

LAPORAN PRAKTIKUM ALGORITMA PEMROGRAMAN
PERULANGAN WHILE, DO WHILE DAN SENTINEL PADA PROGRAM
JAVA



Oleh :

DERIEL CHAERAHMAN

NIM 2411533007

ALGORITMA DAN PEMROGRAMAN

DOSEN PENGAMPU :

DR. WAHYUDI, S.T, M.T

FAKULTAS TEKNOLOGI INFORMASI

DEPARTEMEN INFORMATIKA

UNIVERSITAS ANDALAS

2024

A. Pendahuluan

Praktikum ini dilakukan untuk membuat program yang dapat melakukan suatu kondisi perulangan tanpa batas selama kondisi bernilai True dengan menggunakan perulangan While, Do While dan sentinel. Menggunakan `java.util.Random` untuk menghasilkan angka acak dan digabungkan dengan perulangan untuk menghasilkan program penjumlahan sederhana. Menggunakan scanner class untuk dapat menerima input dari user sebagai suatu kondisi/batasan perulangan yang akan dijalankan oleh perulangan dan menggabungkannya dengan If statement untuk menampilkan outputnya lebih kompleks dalam pemrograman bahasa Java.

1. Random Class

Digunakan untuk menghasilkan nilai/angka acak. Untuk menggunakannya perlu mengimpor kelasnya dari paket `java.util.Random`. Dapat diberikan seed (nilai awal) untuk memastikan hasil yang dihasilkan dapat direproduksi, namun jika tidak menggunakan seed maka sistem akan menggunakan nilai default. Dapat digunakan untuk menghasilkan nilai acak untuk membuat program game penjumlahan (dadu) sederhana, game suit, pengujian dan validasi untuk menguji berbagai skenario dan memastikan bahwa program dapat menangani input yang tidak terduga dan Kriptografi untuk tugas-tugas yang tidak memerlukan keamanan tinggi (lebih direkomendasikan menggunakan `Java SecureRandom`). Cara menggunakannya :

1. Import class dengan syntax : `import java.util.Random;`
2. Buat objek dari kelas : `Random random = new Random();`
3. Method dari objek Random dapat dibuat dengan :
 - a. Int, angka acak tipe bilangan bulat. Syntaxnya '`nextInt();`'
Contoh : `int randomInt = random.nextInt();`
 - b. Double, angka acak tipe bilangan berkoma. Syntaxnya '`nextDouble();`'
Contoh : `double randomDouble = random.nextDouble();`
 - c. Boolean, nilai Acak dari True or False. Syntaxnya '`nextBoolean();`'
Contoh : `boolean randomBoolean = random.nextBoolean();`

2. Perulangan While (While loop)

Perulangan While digunakan untuk mengulang eksekusi suatu statement tanpa batas selama kondisi bernilai True. Syarat berhentinya perulangan While adalah ketika kondisi bernilai False. Struktur dari While loop hanya terdiri dari kondisi, tapi untuk menghindari infinite loop yang dikarenakan kondisi selalu bernilai True, dapat ditambahkan updation agar nilai dari variabel pada 'kondisi' dapat mencapai kondisi puncak/false. Syntax nya :

```
while (Kondisi) {  
    //statement  
    //updation (increment dan decrement)  
}
```

3. Perulangan Do While

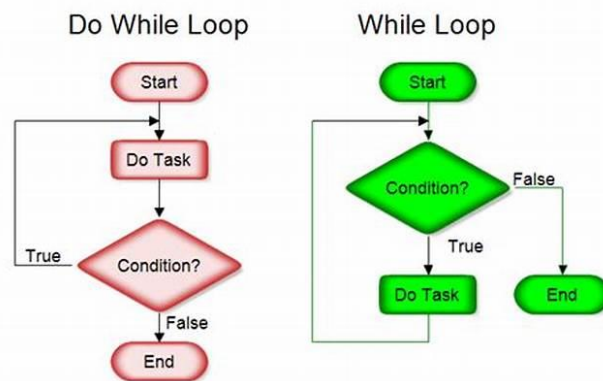
Perulangan ini mirip dengan While loops, bedanya Do While loop akan langsung mengeksekusi statement setidaknya sekali. Blok kode didalam kurawal Do akan

