

Laporan Pemrograman Berorientasi Objek Dengan Bahasa Pemrograman Java

Dosen Pengampu : IMANUEL HARKESPAN, M.Kom.

Tanggal Pertemuan : 9 April 2025

Nama : Ferdian Nur Fariza
NIM : A11.2023.15074
Kelas : A11.4403
Mata Kuliah : Pemrograman Berorientasi Objek

Topik :

Tugas Praktikum 6 – Simple Game Palindrome

Tabel Deskripsi :

Pertemuan	Pembahasan Laporan	Link Dokumentasi Github
6	Palindrome.java	Link Here
	FancyTextGame.java	
	TextGame.java	
	Screenshot Hasil	

Palindrome.java

```
Palindrom.java praktek_java\week_6 M
1 package praktek_java.week_6;
2 import java.util.Scanner;
3 // Kelas Main (program utama)
4 public class Palindrome {
    Run main | Debug main | Run | Debug
5     public static void main(String[] args) {
6         Scanner scan = new Scanner(System.in);
7
8         System.out.println("===== Palindrome =====");
9         System.out.println("===== Ferdian Nur Fariza =====");
10        System.out.println("===== A11.2023.15074 =====");
11        String again = "y";
12    }
```

Pada file Palindrome.java ini adalah program main yang akan di eksekusi oleh komputer. Dan saya isi sedikit identitas saya. Karena program ini menggunakan input dari user maka menggunakan library java yaitu :

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

Di dalam program main terdapat pembuatan objek method Scanner dengan nama scan, yang nantinya scan akan dipanggil.

```
13 while(again.equalsIgnoreCase("y")){
14     System.out.print("Masukkan kata: ");
15     String input = scan.nextLine();
16     TextGame game1 = new TextGame();
17     TextGame game2 = new FancyTextGame();
18     game1.displayResult(input);
19     game2.displayResult(input);
20     System.out.print("Apakah Anda ingin mencoba lagi? (y/n): ");
21     again = scan.nextLine();
22     System.out.println();
23 }
24
```

Kondisi while disini berfungsi agar program akan tetap di eksekusi ketika user memberikan input “y” setelah program berhasil melakukan 1 iterasi dengan variabel String again sebagai wadah dari nilai input user. Di dalam kondisi while terdapat String input yang menampung

String dari user. Kemudian dilanjutkan dengan pembuatan objek game1 dan game2 dari method TextGame dan FancyTextGame.

Kemudian memanggil objek game 1 dan 2 yang akan di display dengan menggunakan method displayResult dari TextGame dan FancyTextGame berparameter input user.

Setelah system out println dijalankan, user akan menginput apakah ingin melanjutkan permainan atau tidak dengan y/n.

TextGame.java

```
TextGame.java praktek_java\week_6
1  package praktek_java.week_6;
2  import java.util.ArrayList;
3
4  public class TextGame {
5      protected ArrayList<Character> characters = new ArrayList<>();
6
7      public void inputText(String text) {
8          characters.clear();
9          for (char c : text.toCharArray()) {
10             characters.add(Character.toUpperCase(c));
11         }
12     }
13 }
```

Menggunakan package dari praktek_java/week_6/. . Menggunakan ArrayList agar memudahkan untuk membuat array tanpa harus menentukan ukuran dari array tersebut. Hal ini memudahkan dalam proses pembuatan array.

Di dalam kelas TextGame mendeklarasikan objek baru ArrayList bernama characters.

Pada method inputText berparameter String text, characters.clear() ini berfungsi agar ketika method dipanggil lagi tidak ada isinya dan mengembalikan nilai yang sesuai input user. Dilanjutkan perulangan berparameter char c : text.toCharArray() untuk mengubah hasil String input user menjadi dipecah pecah menjadi satu persatu char. Kemudian didalamnya terdapat characters.add(Character.toUpperCase(c)) yang fungsinya untuk menambahkan pecahan karakter yang sudah di Uppercase tadi ke ArrayList dengan isi char c tadi.

