**TUGAS 02**

**MENGUBAH PEMROGRAMAN JAVA KE DART**

**GAMES APP**



Pemrograman Berbasis Web - A

Nama :

Adelia Nurlina Putri 4521210059

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS PANCASILA**

**JAKARTA**

**2024**

1. **Public dan Private Variabel**

Pada bahasa pemrograman Dart, tidak ada parameter public dan private untuk menentukan visibility variabel. Secara default variabel pada Dart dapat diakses secara public. Untuk mengatur visibilitas menjadi private, dapat ditambahkan tanda ( \_ ) sebelum nama variabel. Contoh ( \_nama ), maka variabel “nama” hanya boleh diakses dalam class Senjata.

1. **Link Repo GitHub**

<https://github.com/adelianurlinap/PBM.git>

|  |
| --- |
| git clone <https://github.com/adelianurlinap/PBM.git>  git config --global user.name "adelianurlinap"  git config --global user.email "adelianurlinap@gmail.com" |

1. **File Senjata.dart**

|  |  |
| --- | --- |
| package app;  class Senjata {  private String nama;  private int kekuatan;  public Senjata(String nama, int kekuatan) {  this.nama = nama;  this.kekuatan = kekuatan;  }  public String getNama() {  return this.nama;  }  public int getKekuatanSerang() {  return this.kekuatan \* 2;  }  } | import 'dart:io';  class Senjata {  String nama;  int kekuatan;  Senjata(this.nama, this.kekuatan);  String getNama() {  return nama;  }  int getKekuatanSerang() {  return kekuatan \* 2;  }  } |

Senjata.dart

Senjata.java

Pada file pertama terdapat file yang berisikan class bernama “Senjata”. Class Senjata memiliki 2 variabel yaitu “nama” dan “kekuatan”.

|  |
| --- |
| class Senjata {  String nama;  int kekuatan; |

Class Senjata menggunakan konstruktor parameter posisional, hal tersebut memungkinkan untuk inisialisasi variabel instance dengan nilai yang berbeda-beda.

|  |
| --- |
| Senjata(this.nama, this.kekuatan); |

Selanjutnya terdapat getter method yang digunakan untuk memberikan return value dari variabel “nama” dan “kekuatan”. Dimana jika fungsi “getKekuatanSerang” dipanggil, akan memberikan return value (kekuatan \* 2).

|  |
| --- |
| String getNama() {  return nama;  }  int getKekuatanSerang() {  return kekuatan \* 2;  } |

1. **File Senjata.dart**

|  |  |
| --- | --- |
| package app;  class Jubah {  private String nama;  private int kekuatan;  private int kesehatan;  public Jubah(String nama, int kekuatan, int kesehatan) {  this.nama = nama;  this.kekuatan = kekuatan;  this.kesehatan = kesehatan;  }  public String getNama() {  return this.nama;  }  public int getTambahKesehatan() {  return this.kesehatan \* 10;  }  public int getNilaiKekuatan() {  return this.kekuatan \* 2;  }  } | import 'dart:io';  class Jubah {  String nama;  int kekuatan;  int kesehatan;  Jubah(this.nama, this.kekuatan, this.kesehatan);  String getNama() {  return nama;  }  int getTambahKesehatan() {  return kesehatan \* 10;  }  int getNilaiKekuatan() {  return kekuatan \* 2;  }  } |

Jubah.dart

Jubah.java

Pada file kedua terdapat file yang berisikan class bernama “Jubah”. Class Jubah memiliki 3 variabel yaitu “nama” dan “kekuatan” dan “kesehatan”.

|  |
| --- |
| class Jubah {  String nama;  int kekuatan;  int kesehatan; |

Seperti class Senjata, class Jubah juga menggunakan konstruktor parameter posisional, hal tersebut memungkinkan untuk inisialisasi variabel instance dengan nilai yang berbeda-beda.

