

MODUL

Mata Pelajaran : Pemrograman Berorientasi Objek

Kelas/Semester : XII / Ganjil

Tahun Pelajaran : 2020 / 2021

A. Judul Materi : konsep data bentuk string dan pelbagai propertinya

B. Kegiatan Belajar:

1. Menjelaskan konsep data bentuk string dan pelbagai propertinya dalam pemrograman aplikasi berorientasi obyek.
2. Menjelaskan prosedur data bentuk string dan pelbagai propertinya dalam pemrograman aplikasi berorientasi obyek
3. Menentukan penggunaan data bentuk string dan pelbagai propertinya dalam pemrograman aplikasi berorientasi obyek.
4. Menerapkan data bentuk string dan pelbagai propertinya dalam pemrograman aplikasi berorientasi obyek.
5. Merancang program aplikasi berorientasi obyek dengan penerapan data bentuk string dan pelbagai propertinya.
6. Membuat kode program aplikasi berorientasi obyek yang menerapkan data bentuk string dan pelbagai propertinya.
7. Menguji program aplikasi berorientasi obyek yang menerapkan data bentuk string dan pelbagai propertinya.

C. Kompetensi Dasar:

3.11 Menerapkan data bentuk string dan pelbagai propertinya

4.11 Membuat kode program untuk mengolah data bentuk string

D. Tujuan Pembelajaran:

1. Siswa dapat mengidentifikasi dan merumuskan masalah tentang data bentuk string dan pelbagai propertinya
2. Siswa dapat mengumpulkan data tentang data bentuk string dan pelbagai propertinya
3. Siswa dapat mengolah data data bentuk string dan pelbagai propertinya
4. Siswa mampu mengomunikasikan tentang data bentuk string dan pelbagai propertinya

E. Uraian Materi :

STRING

Dalam pemrograman Java string merupakan aspek penting, karena dapat mempelajari mengenai class dan objek melalui penggunaan string. String sebenarnya merupakan class yang terdapat dalam library Java. Java String merupakan salah satu kelas dasar yang disediakan oleh Java untuk memanipulasi karakter.

A. Membuat Objek String

Java mendefinisikan class String dalam package `java.lang.String`, sehingga tidak perlu melakukan impor secara eksplisit. Java String digunakan untuk mendefinisikan string yang konstant (tidak bisa berubah). Untuk membuat string, dapat melakukannya dengan beberapa cara, dan yang sering digunakan adalah contoh sebagai berikut.

Perhatikan kode berikut !

```
String varString='abcd';
```

Kode diatas adalah bentuk singkat dari :

```
Char[] datanya={'a','b','c','d'};
```

```
String varString= new String(datanya);
```

Jadi dapat disimpulkan bahwa String terdiri dari data array yang bertipe char, dan kita juga dapat membuat objek String dengan menggunakan keyword `new` yang biasa digunakan untuk membuat objek dari class.

Latihan 1. LatihanString.java

```
class LatihanString{  
    public static void main(String[] args) {  
        String str="Selamat Datang di Program Studi Ilmu Komputer";  
        System.out.println("Variabel Str : " + str);  
    }  
}
```

B. Menggabungkan String

Seringkali dalam pemrograman kita perlu menggabungkan String untuk mendapatkan Stringbaru. Kita dapat menggunakan operator (+) untuk menggabungkan beberapa String.

Contoh penggunaan :

Stringku="ini adalah contoh"+"penggabungan string";

Kita juga dapat menggunakan operator += untuk menggabungkan variabel String,

Misalnya seperti contoh berikut :

String kata = "Ini perkataan";

Kata += "saya sendiri";

Latihan 2. MenggabungString.java

```
class MenggabungString {  
    public static void main(String[] args) {  
        String str;  
        str="Selamat Datang "+" Mahasiswa Ilmu Komputer";  
        System.out.println(str);  
        str += "\nUniversitas Gadjah Mada";  
        System.out.println(str);  
    }  
}
```

C. Membandingkan Dua String

Untuk membandingkan dua objek String, kita dapat menggunakan fungsi sebagai berikut :