langsung dieksekusi kemudian akan diperiksa 'kondisi' pada While untuk dilakukan perulangan. Syntax nya :

```
do {  
    System.out.println("Input Passowrd: ");  
    phrase = console.nextLine();  
} while (!phrase.equals("abcd"));
```

4. Sentinel loop

Perulangan ini menggunakan nilai khusus (sentinel) untuk menentukan berhentinya perulangan. Nilai sentinel biasanya berasal dari input user yang digunakan untuk mengakhiri perulangan. Dapat dibuat dengan menggunakan While/Do While loop yang 'kondisi'nya terdiri dari operasi antara input(user) dengan nilai sentinel yang ditetapkan contohnya (input != nilaiSentinel), program akan terus bernilai True selama nilai input tidak sama dengan nilai sentinel.



B. Tujuan

Tujuan dari dilakukannya praktikum ini adalah :

1. Mengetahui dan memahami Perulangan yang kondisinya tidak pasti/tidak terbatas dengan While/ Do While loop dan sentinel loop.
2. Menghasilkan nilai/angka acak dengan Java.util.Random
3. Memahami dan mengaplikasikan While loop dalam program java.
4. Memahami dan mengaplikasikan Do While loop dalam program java.
5. Memahami dan mengaplikasikan Sentinel loop dalam program java.
6. Membuat program dengan menggunakan random class dan While/Do While/ Sentinel loop.

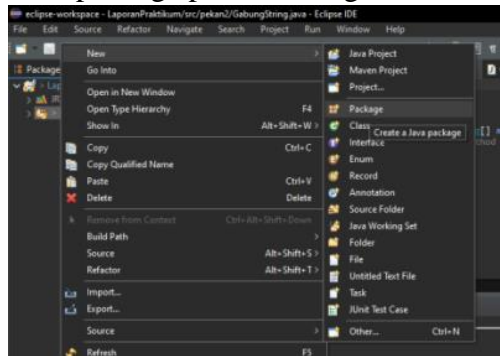
C. Langkah kerja praktikum

a. Alat dan Bahan

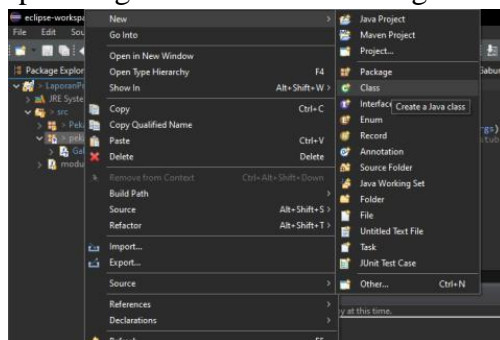
1. Perangkat computer atau laptop
2. Jaringan internet
3. IDE (Integreted Development Environment) direkomendasikan Eclipse IDE
4. Java JDK (Java Development Kit)

b. Program Perulangan While 1

1. Buat package pekan 6 dengan cara klik kanan pada src.



2. Klik kanan pada package pekan 6 untuk membuat class baru dan beri judul 'perulanganWhile1' dan centang tombol pada method public.



