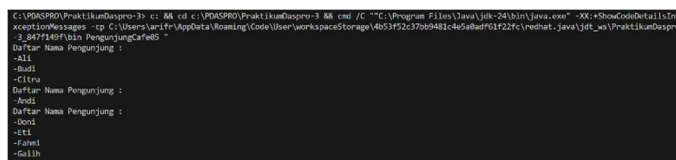


2.4 Percobaan 4: Fungsi Varargs

- **Kode Java**



- **Hasil Run**



Pertanyaan

1. Jelaskan mengapa penulisan parameter di praktikum 4 ditulis dengan String... namaPengunjung!
- Karena itu adalah varargs yang digunakan agar method daftarPengunjung bersifat fleksibel dan dapat menerima jumlah argument yang dinamis.
2. Modifikasi method daftarPengunjung menggunakan for-each loop.

```
public class PengunjungCafe05 {
    static void daftarPengunjung (String...namaPengunjung) {
        System.out.println(x: "Daftar Nama Pengunjung :");
        for (String nama : namaPengunjung) {
            System.out.println("-" + nama);
        }
    }

    public static void main(String[] args) {
        daftarPengunjung(...namaPengunjung: "Ali", "Budi", "Citra");
        daftarPengunjung(...namaPengunjung: "Andi");
        daftarPengunjung(...namaPengunjung: "Doni", "Eti", "Fahmi", "Galih");
    }
}
```

3. Bisakah menggunakan dua tipe data varargs dalam satu fungsi? Jelaskan jawaban Anda berdasarkan aturan varargs di Java, dan berikan contohnya!
- Tidak, karena varargs hanya bisa satu parameter dalam sebuah method, dan parameter varargs harus diletakkan terakhir dalam posisi parameter.
4. Jelaskan apa yang terjadi jika fungsi daftarPengunjung dipanggil tanpa argumen. Apakah program akan error saat kompilasi, error saat dijalankan, atau tetap berjalan? Jika tetap berjalan, bagaimana output yang dihasilkan?
- Program akan tetap berjalan, tetapi outputnya akan mencetak judulnya saja karena loop tidak akan dieksekusi.
5. Commit dan push hasil modifikasi Anda ke Github dengan pesan “Modifikasi Percobaan 4”



3. Tugas

1. Buatlah sebuah class KubusNoPresensi yang di dalamnya terdapat fungsi untuk menghitung volume kubus dan luar permukaan kubus!
Commit dan push program Anda ke Github dengan pesan “Tugas1”

- **Kode Java**

```

1 import java.util.Scanner;
2 public class KuisDasar {
3     static int hitungDasar (int sisi) {
4         return sisi * sisi * sisi;
5     }
6
7     static int hitungLuasPrismn (int sisi) {
8         return 6 * sisi * sisi;
9     }
10
11     public static void main(String[] args) {
12         Scanner sc = new Scanner(System.in);
13
14         System.out.print(
15             "Masukkan panjang sisi kubus : ");
16         int sisi = sc.nextInt();
17
18         int volume = hitungDasar(sisi);
19         int luasPermukaan = hitungLuasPrismn(
20             sisi);
21
22         System.out.println("Volume kubus : "
23             + volume);
24         System.out.println(
25             "Luas permukaan kubus : " + luasPermukaan);
26         sc.close();
27     }
28 }

```

- **Hasil Run**

[illegible]

2. Di suatu kelas, terdapat N mahasiswa yang mengikuti satu mata kuliah. Nilai tiap mahasiswa untuk mata kuliah tersebut akan diinputkan oleh pengguna dan disimpan dalam sebuah array 1 dimensi berukuran N. Buatlah sebuah class NilaiMahasiswaNoPresensi yang di dalamnya terdapat beberapa fungsi berikut:
- Fungsi isianArray untuk:
 - Mengisi elemen-elemen array dengan nilai yang diinputkan oleh user (jumlah elemen = N, dibaca dari input)
 - Fungsi tampilArray untuk:
 - Menerima parameter berupa array nilai mahasiswa
 - Menampilkan seluruh nilai yang telah diinputkan
 - Fungsi hitTot untuk:
 - Menerima parameter berupa array nilai mahasiswa
 - Menghitung dan mengembalikan total nilai seluruh mahasiswa (return tipe int atau double)
 - Fungsi main untuk:
 - Membaca nilai N (jumlah mahasiswa) dari input pengguna
 - Membuat array dengan ukuran N
 - Memanggil fungsi isianArray, tampilArray, dan hitTot
 - Menampilkan total nilai yang dikembalikan oleh fungsi hitTot

Commit dan push program Anda ke Github dengan pesan “Tugas2”

- a. Buatlah sebuah class `RekapPenjualanCafeNoPresensi` yang digunakan untuk menyimpan dan mengolah data penjualan 5 menu kafe dari hari pertama hingga hari ketujuh. Di dalam class tersebut, buat dan gunakan beberapa fungsi (method) berikut:
 - Fungsi untuk menginputkan data penjualan setiap menu untuk setiap hari (petunjuk: menggunakan array 2 dimensi).
 - Fungsi untuk menampilkan seluruh data penjualan dari hari pertama hingga hari terakhir dalam bentuk tabel
 - Fungsi untuk menampilkan menu yang memiliki total penjualan tertinggi selama tujuh hari (tampilkan nama menu dan total penjualannya).
 - Fungsi untuk menampilkan rata-rata penjualan untuk setiap menu selama tujuh hari (tampilkan nama menu dan nilai rata-ratanya).

Gunakan fungsi main untuk:

- Memanggil fungsi input data penjualan,
- Menampilkan seluruh data penjualan,
- Menampilkan menu dengan penjualan tertinggi, dan
- Menampilkan rata-rata penjualan untuk setiap menu.

b. Modifikasi program pada 3 (a) sehingga jumlah hari penjualan dan jumlah menu dibaca dari input pengguna, lalu sesuaikan ukuran array dan proses perhitungan agar tetap berjalan dengan benar.

Commit dan push program Anda ke Github dengan pesan “Tugas3”

- **Kode Java**



```
1 // Program 3.1: Menampilkan Data Penjualan
2 // Nama: Muhammad Fauzan Fauzan Fauzan
3 // NPM: 12020201001
4 // Tanggal: 10/05/2022
5 // Deskripsi: Program ini menampilkan data penjualan, mencari menu dengan penjualan tertinggi, dan menampilkan rata-rata penjualan untuk setiap menu.
6 //
7 // Variabel:
8 // - menu: array of menu names
9 // - harga: array of menu prices
10 // - penjualan: array of daily sales data
11 //
12 // Fungsi:
13 // - inputData: function to input daily sales data
14 // - tampilkanData: function to display all sales data
15 // - cariMenuTertinggi: function to find the menu with the highest sales
16 // - rataRata: function to calculate the average sales for each menu
17 //
18 //
19 //
20 //
21 //
22 //
23 //
24 //
25 //
26 //
27 //
28 //
29 //
30 //
31 //
32 //
33 //
34 //
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 //
```