

TUGAS 9

NAMA: MOHAMMAD DWI ANSHOR AL GHIFARI

NIM: 235150607111008

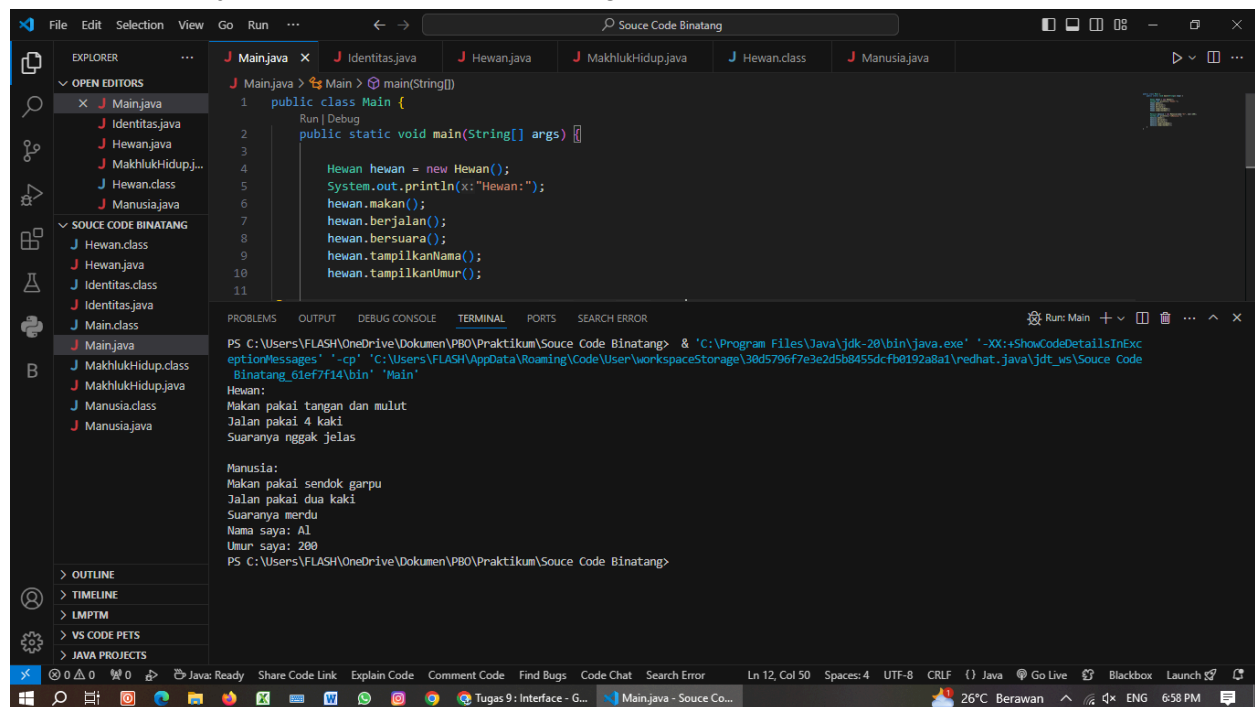
KELAS: PTI A

ASISTEN: 1. MAS DEVAN

2. MAS RIVALDO

1. Lakukan percobaan diatas dan benahi jika menemukan kesalahan serta jelaskan!

Jawab: Tidak terjadi kesalahan pada kode program di atas



The screenshot shows a Java IDE with a project named 'Source Code Binatang'. The 'Main.java' file is open, showing a class 'Main' with a 'main' method. The code creates two objects: 'Hewan' and 'Manusia', and prints their details. The output window shows the following results:

```
PS C:\Users\FLASH\OneDrive\Documents\Praktikum\Source Code Binatang> & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\FLASH\AppData\Roaming\Code\User\workspaceStorage\38d5796f7e3e2d5b845dcfb0192a8a1\redhat_java\jdt_vs\Source Code Binatang_61ef7f14\bin' 'Main'
Hewan:
Makan pakai tangan dan mulut
Jalan pakai 4 kaki
Suaranya nggak jelas

Manusia:
Makan pakai sendok garpu
Jalan pakai dua kaki
Suaranya merdu
Nama saya: Al
Umur saya: 200
PS C:\Users\FLASH\OneDrive\Documents\Praktikum\Source Code Binatang>
```

2. Apakah class yang berbentuk Interface bisa diinstansiasi menjadi sebuah objek?
Jelaskan alasannya!

Jawab: Tidak bisa diinstansiasi menjadi sebuah objek.