3. Import Scanner class dari paket Java.util.Scanner terlebih dahulu untuk menerima input dari user.

```
1 package pekan6;
2
3 import java.util.Scanner;
4
5 public class perulanganWhile1 {
```

4. Deklarasi variabel, variabel 'counter' bertipe int dengan inisialisasi awal 0 (nol) yang nanti akan bertambah ketika dengan perulangan dilakukan. Variabel 'jawab' bertipe String untuk menerima output user dan variabel 'running' bertipe boolean dengan nilai True digunakan sebagai kondisi pada While loop.

```
7 public static void main(String[] args) {
8     int counter = 0;
9     String jawab;
10    boolean running = true;
```

5. Lakukan deklarasi scan dengan membuat objek scan.

```
11    //deklarasi scanner
12    Scanner scan = new Scanner(System.in);
```

6. Buat While loop dengan kondisinya diisi oleh variabel 'running'. Gunakan variabel 'counter' untuk yang nilainya akan bertambah ketika perulangan dilakukan.

```
13    while (running) {
14        counter++;
```

7. Buat statement didalan While loop, Buat printf untuk output variabel counter dan println untuk memberi keterangan kepada user saat menginput pada kotak input. Buat metode scanner dari variabel string 'jawab' dengan 'nextLine();'.

```

15         System.out.println("Jumlah = " + counter);
16         System.out.println("Apakah lanjut (ya / tidak?)");
17         jawab = scan.nextLine();

```

8. If statement ini ditulis masih di dalam While loop statement. Buat kondisi pada If statement yang jika input pada variabel 'jawab' adalah 'tidak' maka kondisi akan bernilai True maka blok kode variabel running akan diperbarui menjadi False sehingga While loop berhenti. Dengan menggunakan 'equalsIgnoreCase' agar input tidak case sensitif ('tidak', 'Tidak').

```

18         //cek jawab = tidak, perulangan berhenti
19         if (jawab.equalsIgnoreCase("tidak")) {
20             running = false;
21         }
22     }

```

9. Setelah perulangan selesai, buat println variabel counter untuk menampilkan jumlah perulangan yang telah dilakukan.

```

23         System.out.println("Anda sudah melakukan perulangan sebanyak "+counter+" kali");
24     }
25 }

```

c. Program Do While 1

1. Buat terlebih dahulu classnya dengan klik kanan pada package pekan 6 dan beri judul 'doWhile1', centang method public static void.
2. Import Scanner class dari paket Java.util.Scanner terlebih dahulu untuk menerima input dari user.

```

1 package pekan6;
2
3 import java.util.Scanner;
4
5 public class doWhile1 {

```

3. Lakukan deklarasi console dengan membuat objek scan dan deklarasi variabel String 'phrase'.

```

7 public static void main(String[] args) {
8     Scanner console = new Scanner(System.in);
9     String phrase;

```

4. Buat 'do' statement, dan isi blok kode dengan println("Input Passowrd: "); sebagai keterangan ketika user menginputkan. Buat metode scanner dari variabel string 'phrase' dengan '.nextLine();'.

```

10     do {
11         System.out.println("Input Passowrd: ");
12         phrase = console.nextLine();

```

5. Buat While loop dengan kondisi '!phrase.equals("abcd")' untuk memeriksa apakah input pengguna bukan "abcd". Jika input bukan 'abcd' maka perulangan akan dilakukan, namun jika menginputkan 'abcd' maka kondisi akan bernilai False (Phrase='abcd' equals dengan ("abcd") sehingga kondisi bernilai True, tapi terdapat tanda seru '!' yang berfungsi membalikan nilai/logika not jadi kondisi akhir menjadi False) sehingga perulangan berhenti.

```

13     } while (!phrase.equals("abcd"));
14 }
15
16 }

```

d. Program Sentinel Loop

1. Buat terlebih dahulu classnya dengan klik kanan pada package pekan 6 dan beri judul 'SentinelLoop', centang method public static void.
2. Import Scanner class dari paket Java.util.Scanner terlebih dahulu untuk menerima input dari user.

```
1 package pekan6;
2
3 import java.util.Scanner;
4
5 public class SentinelLoop {
```