- equals(String s)

Dengan fungsi ini maka objek yang bersangkutan akan dibandingkan dengan objek string s. Pada parameter fungsi ini dengan membedakan antara huruf besar dan huruf kecil

- equalsIgnoreCase(String s)

Dengan fungsi ini maka objek yang bersangkutan akan dibandingkan dengan objek string s. Pada parameter fungsi ini tanpa membedakan antara huruf besar dan huruf kecil . Keduafungsi diatas akan menghasilkan nilai Boolean true apabila benar dan false apabila salah

Latihan 3 : String3.java

```
class MembandingkanString {  
    public static void main(String[] args) {  
        String str1 = "Ilmu";  
        String str2 = "Komputer";  
        String str3 = "IlmuKomputer";  
        System.out.println("String 1:"+str1);  
        System.out.println("String 2:"+str2);  
        System.out.println("String 3:"+str3);  
        System.out.println("String 1 = String 3 ==> "+str1.equals(str3));  
    }  
}
```

```

        System.out.println("String 2 = String 3 ==> "+str2.equals(str3));
        System.out.println("String 1 + String 2 = String 3 ==> "+(str1+str2).equals(str3));
        String str4 = "Universitas Gadjah Mada";
        String str5 = "universitas gadjah mada";
        System.out.println("String 4:"+str4);
        System.out.println("String 5:"+str5);
        System.out.println("Perintah Pembandingan dibawah mengabaikan antara huruf besar
dengan huruf kecil");
        System.out.println("String 4 = String 5 ==> " +str4.equalsIgnoreCase(str5));

System.out.println("String4+String5=String5+String4==>"+(str4+str5).equalsIgnoreCase(str5+str4));

        String str6 = "MIPA";
        String str7 = new String(str6);
        String str8 = "mipa";
        String str9 = "MIPA";
        System.out.println("String 6:"+str6);
        System.out.println("String 7:"+str7);
        System.out.println("String 8:"+str8);
        System.out.println("String 9:"+str9);
        System.out.println("Perintah Pembandingan dengan ==");
        System.out.println("String 6 = String 7 ==> "+ (str6==str7));
        System.out.println("String 6 = String 8 ==> "+ (str6==str8));
        System.out.println("String 6 = String 9 ==> "+ (str6==str9));
    }
}

```

D. Menentukan Awal Dan Akhir String

Untuk menentukan awal dan akhir String, kita dapat menggunakan dua fungsi utama, yaitu : `startsWith(String s)` Dengan fungsi ini, maka objek String yang bersangkutan akan diperiksa, apakah diawali oleh objek String s, pada parameter fungsi ini `endsWith(String s)` Dengan fungsi ini, maka objek string yang bersangkutan akan diperiksa, apakah diakhiri oleh objek string s, pada parameter fungsi ini. Fungsi diatas akan menghasilkan nilai boolean true bila benar dan false bila salah.

E. Mengurutkan String

Dapat juga melakukan pengurutan string dengan method `compareTo()`. Method ini membandingkan karakter-karakter pada String secara berurutan dari awal String. Misalnya string pertama bernilai “a” dan string kedua bernilai “b”, maka apabila `Stringpertama.compareTo(String kedua)` akan menghasilkan nilai negatif (<0) dan apabila

dilakukan sebaliknya akan menghasilkan nilai positif (>0). Nilai 0 akan dihasilkan apabila string pertama dan kedua sama.

Contoh

```
public class ST{  
    public static void main (String args[]){  
        String kata = "imam";;  
        String kata2 = "agung";;  
        String kata3 = "uding";;  
        System.out.println(kata2.compareTo(kata3));  
        System.out.println(kata.compareTo(kata2));  
    }  
}
```

F. Mendapatkan Panjang String

Kita dapat memperoleh panjang string dengan menggunakan method length(); seperti contoh berikut ini :

Listing Program

```
public class ST{  
    public static void main (String args[]){  
        String panjang = "panjang string";;  
        System.out.println(panjang.length());  
    }  
}
```

G Mencari Karakter Pada Index Tertentu

Kita bisa mencari tahu karakter apa yang ada pada index tertentu yang terdapat pada String.

