

LAPORAN PRAKTIKUM 13
PEMROGRAMAN BERORIENTASI OBJEK
Generic



Nama : Naomi Deva Theresa Sihombing

NIM : 11S21021

Prodi : S1 Informatika

INSTITUT TEKNOLOGI DEL
SITOLUAMA, KEC.LAGUBOTI, KAB. TOBASA
TA. 2022/2023

Basic.java

```
Basic.java X
Basic.java
1  public class Basic<T> {
2      private T data;
3
4      public T getData() {
5          return data;
6      }
7
8      public void setData(T data) {
9          this.data = data;
10     }
11 }
12
```

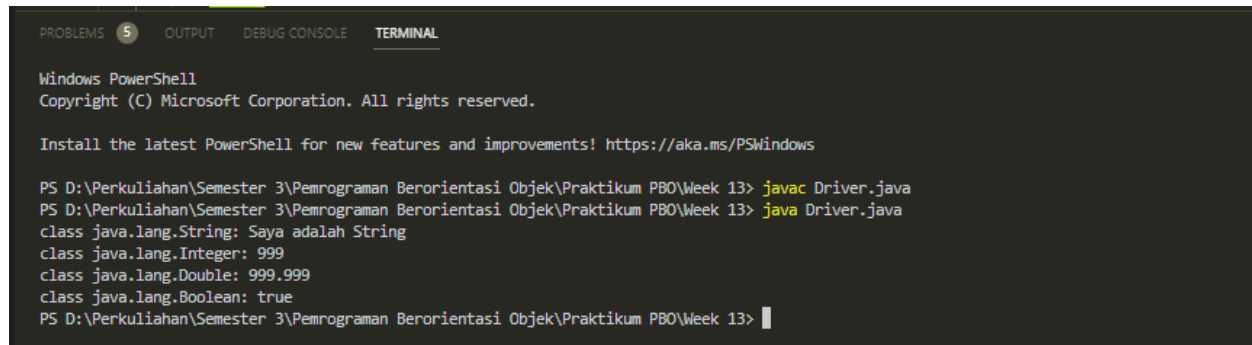
Driver.java

```
Driver.java X
Driver.java > Driver > main(String[])
1  public class Driver {
2
3      public static void main(String[] args) {
4
5          // Block 1
6          {
7              Basic<String> myData = new Basic<>();
8              myData.setData(data; "Saya adalah String");
9              System.out.println(myData.getData().getClass() + ": " + myData.getData());
10         }
11
12         // Block 2
13         {
14             Basic<Integer> myData = new Basic<>();
15             myData.setData(data; 999);
16             System.out.println(myData.getData().getClass() + ": " + myData.getData());
17         }
18
19         // Block 3
20         {
21             Basic<Double> myData = new Basic<>();
22             myData.setData(data; 999.999);
23             System.out.println(myData.getData().getClass() + ": " + myData.getData());
24         }
25
26         // Block 4
27         {
28             Basic<Boolean> myData = new Basic<>();
29             myData.setData(data; true);
30             System.out.println(myData.getData().getClass() + ": " + myData.getData());
31         }
32     }
33 }
34 }
```

Silahkan amati dan jalankan class Driver dan lihat keluarannya. Apakah hal yang sama dan berbeda dari setiap implementasi block code yang ada? Jelaskan type data apa yang dikembalikan saat pemanggilan *method* *getData()*; dan type data apa yang harus diberikan saat pemanggilan *method* *setData()*;

Jawab :

Output



```
PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Perkuliahan\Semester 3\Pemrograman Berorientasi Objek\Praktikum PBO\Week 13> javac Driver.java
PS D:\Perkuliahan\Semester 3\Pemrograman Berorientasi Objek\Praktikum PBO\Week 13> java Driver.java
class java.lang.String: Saya adalah String
class java.lang.Integer: 999
class java.lang.Double: 999.999
class java.lang.Boolean: true
PS D:\Perkuliahan\Semester 3\Pemrograman Berorientasi Objek\Praktikum PBO\Week 13> |
```

Apakah hal yang sama dan berbeda dari setiap implementasi block code yang ada? Hal yang sama pada kode program **Basic.java** adalah sama sama menggunakan *Basic myData*. Dan yang menjadi perbedaannya adalah bagian implementasi pada setiap blok. Jelaskan type data apa yang dikembalikan saat pemanggilan *method getData()*; dan type data apa yang harus diberikan saat pemanggilan *method setData()*? *method getData* berfungsi untuk mengembalikan tipe data yang sesuai begituh juga pada bagian *method setData* berfungsi untuk menyimpan tipe data yang sesuai dengan keempat block kode tersebut.