|  |
| --- |
| Jubah(this.nama, this.kekuatan, this.kesehatan); |

Selanjutnya terdapat getter method yang digunakan untuk memberikan return value dari variabel “nama”, “kesehatan” dan “kekuatan. Dimana jika fungsi “getTambahKesehatan” dipanggil, akan memberikan return value (kesehatan \* 10) dan berlaku juga untuk fungsi “getNilaiKekuatan”.

|  |
| --- |
| String getNama() {  return nama;  }  int getTambahKesehatan() {  return kesehatan \* 10;  }  int getNilaiKekuatan() {  return kekuatan \* 2;  } |

1. **File Jagoan.dart**

Pada file ketiga terdapat file yang berisikan class bernama “Jagoan”. Dimana class Jagoan memanggil class lain yaitu Jubah dan Senjata pada yang berada pada file Jubah.dart dan Senjata.dart. Maka dari itu digunakan metode import agar file dapat saling terhubung.

|  |
| --- |
| import 'dart:io';  import 'Jubah.dart';  import 'Senjata.dart'; |

Class Jagoan memiliki 8 variabel dengan 1 String, 6 integer, dan 1 boolean. Pada class Jagoan, diinialisasikan Class Jubah menjadi jubah dan Senjata menjadi senjata. Kedua variabel tersebut diset menggunakan variabel late, agar variabel dapat diinisialisasi nanti. Dan agar terhindar dari error null (NullSafety).

|  |
| --- |
| class Jagoan {  String nama;  int kesehatanDasar;  int kekuatanDasar;  int derajat;  int totalKerusakan = 0;  int kenaikanKesehatan;  int kenaikanKekuatan;  bool hidup;  late Jubah jubah;  late Senjata senjata; |

Salah satu kelebihan Dart yaitu dapat membuat konstruktor dimana parameter diinialisasi pada baris yang sama dengan konstruktor, sedangkan pada java perlu dilakukan inisialisasi kembali pada body konstruktor.

|  |
| --- |
| Jagoan(this.nama,this.kesehatanDasar,this.kekuatanDasar,this.derajat,this.kenaikanKesehatan,this.kenaikanKekuatan,this.hidup); |

Untuk code selanjutnya pada file Jagoan.dart yaitu code fungsi getter, setter, dan void untuk membangun fungsi-fungsi yang diperlukan.