Overloading

```
30      // Default behavior
31      public boolean isPalindrome() {
32          return isPalindrome(this.characters);
33      }
```

Jika method isPalindrome dipanggil tanpa parameter maka akan me return isPalindrome dengan parameter ArrayList dari variabel characters.

```

14      // Overload 1 - String
15      public boolean isPalindrome(String text) {
16          inputText(text);
17          return isPalindrome();
18      }
19
20      // Overload 2 - ArrayList
21      public boolean isPalindrome(ArrayList<Character> list) {
22          for (int i = 0, j = list.size() - 1; i < j; i++, j--) {
23              if (!list.get(i).equals(list.get(j))) {
24                  return false;
25              }
26          }
27          return true;
28      }

```

Pada isPalindrome berparameter String text ini adalah memanggil method inputText berparameter text, dan mengembalikan nilai isPalindrome tanpa parameter

Kemudian untuk overloading 2 adalah isPalindrome diisi dengan parameter ArrayList, yang didalamnya i sebagai objek kiri dan j objek kanan, dengan kondisi i harus selalu kurang dari j, i++ karena objek kiri terus bergerak ke kanan dan j-- karena bergerak ke kiri. Di cek menggunakan kondisi jika character ke-i dan ke-j tidak bernilai sama akan di return false.

Jika tidak memenuhi kondisi tersebut maka isPalindrome akan mengembalikan nilai true.

```

35      public void displayResult(String word) {
36          System.out.println("Kata: " + word + (isPalindrome(word) ? " adalah Palindrome" : " bukan Palindrome"));
37      }
38  }

```

Kemudian untuk menampilkan hasil, membuat method displayResult(String word) yang di dalamnya memberikan output dari input (word) apakah palindrome atau tidak.

FancyTextGame.java

```
1 package praktik_java.week_6;
2
3 class FancyTextGame extends TextGame {
4     // Override pengecekan palindrome dengan tampilan animasi sederhana, harus menggunakan @Override
5     @Override
6     public boolean isPalindrome() {
7         System.out.print("Memeriksa kata: ");
8         for (Character c : characters) {
9             System.out.print(c + " ");
10            try {
11                Thread.sleep(200); // animasi sederhana
12            } catch (InterruptedException e) {
13                Thread.currentThread().interrupt();
14            }
15        }
16        System.out.println();
17        boolean result = super.isPalindrome();
18        System.out.println(result ? "→ Ini adalah palindrome!" : "→ Ini bukan palindrome.");
19        return result;
20    }
21 }
```

FancyTextGame adalah subclass dari TextGame yang isi methodnya adalah overriding dari isPalindrome tanpa parameter, maka dari itu harus menambahkan @Override, di dalamnya terdapat perulangan untuk mengiterasi setiap karakter di dalam array characters. Try thread sleep 200 berfungsi agar eksekusi program di tunda selama 200 ms agar seperti efek animasi.

Dan memberikan line baru.

Pada variabel boolean result merupakan superclass dari dari overriding isPalindrome dari kelas TextGame.java . hasilnya kemudian akan ditampilkan melalui variabel result, dan akhirnya akan mengembalikan nilai result.

Screenshot Hasil

```
Windows PowerShell
PS C:\Users\riza\Java_OOP> javac praktek_java/week_6/Palindrome.java
PS C:\Users\riza\Java_OOP> java praktek_java/week_6/Palindrome
===== Palindrome =====
===== Ferdian Nur Fariza =====
===== A11.2023.15074 =====
Masukkan kata: malam
Kata: malam adalah Palindrome
Memeriksa kata: M A L A M
→ Ini adalah palindrome!
Kata: malam adalah Palindrome
Apakah Anda ingin mencoba lagi? (y/n): y

Masukkan kata: PETER
Kata: PETER bukan Palindrome
Memeriksa kata: P E T E R
→ Ini bukan palindrome.
Kata: PETER bukan Palindrome
Apakah Anda ingin mencoba lagi? (y/n): y

Masukkan kata: petep
Kata: petep adalah Palindrome
Memeriksa kata: P E T E P
→ Ini adalah palindrome!
Kata: petep adalah Palindrome
Apakah Anda ingin mencoba lagi? (y/n): n

PS C:\Users\riza\Java_OOP> _
```