3. Deklarasi dan insialisasi, buat variabel int 'sum' beri nilai awal nol, buat variabel int 'number' beri nilai awal 12(tidak wajib, karena nantinya variabel ini akan digunakan untuk obek scanner sehingga nilai nya akan diperbarui). Buat objek scan yang berada didalam 'try'. 'try' ini digunakan untuk mendeklarasikan resource yaitu objek yang membutuhkan penutupan eksplisit setelah digunakan seperti input scanner. Resource(konteksnya disini scanner) yang berada di dalam kurung secara otomatis akan ditutup setelah blok try selesai, jadi tidka perlu pakai seperti 'console. close();'.

```
7 public static void main(String[] args) {
8     try (Scanner console = new Scanner (System.in)) {
9         int sum = 0;
10        int number = 12; // "dummy value", anything but 0
```

4. Buat While loop dengan kondisinya menggunakan variabel number dan nilai nol dengan operator logika not (!=). Buat statement yang akan dijalankan jika kondisi bernilai True. Buat println untuk memberi keterangan kepada user saat menginput pada kotak input. Buat metode scanner dari variabel Int 'number' dengan '.nextInt();' untuk memperbarui nilai dari variabel number(selain input nol karena kondisinya akan bernilai False sehingga perulangan dihentikan). Dan panggil kembali variabel sum yang digunakan untuk menghitung nilai dari input user pada variabel number yang disimpan pada variabel.

```
12        while (number != 0) {
13            System.out.println("Masukkan angka (0 untuk keluar): ");
14            number = console.nextInt();
15            sum = sum + number;
16        }
```

5. Setelah tutup kurawal while loop, tulis println untuk menampilkan variabel sum.

```
16        }
17        System.out.println("totalnya adalah " + sum);
18    }
19 }
20 }
```

e. Program Lempar Dadu

1. Buat terlebih dahulu classnya dengan klik kanan pada package pekan 6 dan beri judul 'Lempardadu', centang method public static void.
2. Import Random class dari paket Java.util.Random terlebih dahulu untuk membuat program yang dapat menghasilkan angka/nilai acak.

```
1 package pekan6;
2
3 import java.util.Random;
4
5 public class Lempardadu {
```

3. Lakukan deklarasi rand dengan membuat objek Random. Deklarasi dan inisialisasi variabel int 'tries' dan variabel int 'sum', beri nilai awal keduanya nol.

```
7 public static void main(String[] args) {
8     Random rand = new Random();
9     int tries = 0;
10    int sum = 0;
```

4. Buat While loop yang kondisinya menggunakan variabel sum dan nilai batas 7 dengan operator logika not (!=).

```
11    while (sum != 7) {
```

5. Di dalam while loop buat statement, buat variabel int dadu1 dan dadu2 lalu buat method objek randomnya dengan nilai (6) + 1, diberi nilai tambah satu untuk memulai seed awal nya menjadi 1 dan sampai 6, jika tidak ditambah satu maka seed awal nya mulai dari nol dan sampai 5.

```
12        //roll the dice once
13        int dadu1 = rand.nextInt(6) + 1;
14        int dadu2 = rand.nextInt(6) + 1;
```

6. Gunakan variabel sum untuk menambahkan nilai dari angka acak dadu 1 dan dadu 2 dan disimpan di variabel sum. Setelahnya buat println yang akan menampilkan output dari penambahan dadu 1 dan dadu 2. Terakhir buat updation dengan menggunakan variabel 'tries' yang akan ditambah satu untuk setiap perulangan yang dilakukan.

```
15        sum = dadu1 + dadu2;
16        System.out.println(dadu1 + " + " + dadu2 + " = " + sum);
17        tries++;
18    }
```

7. Buat println yang menyatakan banyaknya perulangan yang dilakukan dengan menggunakan output dari variabel tries.

```
19        System.out.println("You won after " + tries + " tries! ");
20    }
21
22 }
```

f. Program Game Penjumlahan

1. Buat terlebih dahulu classnya dengan klik kanan pada package pekan 6 dan beri judul 'GamePenjumlahan', centang method public static void.
2. Import Random class dari paket Java.util.Random untuk membuat program yang dapat menghasilkan angka/nilai acak dan import Scanner class dari paket Java.util.Scanner terlebih dahulu untuk menerima input dari user.

```
1 package pekan6;
2
3 import java.util.Random;
4 import java.util.Scanner;
5
6 public class GamePenjumlahan {
```