Alasan mengapa interface tidak dapat diinstansiasi:

- a. Abstrak yang secara alami: karena tipe sebuah abstrak hanya mendefinisikan kontrak dalam sebuah bentuk metode yang dapat diimplementasikan oleh kelas kelas yang harus mengimplementasikan interface tersebut.
- b. Tidak adanya sebuah konstruktor: Sebuah interface tidak memiliki konstruktor, karena pada kontrak sendiri merupakan sebuah metode yang khusus untuk menginisiasikan sebuah objek
- c. Tujuan dari interface tersebut: Adanya interface yaitu untuk mendefinisikan perilaku yang harus diikuti oleh kelas yang mengimplementasikannya atau dengan kata lain interface merupakan sebuah metode yang dapat mendeklarasikan tanpa menyediakan sebuah implementasinya

3. Apakah suatu class dapat mengimplementasi class interface yang jumlahnya lebih dari satu? Jelaskan alasannya!

Jawab: Benar suatu kelas dapat mengimplementasikan class dan dapat mendeklarasikan class interface yang jumlahnya lebih dari satu. Class dapat mengimplementasi lebih dari satu interface untuk mengadopsi dari berbagai perilaku yang didefinisikan oleh interface tersebut, yang memungkinkan desain yang lebih fleksibel dan modular serta mengatasi sebuah keterbatasan dari pewarisan yang ganda dalam banyak bahasa pemrograman berorientasi objek.

4. Pada interface Identitas.java hapus method tampilkan nama, amati apa yang terjadi dan mengapa demikian?

Jawab: Yang terjadi adalah error pada kode program manusia, karena

```
24  @Override
25  public void tampilkanNama() {
26      System.out.println("Nama saya: " + this.nama);
27  }
28
```

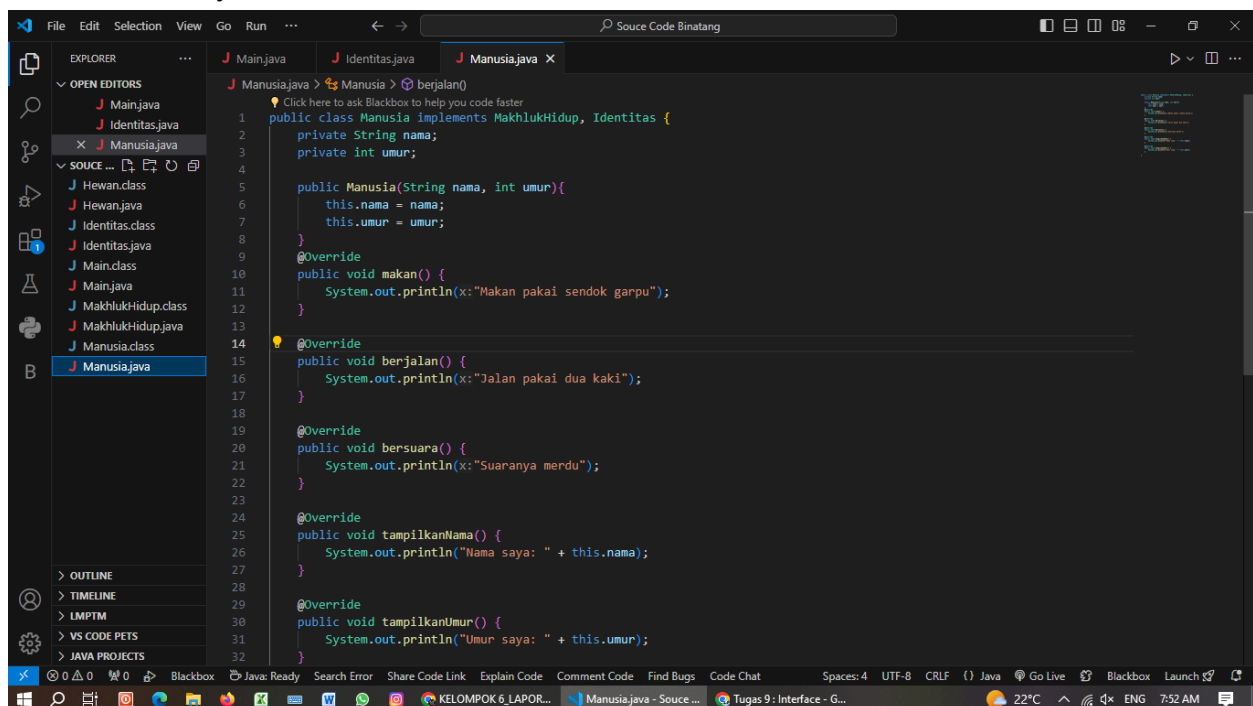
5. Jika pada class hewan kita hanya ingin mengimplements interface MakhlukHidup saja apa yang terjadi? Jelaskan