1. **File App.dart**

|  |  |
| --- | --- |
| package app;  public class App {  public static void main(String[] args) throws Exception {  Jagoan pitung = new Jagoan("Pitung");  Jubah jubahSilatPutih = new Jubah("Jubah Silat Putih", 7, 50);  Senjata golok = new Senjata("Golok", 20);  pitung.setJubah(jubahSilatPutih);  pitung.setSenjata(golok);  Jagoan jakaSembung = new Jagoan("Jaka Sembung");  Jubah jubahSilatHitam = new Jubah("Jubah Silat Hitam", 10, 50);  Senjata toya = new Senjata("Toya", 25);  jakaSembung.setJubah(jubahSilatHitam);  jakaSembung.setSenjata(toya);  pitung.info();  jakaSembung.info();  jakaSembung.menyerang(pitung);  jakaSembung.menyerang(pitung);  jakaSembung.menyerang(pitung);  jakaSembung.menyerang(pitung);    pitung.menyerang(jakaSembung);  pitung.menyerang(jakaSembung);  pitung.menyerang(jakaSembung);  pitung.menyerang(jakaSembung);  System.out.println(pitung.getNama() + ": " + pitung.getNilaiKesehatan());  System.out.println(jakaSembung.getNama() + ": " + jakaSembung.getNilaiKesehatan());  if (pitung.getStatus() && jakaSembung.getStatus() == false) {  System.out.println(pitung.getNama() + " menang cuy!");  } else if (pitung.getStatus() == false && jakaSembung.getStatus()) {  System.out.println(jakaSembung.getNama() + " menang cuy!");  } else {  if (pitung.getNilaiKesehatan() > jakaSembung.getNilaiKesehatan()) {  System.out.println(pitung.getNama() + " menang cuy!");  } else if (pitung.getNilaiKesehatan() < jakaSembung.getNilaiKesehatan()) {  System.out.println(jakaSembung.getNama() + " menang cuy!");  } else {  System.out.println("Kedua jagoan seri, cuy!");  }  }  }  }; | import 'dart:io';  import 'Jagoan.dart';  import 'Jubah.dart';  import 'Senjata.dart';  void main() {  Jagoan pitung = Jagoan("Pitung",100,100,1,10,10,true);  Jubah jubahSilatPutih = Jubah("Jubah Silat Putih", 7, 50);  Senjata golok = Senjata("Golok", 20);  pitung.setJubah(jubahSilatPutih);  pitung.setSenjata(golok);  Jagoan jakaSembung = Jagoan("Jaka Sembung",100,100,1,10,10,true);  Jubah jubahSilatHitam = Jubah("Jubah Silat Hitam", 10, 50);  Senjata toya = Senjata("Toya", 25);  jakaSembung.setJubah(jubahSilatHitam);  jakaSembung.setSenjata(toya);  pitung.info();  jakaSembung.info();  jakaSembung.menyerang(pitung);  jakaSembung.menyerang(pitung);  jakaSembung.menyerang(pitung);  jakaSembung.menyerang(pitung);  pitung.menyerang(jakaSembung);  pitung.menyerang(jakaSembung);  pitung.menyerang(jakaSembung);  pitung.menyerang(jakaSembung);  print("${pitung.nama}: ${pitung.getNilaiKesehatan()}");  print("${jakaSembung.nama}: ${jakaSembung.getNilaiKesehatan()}");  if (pitung.getStatus() && !jakaSembung.getStatus()) {  print("${pitung.nama} Menang cuy!");  } else if (!pitung.getStatus() && jakaSembung.getStatus()) {  print("${jakaSembung.nama} Menang cuy!");  } else {  if (pitung.getNilaiKesehatan() > jakaSembung.getNilaiKesehatan()) {  print("${pitung.nama} Menang cuy!");  } else if (pitung.getNilaiKesehatan() < jakaSembung.getNilaiKesehatan()) {  print("${jakaSembung.nama} Menang cuy!");  } else {  print("Kedua jagoan seri, cuy!");  }  }  } |

App.dart

App.java

File terakhir yaitu file App.dart yang berisikan fungsi void main(). Fungsi main akan memanggil seluruh class yaitu Jagoan, Jubah dan Senjata. Maka dari itu digunakan metode import agar file dapat saling terhubung.

|  |
| --- |
| import 'dart:io';  import 'Jagoan.dart';  import 'Jubah.dart';  import 'Senjata.dart'; |

Pada fungsi main dilakukan inisialisasi dari variabel yang berada pada class sebelumnya. Karna menggunakan konstruktor berparameter bukan default, maka inisialisasi dapat dilakukan pada file App.dart.

|  |
| --- |
| void main() {  Jagoan pitung = Jagoan("Pitung",100,100,1,10,10,true);  Jubah jubahSilatPutih = Jubah("Jubah Silat Putih", 7, 50);  Senjata golok = Senjata("Golok", 20);  pitung.setJubah(jubahSilatPutih);  pitung.setSenjata(golok);  Jagoan jakaSembung = Jagoan("Jaka Sembung",100,100,1,10,10,true);  Jubah jubahSilatHitam = Jubah("Jubah Silat Hitam", 10, 50);  Senjata toya = Senjata("Toya", 25); |