Constrain.java

```
Constrain.java X
Constrain.java > Constrain<T extends Number>
1  public class Constrain<T extends Number> {
2
3      private T data;
4
5      public T getData() {
6          return data;
7      }
8
9      public void setData(T data) {
10         this.data = data;
11     }
12
13 }
```

Driver2.java

```
Driver2.java 2 X
Driver2.java > ...
1  public class Driver2 {
2
3      Run | Debug
4      public static void main(String[] args) {
5
6          // Block 1
7          {
8              Constrain<Integer> myData = new Constrain<>();
9              myData.setData(data: 10);
10             System.out.println(myData.getData().getClass() + ": " + myData.getData());
11         }
12
13         // Block 2
14         {
15             Constrain<Double> myData = new Constrain<>();
16             myData.setData(data: 20.22);
17             System.out.println(myData.getData().getClass() + ": " + myData.getData());
18         }
19
20         // Block 3
21         {
22             Constrain<String> myData = new Constrain<>();
23             myData.setData(data: "Ini adalah string");
24             System.out.println(myData.getData().getClass() + ": " + myData.getData());
25         }
26     }
27
28 }
```

Perhatikan class Constrain apakah maksud dari potongan kode “<T extends Number?””, kemudian amati dan jalankan class Driver dan lihat keluarannya. Apakah pesan error yang tampil dan mengapa hal tersebut bisa terjadi ? kemudian lakukan jelaskan cara perbaikan sehingga kode tidak lagi mengalami error.

Jawab:

Output sebelum block 3 di hapus

```
PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Perkuliahan\Semester 3\Pemrograman Berorientasi Objek\Praktikum PBO\Week 13> javac Driver2.java
Driver2.java:21: error: type argument String is not within bounds of type-variable T
    Constrain<String> myData = new Constrain<>();
            ^
  where T is a type-variable:
    T extends Number declared in class Constrain
Driver2.java:21: error: incompatible types: cannot infer type arguments for Constrain<>
    Constrain<String> myData = new Constrain<>();
            ^
  reason: inference variable T has incompatible bounds
    equality constraints: String
    lower bounds: Number
  where T is a type-variable:
    T extends Number declared in class Constrain
2 errors
PS D:\Perkuliahan\Semester 3\Pemrograman Berorientasi Objek\Praktikum PBO\Week 13> 
```

Output setelah block 3 di hapus

```
PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Perkuliahan\Semester 3\Pemrograman Berorientasi Objek\Praktikum PBO\Week 13> javac Driver2.java
PS D:\Perkuliahan\Semester 3\Pemrograman Berorientasi Objek\Praktikum PBO\Week 13> java Driver2.java
class java.lang.Integer: 10
class java.lang.Double: 20.22
PS D:\Perkuliahan\Semester 3\Pemrograman Berorientasi Objek\Praktikum PBO\Week 13> 
```

Perhatikan class Constrain apakah maksud dari potongan kode “<T extends Number?””, kemudian amati dan jalankan class Driver dan lihat keluarannya. <T> Yang berarti Tipe Parameter yang digunakan untuk menentukan jenis tipe data yang ingin digunakan pada variable atau atribut didalamnya, T bisa kita ubah menjadi tipe data apa saja, seperti Integer, String, Boolean, Character, Float, Byte, dll. Hasil keluarannya dapat kita lihat pada gambar diatas. Apakah pesan error yang tampil dan mengapa hal tersebut bisa terjadi ? kemudian lakukan jelaskan cara perbaikan sehingga kode tidak lagi mengalami error. Yang error yaitu pada block ke 3. Dimana Class Cosntrain mengextends tipe Number yang dimana generic class nya hanya dapat bertipe Number.

Item.java (Comparable)

```
Item.java 2 x
Item.java > ...
1  public class Item<T extends Number> implements Comparable<Item> {
2
3      public String code;
4      public String name;
5      public String currency;
6      public T prices;
7      public Integer stocks;
8
9      public Item(String code, String name, String currency, T prices, Integer stock) {
10         this.code = code;
11         this.name = name;
12         this.currency = currency;
13         this.prices = prices;
14         this.stocks = stock;
15     }
16
17     @Override
18     public String toString() {
19         return code + ":" + name + "|" + currency + prices + "|stock(s):" + stocks;
20     }
21
22     @Override
23     public int compareTo(Item o) {
24         if (stocks == o.stocks) {
25             return 0;
26         } else if (stocks >= o.stocks) {
27             return 1;
28         } else {
29             return -1;
30         }
31     }
32 }
```

Driver3.java

```
Driver3.java 2 x
Driver3.java > Driver3
1  import java.util.ArrayList;
2
3  public class Driver3 {
4
5      Run | Debug
6      public static void main(String[] args) {
7
8          ArrayList<Item> items = new ArrayList<>();
9
10         items.add(new Item<Double>(code: "A001", name: "SSD 1TB", currency: "$", prices: 100.25, stock: 5));
11         items.add(new Item<Integer>(code: "A003", name: "RAM 16GB", currency: "Rp", prices: 560000, stock: 11));
12         items.add(new Item<Double>(code: "A002", name: "NVme 500GB", currency: "€", prices: 150.73, stock: 3));
13
14         for(Item i: items){
15             System.out.println(i);
16         }
17     }
18
19 }
```

Perhatikan class Item.java apakah maksud dari potongan kode Comparable <Item>”, kemudian amati dan jalankan class Driver dan lihat keluarannya.