3. Lakukan deklarasi rand dengan membuat objek Random dan deklarasi console dengan membuat objek scan.

```
8 public static void main(String[] args) {
9     Scanner console = new Scanner (System.in);
10    Random rand = new Random();
```

4. Deklarasi dan inisialisasi variabel int 'points' dan variabel int 'wrong' dengan nilai awal keduanya 0 (nol).

```
11 //play until user gets 3 wrong
12 int points = 0;
13 int wrong = 0;
```

5. Buat while loop yang kondisinya menggunakan variabel 'wrong' dan nilai batas 3 dengan operator aritmatika <.

```
14 while (wrong < 3) {
```

6. Buat statement di dalam while loop. Buat variabel int 'result' dengan method 'play' yang berfungsi untuk menampilkan soal penjumlahan dan mendapatkan input jawaban dari user. Buat If statement sebagai sistem penilaian/score, untuk bagian kondisi menggunakan variabel result dengan nilai dari input user dengan pembandingan nol dan operator aritmatika >, yang jika input user >0 maka akan bernilai True dan dieksekusi variabel points yang nilainya akan ditambah satu, jika jawabannya salah, maka else akan dieksekusi sehingga variabel wrong akan ditambah dengan nilai satu.

```
14 while (wrong < 3) {
15     int result = play(console, rand); //play one game
16     if (result > 0) {
17         points++;
18     } else {
19         wrong++;
20     }
21 }
```

7. Buat println untuk menunjukkan jumlah benar dengan variabel points.

```
22 System.out.println("You earned " + points + " total points.");
23 }
```

8. Buat method baru, 'public static int play(Scanner console, Random rand)', int adalah tipe data return dari method ini yang artinya, method play akan mengembalikan nilai bertipe integer (int). play digunakan untuk mengembalikan nilai berupa skor jika jawaban user benar dan jika salah mengembalikan nilai 0 (nol). Scanner console digunakan agar method play dapat menerima input dari scanner. Random rand digunakan agar method play dapat menerima objek Random untuk menghasilkan nilai acak.

```
24 //membuat soal penjumlahan dan ditampilkan ke user
25 public static int play(Scanner console, Random rand) {
```

9. Program berikut berfungsi untuk membuat soal. Buat statement didalam public static, deklarasi variabel int operands lalu buat method objek randomnya dengan nilai (4) + 2 dan buat variabel int sum dengan method objek randomnya dengan nilai (10) + 1. Buat println untuk menampilkan variabel sum.

```
24 //membuat soal penjumlahan dan ditampilkan ke user
25 public static int play(Scanner console, Random rand) {
26     //print the operands being added, and sum them
27     int operands = rand.nextInt(4) + 2;
28     int sum = rand.nextInt(10) + 1;
29     System.out.print(sum);
```

10. Masih di dalam public static, buat For statement dengan inisialisasi i = 2, 'kondisi' antara i dengan variabel operands dengan operator relasional <=. Updation 'i' dengan nilai satu. Kemudian buat random objek dari variabel int n dengan jangkauan nilai acaknya (10) + 1. Panggil variabel sum untuk ditambahkan dengan nilai dari variabel n, ini berfungsi untuk menghitung

hasil akhir(tapi tidak ditampilkan ke user). Buat print untuk menampilkan soal yang angkanya berasal variabel n dan buat string operator tambah agar diantara angka terdapat operator tambah. Setelah kurawal tutup for statement buat print untuk memunculkan '=' pada output dari angka acak dari print n sebelumnya sehingga terbuatlh soal angka acaknya.

```
30     for (int i = 2; i <= operands; i++) {
31         int n = rand.nextInt(10)+1;
32         sum += n;
33         System.out.print(" + " + n);
34     }
35     System.out.print(" = ");
```