Jawab:

6. Buatlah konstruktor pada manusia dengan parameter umur dan nama kemudian panggil pada Class Main dengan menginstan objek bernama nama anda!

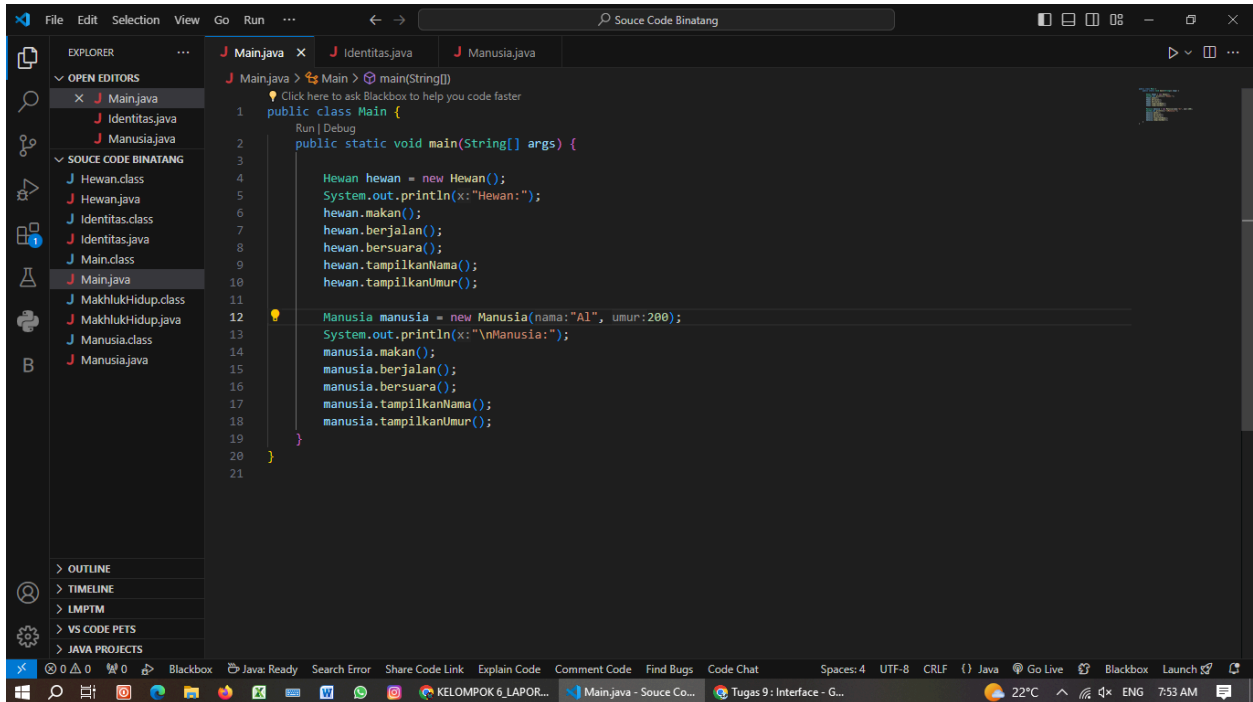
Jawab: Sudah.