Contoh:

Listing Program

```
public class ST{  
  
    public static void main (String args[]){  
  
        String str01 = "contoh string";  
  
        System.out.println(str01.charAt(0));  
  
    }  
  
}
```

H. Mencari Posisi Karakter Atau SubString Dari String Ada dua method yang dapat digunakan untuk mencari posisi karakter dari string dan dua method untuk mendapatkan posisi subString dari string.

I. Method untuk mencari posisi karakter pada String : IndexOfChar (karakter) Memerlukan argumen berupa karakter dan akan mengembalikan nilai posisi indeks dari karakter yang dicari. Posisi yang dikembalikan adalah posisi pertama dari karakter yang ditemukan. Bila karakter tidak ditemukan, maka akan mengembalikan nilai -1. IndexOf(char karakter, int indeks) sama dengan sebelumnya, tetapi memerlukan argumen tambahan, yaitu indeks posisi awal pencarian dalam integer.

Contoh:

Listing Program

```
public class ST{  
  
    public static void main (String args[]){  
  
        String str01 = "contoh string";  
  
        System.out.println(str01.indexOf("h"));  
  
        System.out.println(str01.indexOf("h", 7));  
  
    }  
  
}
```

```

    }
}

```

J. Method untuk mencari posisi subString pada String :

Method untuk mencari posisi subString pada String :

indexOf(String Str) Penggunaan dan fungsi sama dengan method untuk char.

indexOf(String str, int indeks) Penggunaan dan fungsi sama dengan method untuk char.

K. Method untuk mengubah ke huruf besar semua

Penggunaan method ini berfungsi untuk mengubah huruf pada suatu string menjadi huruf besar semua. Contoh:

Listing Program

```

public class ST{
    public static void main (String args[]){
        String kalimat = new String("saya pasti bisa");
        System.out.println(kalimat.toUpperCase());
    }
}

```

L. Method untuk mengubah ke huruf kecil semua

Penggunaan method ini berfungsi untuk mengubah huruf pada suatu string menjadi huruf kecil semua.

Contoh: Listing Program

```

public class ST{
    public static void main (String args[]){
        String kalimat = new String("SAYA PASTI BISA");
        System.out.println(kalimat.toLowerCase());
    }
}

```

M. Method untuk mengganti huruf tertentu

Penggunaan method ini berfungsi untuk mengganti char tertentu dengan char yang baru, jadi nantinya akan muncul String yang baru.

Contoh: Listing Program

```
public class ST{  
    public static void main (String args[]){  
        String kata= "malam";  
        System.out.println(kata.replace("l", "k"));  
    }  
}
```

Rangkuman

Dalam pemrograman java, String merupakan aspek penting,karena mempelajari class dan objek melalui penggunaan String. String adalah class yang terdapat dalam library Java. Java String merupakan salah satu kelas dasar yang disediakan untuk memanipulasi karakter. Definisi String pada Java yaitu java.lang.String, sehingga tidak perlu import secara eksplisit. String merupakan class yang immutable, yang berarti isi dari String tidak dapat diubah ketika sudah terbentuk.

Tugas Tugas

1 Buatlah program untuk mengurutkan kata :

Pemrograman belajar itu java menyenangkan Menjadi ***belajar pemrograman java itu menyenangkan***

2. Buatlah sebuah program dimana program berisi penggabungan 2 string,

misal: String 1 : “Bermain”

String 2 : “Bola”

Tampilan pada layar : “Bermain Bola”

Mengamati Listing Program dan Output Program

1. Tentukan nama class yang akan digunakan.
2. Tentukan variabel yang akan digunakan.
3. Tentukan method yang akan digunakan.
4. Buatlah class diagram yang menggambarkan bagian-bagian dari class yang telah ditentukan.
Contoh class diagram : Nama class Method Operasi
5. Buatlah listing programnya.
6. Lakukan kompilasi dan debug pada program.