Jawab:

Merupakan Class interface yang digunakan untuk mengurutkan objek dari class yang telah ditentukan.

Output

```
PROBLEMS 4 OUTPUT TERMINAL JUPYTER DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Asus\Downloads\Week 13\Week 13> & 'C:\Program Files\Java\jdk-17.0.2\bin\java.exe'
33747aced1b639552a3da60eb8e7bcb6\redhat.java\jdt_ws\Week 13_8310b1e9\bin' 'Driver3'
A001:SSD 1TB|$100.25|stock(s):5
A003:RAM 16GB|Rp560000|stock(s):11
A002:NVme 500GB|?150.73|stock(s):3
PS C:\Users\Asus\Downloads\Week 13\Week 13>
```

Setelah itu lakukan perubahan pada class Driver dengan kode program berikut:

```
Driver3.java 3 x
Driver3.java > Driver3 > main(String[])
1  import java.util.ArrayList;
2  import java.util.Collections;
3
4  public class Driver3 {
5
6      Run | Debug
      public static void main(String[] args) {
7
8          ArrayList<Item> items = new ArrayList<>();
9
10         items.add(new Item<Double>(code: "A001", name: "SSD 1TB", currency: "$", prices: 100.25, stock: 5));
11         items.add(new Item<Integer>(code: "A003", name: "RAM 16GB", currency: "Rp", prices: 560000, stock: 11));
12         items.add(new Item<Double>(code: "A002", name: "NVme 500GB", currency: "€", prices: 150.73, stock: 3));
13
14         Collections.sort(items);
15         for(Item i: items){
16             System.out.println(i);
17         }
18
19     }
20
21 }
22
```

Karena dalam Item tipe data yang dapat dibandingkan adalah tipe data Number.

Tantangan Praktikum

Silahkan memodifikasi kelas Item.java sehingga data yang terurut berdasarkan atribut "code". Sebagai catatan, kamu harus menggunakan Comparable untuk menyelesaikan tantangan praktikum ini.

```
Item1.java 3 X
Item1.java > ...
1 public class Item <T extends Number> implements Comparable<Item> {
2
3     public String code;
4     public String name;
5     public String currency;
6     public T prices;
7     public Integer stocks;
8     public Integer num;
9     public Integer num1;
10
11     public Item(String code, String name, String currency, T prices, Integer stock){
12         this.code = code;
13         this.name = name;
14         this.currency = currency;
15         this.prices = prices;
16         this.stocks = stock;
17         num = Integer.parseInt(code.replaceAll(regex: "[^0-9]", replacement: ""));
18     }
19
20     @Override
21     public String toString(){
22         return code + ":" + name + "|" + currency + prices + "|stocks(s):" + stocks;
23     }
24
25     @Override
26     public int compareTo(Item o){
27         num1 = Integer.parseInt(o.code.replaceAll(regex: "[^0-9]", replacement: ""));
28         if(num == num1){
29             return 0;
30         } else if(num >= num1){
31             return 1;
32         }else{
33             return -1;
34         }
35     }
36 }
```

Driver3.java

```
Item1.java 3 Driver3.java 2 X
Driver3.java > ...
1 import java.util.ArrayList;
2 //import java.util.Collections;
3
4 public class Driver3 {
5
6     Run | Debug
7     public static void main(String[] args) {
8
9         ArrayList<Item> items = new ArrayList<>();
10
11         items.add(new Item<Double>(code: "A001", name: "SSD 1TB", currency: "$", prices: 100.25, stock: 5));
12         items.add(new Item<Integer>(code: "A003", name: "RAM 16GB", currency: "Rp", prices: 560000, stock: 11));
13         items.add(new Item<Double>(code: "A002", name: "NVme 500GB", currency: "€", prices: 150.73, stock: 3));
14
15         // Collections.sort(items);
16         for(Item i: items){
17             System.out.println(i);
18         }
19     }
20
21 }
22 }
```


Maka hasil output nya adalah :

```
PROBLEMS 7 OUTPUT TERMINAL JUPYTER DEBUG CONSOLE

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Asus\Downloads\Week 13\Week 13> & 'C:\Program Files\Java\jdk-17.0.2\bin\java.exe' '-e\
User\workspaceStorage\33747aced1b639552a3da60eb8e7bcb6\redhat.java\jdt_ws\Week 13_8310b1e9\bin'
A001:SSD 1TB|$100.25|stock(s):5
A003:RAM 16GB|Rp560000|stock(s):11
A002:NVme 500GB|?150.73|stock(s):3
PS C:\Users\Asus\Downloads\Week 13\Week 13>
```