11. Program berikut dibuat untuk memeriksa jawaban dari input user. Buat variabel int 'guess' dengan objek console untuk membaca input yang nantinya input dari user akan disimpan di variabel ini. Buat If statement dengan kondisi nilai variabel 'guess'(input user) apakah sama dengan nilai dari variabel sum (hasil penjumlahan yang dihitung sebelumnya ketika membuat program soal penjumlahan).

```
37     //read user's guess and report whether it was correct
38     int guess = console.nextInt();
39     if (guess == sum) {
40     }else {
41         System.out.println("Wrong! The answer was " + sum);
42         return 0;
43     }
44     return guess;
45 }
46
47 }
```

12. Jika kondisi If statement bernilai True, tidak ada blok kode yang dieksekusi melainkan program akan menjalankan return guess yang menunjukkan bahwa jawaban benar dan proses soal tersebut selesai, lalu program akan mencetak ulang soal.
13. Jika kondisi If statement bernilai False, maka blok kode pada else akan dieksekusi dan menampilkan output dari println jawaban dari user salah dan menampilkan hasil yang benar yang tersimpan di variabel sum (yang sebelumnya digunakan untuk membuat program soal). Return 0 juga akan dieksekusi yang digunakan dalam logika permainan untuk menghitung jumlah jawaban yang salah, yang nantinya variabel int **wrong** pada **publik main** akan ditambah.

g. Program Tugas Melempar Dadu

1. Buat terlebih dahulu classnya dengan klik kanan pada package pekan 6 dan beri judul 'Tugasmelopardadu, centang method public static void.
2. Import Scanner class dan Random class

```
1 package pekan6;
2
3 import java.util.Random;
4 import java.util.Scanner;
```

3. Deklarasi rand dengan membuat objek Random dan deklarasi console dengan membuat objek scan. Deklarasi variabel boolean running dengan inisialisasi True, buat variabel int sum, dan buat variabel int count dengan inisialisasi 0.

```

8 public static void main(String[] args) {
9     Scanner console = new Scanner (System.in);
10    Random rand = new Random();
11    boolean running = true;
12    int sum;
13    int count = 0;

```

4. Buat While loop dengan kondisi menggunakan variabel running, buat dice dengan deklarasi variabel int dice 1 dan dice 2 dan buat sebagai method objek random dengan jangkauan nilai (6) + 1 (1 sampai 6). Gunakan variabel sum sebagai tempat penyimpanan penjumlahan dari 2 dice. Buat println untuk menampilkan output dari dice 1 dan dice 2 serta hasil penjumlahannya yang disimpan di variabel sum. Panggil variabel count dengan updation + 1 untuk setiap perulangan yang dilakukan.

```

15    while (running) {
16        int dice1 = rand.nextInt(6) + 1;
17        int dice2 = rand.nextInt(6) + 1;
18        sum = dice1 + dice2;
19        System.out.println(dice1 + " + " + dice2 + " = " + sum);
20        count++;

```

5. Masih di dalam While loop, buat if statement dengan kondisi apakah nilai dari sum sama dengan 7, jika If kondisi True buat println yang menampilkan pesan jika tebakan benar/sum bernilai 7 dan tampilkan variabel count untuk melihat banyaknya perulangan. Buat break untuk mengakhiri perulangan.

```

22        if (sum == 7) {
23            System.out.println("Selamat! Tebakan Anda Benar!");
24            System.out.println("Anda menang setelah " + count + " percobaan!");
25            break;

