

Nama: Harry LBI  
NIM: A11.2023.15332  
Kelompok: A11.4405

# TUGAS PALINDROME

```
1 package palindrome;
2
3 import java.util.Scanner;
4
5 public class Palindrome {
6
7     Run | Debug
8     public static void main(String[] args) {
9         TextGame game = new TextGame();
10        Scanner scanner = new Scanner(System.in);
11
12        char isLanjut;
13        boolean lanjut;
14
15        do {
16            // Meminta input dari pengguna
17            System.out.print("Masukkan kata: ");
18            String input = scanner.nextLine();
19            // Menampilkan hasil
20            game.displayResult(input);
21
22            // Menanyakan apakah ingin lanjut
23            System.out.print("Apakah ingin lanjut (Y/N)? ");
24            isLanjut = scanner.nextLine().charAt(0);
25
26            if (isLanjut == 'Y' || isLanjut == 'y') {
27                lanjut = true;
28            } else {
29                lanjut = false;
30            }
31        }
```

Untuk membuat program lebih dinamis, fitur untuk menangkap input dari pengguna ditambahkan menggunakan kelas `Scanner`. Loop `do-while` digunakan agar program dapat meminta input secara berulang sampai pengguna memutuskan untuk berhenti. Berikut penjelasan langkah-langkahnya:

1. **Mengimpor Kelas `Scanner`:**

Sebelum digunakan, kelas `Scanner` diimpor:

```
import java.util.Scanner;
```

- 
- Kelas ini memungkinkan program membaca input dari keyboard (`System.in`).

## 2. Membuat Objek `Scanner`:

Objek `scanner` dibuat untuk membaca input pengguna:

```
Scanner scanner = new Scanner(System.in);
```

- 

## 3. Penggunaan Loop `do-while`:

```
do {  
    // Meminta input dari pengguna  
    System.out.print("Masukkan kata: ");  
    String input = scanner.nextLine();  
    // Menampilkan hasil  
    game.displayResult(input);  
  
    // Menanyakan apakah ingin lanjut  
    System.out.print("Apakah ingin lanjut (Y/N)? ");  
    isLanjut = scanner.nextLine().charAt(0);  
  
    if (isLanjut == 'Y' || isLanjut == 'y') {  
        lanjut = true;  
    } else {  
        lanjut = false;  
    }  
  
} while (lanjut);
```

Program menggunakan loop `do-while` agar pengguna dapat memasukkan teks berulang kali hingga memilih untuk berhenti:

```
do {
```

```
// Proses input
```

```
} while (lanjut);
```

- 
- Loop ini memastikan bahwa setidaknya satu kali input akan diproses, bahkan jika pengguna memutuskan untuk tidak melanjutkan setelahnya.

#### 4. Membaca Input Pengguna:

Metode `nextLine()` digunakan untuk membaca teks dari pengguna:

```
String input = scanner.nextLine();
```

- 
- Input ini diteruskan ke metode `displayResult()` untuk diproses.

#### 5. Meminta Konfirmasi Pengguna:

Setelah hasil diproses, pengguna ditanya apakah ingin melanjutkan:

```
System.out.print("Apakah ingin lanjut (Y/N)? ");
```

```
isLanjut = scanner.nextLine().charAt(0);
```

- 
- Pilihan pengguna (karakter pertama) digunakan untuk menentukan apakah loop akan diulang.

#### 6. Menentukan Kelanjutan Program:

Berdasarkan input pengguna (`Y/y` untuk lanjut, selain itu berhenti), variabel `lanjut` diatur:

```
if (isLanjut == 'Y' || isLanjut == 'y') {
```

```
    lanjut = true;
```

```
} else {  
  
    lanjut = false;  
  
}
```

○

## 7. Menutup Scanner:

Setelah pengguna memilih berhenti, scanner ditutup:

```
scanner.close();
```

Hasil dari program palindrome

```
PBO/palindrome on □ main [!?] via ↻ v23.0.2  
> java Palindrome.java  
Masukkan kata: malam  
Kata: malam adalah Palindrome  
Apakah ingin lanjut (Y/N)? y  
Masukkan kata: pagi  
Kata: pagi bukan Palindrome  
Apakah ingin lanjut (Y/N)? n  
  
PBO/palindrome on □ main [!?] via ↻ v23.0.2 took 21s  
> □
```