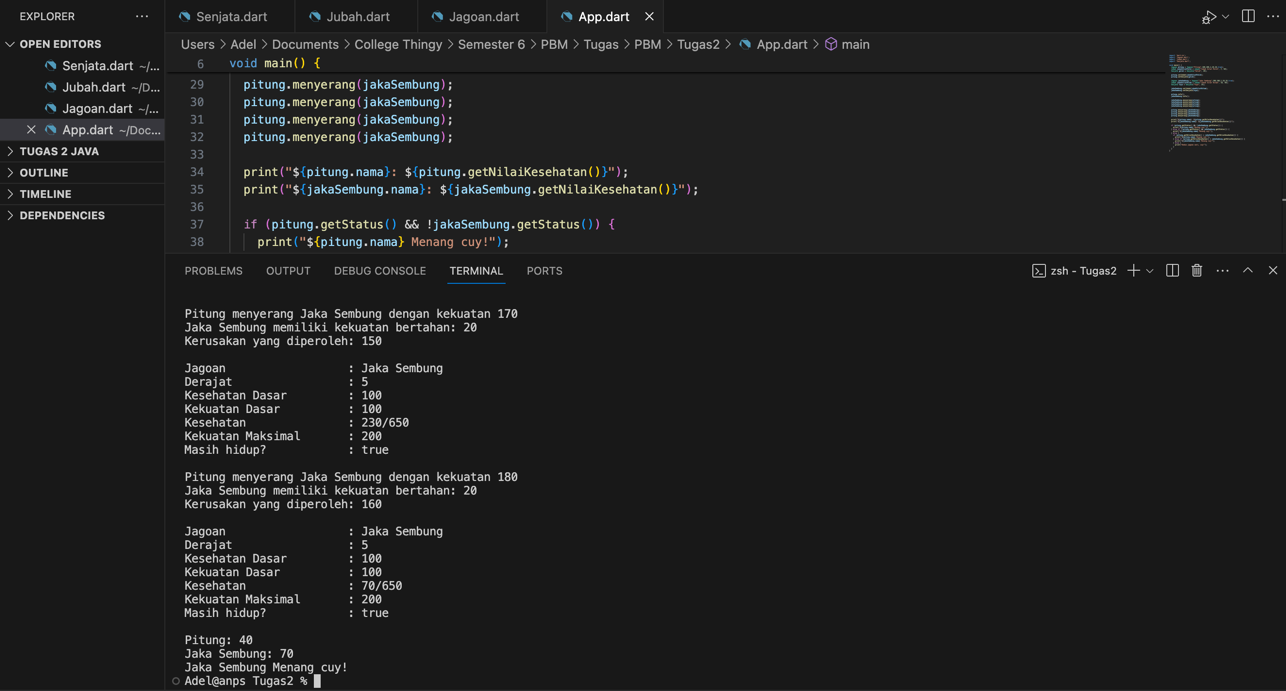
Perbedaan selanjutnya antara Java dan Dart yaitu pada Dart untuk mencetak value dapat menggunakan parameter print() berbeda dengan Java yang menggunakan System.out.println().

|  |
| --- |
| print("${pitung.nama}: ${pitung.getNilaiKesehatan()}");  print("${jakaSembung.nama}: ${jakaSembung.getNilaiKesehatan()}");  if (pitung.getStatus() && !jakaSembung.getStatus()) {  print("${pitung.nama} Menang cuy!");  } else if (!pitung.getStatus() && jakaSembung.getStatus()) {  print("${jakaSembung.nama} Menang cuy!");  } else {  if (pitung.getNilaiKesehatan() > jakaSembung.getNilaiKesehatan()) {  print("${pitung.nama} Menang cuy!");  } else if (pitung.getNilaiKesehatan() < jakaSembung.getNilaiKesehatan()) {  print("${jakaSembung.nama} Menang cuy!");  } else {  print("Kedua jagoan seri, cuy!");  }  } |

1. **Run Program**

Dikarenakan fungsi void main() berada pada file App.dart, maka file yang dijalankan yaitu App.dart.

|  |
| --- |
| source $HOME/.zshrc  dart App.dart |



1. **Push Code to Github**

Dikarenakan fungsi void main() berada pada file App.dart, maka file yang dijalankan yaitu App.dart.

|  |
| --- |
| git remote -v  git add .  git commit -M "Tugas 2 ver Final"  git push  git status |