```

6. Buat else jika kondisi If False yang menampilkan println bahwa tebakan anda salah dan pesan apakah user ingin melempar dadu/mengulangi perulangan lagi. Buat variabel string jawab1 dengan method objek scan dan tambahkan 'toLowerCase()' yang berfungsi mengubah teks input user menjadi huruf kecil untuk memudahkan perbandingan pada program dan menghindari case sensitive.

```

26        } else {
27            System.out.println("Tebakan Anda Salah!");
28            System.out.print("Apakah Anda ingin melempar dadu lagi (ya/tidak)? ");
29            String jawab1 = console.nextLine().toLowerCase();

```

7. Buat If statement yang masih di dalam else statement dengan kondisi input dari user apakah sama dengan 'ya'. Jika input user adalah 'tidak' maka 'tidak' = 'ya' menghasilkan !False menjadi True, sehingga blok kode variabel running akan diperbarui jadi False. Karena kondisi dari while loop menggunakan variabel running, maka kondisi While akan berubah jadi False sekarang dan mengakhiri perulangan.

```

31            if (!jawab1.equals("ya")) { //Untuk menghentikan perulangan
32                running = false;
33                System.out.println("Anda gagal menang setelah " + count + " percobaan.");
34            }
35        }
36    }
37 }
38 }

```

D. Hasil Pratikulum

1. Ouput program Perulangan While 1. Program akan menampilkan println yang meminta user untuk menginputkan ya/tidak. Jika 'ya' maka program akan menampilkan angka yang merupakan jumlah perulangan yang dilakukan dan mengulang println 'ya/tidak' lagi. Jika 'tidak' maka program akan akan menghentikan perulangan dan menampilkan output println banyaknya perulangan yang telah dilakukan. Ketika diinput 'tidak' kondisi If statement akan bernilai True sehingga nilai dari variabel Boolean running(yang digunakan sebagai kondisi While loop) akan berubah jadi False sehingga perulangan berakhir.

```
Jumlah = 1
Apakah lanjut (ya / tidak?)
ya
Jumlah = 2
Apakah lanjut (ya / tidak?)
tidak
Anda sudah melakukan perulangan sebanyak 2 kali
```

2. Output program Do While 1. Blok kode pada akan langsung mengeksekusi menampilkan println yang menyuruh untuk menginputkan password (yang sebelumnya di dalam program sudah ditentukan passwordnya), jika input salah (tidak sesuai dengan kondisi dari While loop), maka kondisi akan bernilai !False = True sehingga while loop akan terus berjalan. Ketika program dijalankan dan di input yang tidak sama dengan 'password' maka terjadi perulangan, namun ketika memasukan password yang benar 'abcd' maka !True = False sehingga kondisi While loop berhenti.

```
Input Passowrd:
xyz
Input Passowrd:
abcd
|
```

3. Output program Sentinel Loop. Program ini akan meminta user untuk menginputkan angka dan akan melakukan perulangan untuk setiap input yang tidak bernilai nol. Ketika diinput angka nol maka perulangan akan berhenti dan semua input sebelumnya akan dijumlahkan dan ditampilkan hasilnya.

```
Masukkan angka (0 untuk keluar):
2
Masukkan angka (0 untuk keluar):
4
Masukkan angka (0 untuk keluar):
6
Masukkan angka (0 untuk keluar):
0
totalnya adalah 12
```

4. Output program Lempar Dadu. Program ini akan langsung berjalan otomatis sampai, menghasilkan angka acak dari 2 dadu yang diprogram dengan random class untuk dapat menghasilkan nilai acak lalu ditambahkan. Perulangan akan terus berjalan sampai mendapatkan penjumlahan 2 dadu yang bernilai 7.

```
4 + 6 = 10
5 + 2 = 7
You won after 2 tries!
```

5. Ouput program Game Penjumlahan. Program ini akan menampilkan soal bilangan dengan menggunakan random class, untuk setiap jawaban yang benar maka sistem akan menghitung pointsnya dengan ditambah satu dan dilakukan perulangan untuk menciptakan soal lagi. Namun jika jawaban salah, maka program akan menampilkan output println yang mengatakan bahwa jawaban yang diinput salah dan jika total salah dilakukan sebanyak 3 kali maka program perulangan (While kondisi akan bernilai False) sehingga perulangan dihentikan. Di akhir akan ditampilkan print yang memberika tahu banyaknya soal yang dijawab dengan benar.