a. Manusia.java



```
1  public class Manusia implements MakhlukHidup, Identitas {
2      private String nama;
3      private int umur;
4
5      public Manusia(String nama, int umur){
6          this.nama = nama;
7          this.umur = umur;
8      }
9
10     @Override
11     public void makan() {
12         System.out.println(x:"Makan pakai sendok garpu");
13     }
14
15     @Override
16     public void berjalan() {
17         System.out.println(x:"Jalan pakai dua kaki");
18     }
19
20     @Override
21     public void bersuara() {
22         System.out.println(x:"Suaranya merdu");
23     }
24
25     @Override
26     public void tampilkanNama() {
27         System.out.println("Nama saya: " + this.nama);
28     }
29
30     @Override
31     public void tampilkanUmur() {
32         System.out.println("Umur saya: " + this.umur);
33     }
34 }
```

b. Main.java



c. Hasil Run

```
PS C:\Users\FLASH\OneDrive\Dokumen\PBO\Praktikum\Souce Code Binatang> c:\cd "c:\Users\FLASH\OneDrive\Dokumen\PBO\Praktikum\Souce Code Binatang"; & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\FLASH\AppData\Roaming\Code\User\workspaceStorage\30d5796f7e3e2d5b8455dcfb0192a8a1\redhat.java\jdt_ws\Souce Code Binatang_61ef7f14\bin' 'Main'  
Hewan:  
Makan pakai tangan dan mulut  
Jalan pakai 4 kaki  
Suaranya nggak jelas  
  
Manusia:  
Makan pakai sendok garpu  
Jalan pakai dua kaki  
Suaranya merdu  
Nama saya: A1  
Umur saya: 200  
PS C:\Users\FLASH\OneDrive\Dokumen\PBO\Praktikum\Souce Code Binatang>
```

- Ubah source code diatas menjadi proses meminta inputan dari user dan buat menjadi interaktif!

Jawab:

```

1 public class Main {
2     public static void main(String[] args) {
3         Scanner scanner = new Scanner(System.in);
4     }
5 }

```

```

PS C:\Users\FLASH\OneDrive\Dokumen\PBO\Praktikum\Souce Code Binatang> c:: cd 'C:\Users\FLASH\OneDrive\Dokumen\PBO\Praktikum\Souce Code Binatang'; & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\FLASH\AppData\Roaming\Code\User\workspaceStorage\30d5796f7e3e2d5b8455dcfb0192a8a1\redhat.java\jdt_ws\Souce Code Binatang_61ef7f14\bin\' 'Main'
Masukkan nama hewan:
Paijo
Masukkan umur hewan:
3 Bulan
Hewan:
Hewan makan pakai tangan dan mulut
Hewan jalan pakai 4 kaki
Hewan suaranya nggak jelas
Nama hewan: Paijo
Umur hewan: 3

Masukkan nama manusia:
A1
Masukkan umur manusia:
20
Manusia:
Makan pakai sendok garpu
Jalan pakai dua kaki
Suaranya merdu
Nama saya: A1
Umur saya: 20
PS C:\Users\FLASH\OneDrive\Dokumen\PBO\Praktikum\Souce Code Binatang>

```

8. Buat objek selain objek diatas dengan menggunakan method yang berbeda dengan yang diatas! (min.1 contoh)

Jawab:

a. Objek baru pada code hewan dan manusia

```

1 public class Main {
2     public static void main(String[] args) {
3         Hewan hewan = new Hewan(nama:"Kucing", umur:4);
4         System.out.println(x:"Hewan:");
5         hewan.makan();
6         hewan.berjalan();
7         hewan.bersuara();
8         hewan.tampilkanNama();
9         hewan.tampilkanUmur();
10
11         // Objek baru untuk Hewan
12         Hewan burung = new Hewan(nama:"Burung", umur:2);
13         System.out.println(x:"\nHewan baru:");
14         burung.makan();
15         burung.berjalan();
16         burung.bersuara();
17         burung.tampilkanNama();
18         burung.tampilkanUmur();
19
20         Manusia manusia = new Manusia(nama:"A1", umur:20);
21         System.out.println(x:"\nManusia:");
22         manusia.makan();
23         manusia.berjalan();
24         manusia.bersuara();
25         manusia.tampilkanNama();
26         manusia.tampilkanUmur();
27
28         // Objek baru untuk Manusia
29         Manusia manusiaBaru = new Manusia(nama:"Budi", umur:25);
30         System.out.println(x:"\nManusia baru:");
31         manusiaBaru.makan();
32         manusiaBaru.berjalan();
33         manusiaBaru.bersuara();
34     }
35 }

```

b. Hasil Run