Penjelasan program : Jika jawaban True soal akan direfresh di public int play dan disana tiap perulangan soal yang dilakukan akan menjadi value dari int result yang berada di publin main, yang diteruskan ke if kondisi True terjadi eksekusi int points ditambah valuenya satu. Pada program public static main terdapat int result dengan valuenya ditentukan dari public int play(program yang membuat soal) dan program yang memeriksa jawaban dari user. Jika jawabn benar atau salah akan menjadi value dari int result yang nantinya akan jadi kondisi dari if statement, jika If kondisi True maka value int point ditambah satu dan jika false value wrong ditambah satu. Ketika value dari variabel wrong = 3 maka while kondisi bernilai False dan program perulangan dihentikan, lalu dikelarkan ouput println yang menyatakan banyaknya jawaban yang benar dari input user.

```
8 + 5 + 1 + 5 = 19
10 + 1 + 5 = 16
2 + 4 + 2 + 9 = 4
Wrong! The answer was 17
3 + 10 + 6 + 10 + 3 = 5
Wrong! The answer was 32
4 + 6 = 6
Wrong! The answer was 10
You earned 2 total points.
```

6. Ouput program Tugas Melempar Dadu, perulangan akan langsung berjalan karena kondisi While bernilai True karena menggunakan variabel boolean running dengan nilai True. Program akan menampilkan 2 angka acak dan hasil penjumlahannya, jika jawabannya tidak sama dengan 7 maka akan menampilkan pesan tebakkan salah dan di minta apakah ingin mengulang lempar dadu/perulangan atau tidak. Jika jawaban 'ya' maka dadu akan dilempar lagi. Jika jawaban 'tidak' maka perulangan akan berakhir dan ditampilkan pesan bahwa gagal mendapatkan angka 7 dan banyaknya perulangan yang dilakukan. Jika mendapat hasil penjumlahan 2 buah dadu nya sama dengan 7 maka akan muncul pesan selamat dan banyaknya perulangan yang dilakukan.

```
terminated> tugasmelempardadu [Java Application] C:\Users\A...
5 + 6 = 11
Tebakan Anda Salah!
Apakah Anda ingin melempar dadu lagi (ya/tidak)?
2 + 2 = 4
Tebakan Anda Salah!
Apakah Anda ingin melempar dadu lagi (ya/tidak)?
1 + 1 = 2
Tebakan Anda Salah!
Apakah Anda ingin melempar dadu lagi (ya/tidak)?
1 + 1 = 2
Tebakan Anda Salah!
Apakah Anda ingin melempar dadu lagi (ya/tidak)?
4 + 3 = 7
Selamat! Tebakan Anda Benar!
Anda menang setelah 5 percobaan!
```

```
2 + 3 = 5
Tebakan Anda Salah!
Apakah Anda ingin melempar dadu lagi (ya/tidak)?
Anda gagal menang setelah 1 percobaan.
```

E. Kesimpulan

Setelah melakukan praktikum ini kita dapat memahami dan mengaplikasikan While/ Do While dan Sentinel loop serta class random untuk membuat program yang lebih kompleks sehingga seperti membuat program interaktif seperti game penjumlahan dan lempar dadu. Menggunakan sentinel loop untuk membuat program penjumlahan. Random class untuk menghasilkan angka acak yang dapat digunakan seperti pada simulasi dadu. Menyesuaikan kondisi dari While/ Do While dan Sentinel loop yang bersifat tidak terbatas menjadi bisa digunakan dengan fleksibel dan menggabungkannya dengan if statement, scanner class, dll untuk menciptakan program yang lebih kompleks.