

Object Oriented Programming Encapsulation



Name

Muhammad Baihaqi Aulia Asy'ari

NIM

2241720145

Class

2I

Department

Information Technology

Study Program

D4 Informatics Engineering

3 Experiment

3.1 Experiment 1 - Encapsulation

```
1 package Experiment1;
2
3 public class Motor {
4     public int velocity = 0;
5     public boolean ignitionOn = false;
6
7     public void printStatus() {
8         if (ignitionOn == true) {
9             System.out.println("Ignition On");
10        } else {
11            System.out.println("Ignition Off");
12        }
13        System.out.println("Velocity " + velocity + "\n");
14    }
15 }
```

```
1 package Experiment1;
2
3 public class MotorDemo {
4     public static void main(String[] args) {
5         Motor motor = new Motor();
6         motor.printStatus();
7         motor.velocity = 50;
8         motor.printStatus();
9     }
10 }
```

```
1 PS D:\Kuliah> & 'C:\Program Files\Java\jdk-18.0.2.1\bin\java.exe'
   ↳ '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
   ↳ 'C:\Users\G4CE-PC\AppData\Roaming\Code\User\workspaceStorage\
   ↳ 80d97a47d24665dc0bce7ab1e048ecbd\redhat.java\jdt_ws\
   ↳ Kuliah_28156aa7\bin' 'Experiment1.MotorDemo'
2 Ignition Off
3 Velocity 0
4
5 Ignition Off
6 Velocity 50
```

3.2 Experiment 2 - Access Modifier

```
1 package Experiment2;
2
3 public class Motor {
4     private int velocity = 0;
5     private boolean ignitionOn = false;
6
7     public void turnIgnitionOn() {
8         ignitionOn = true;
9     }
10
11    public void turnIgnitionOff() {
12        ignitionOn = false;
13        velocity = 0;
14    }
15
16    public void increaseSpeed() {
17        if (ignitionOn == true) {
18            velocity += 5;
19        } else {
20            System.out.println("Can't increase the velocity because
21                               ↳ the ignition is off\n");
22        }
23    }
24
25    public void decreaseSpeed() {
26        if (ignitionOn == true) {
27            velocity -= 5;
28        } else {
29            System.out.println("Can't decrease the velocity because
30                               ↳ the ignition is off\n");
31        }
32    }
33
34    public void printStatus() {
35        if (ignitionOn == true) {
36            System.out.println("Ignition On");
37        } else {
38            System.out.println("Ignition Off");
39        }
40        System.out.println("Velocity " + velocity + "\n");
41    }
42 }
```

```

40 }
41
1  package Experiment2;
2
3  public class MotorDemo {
4      public static void main(String[] args) {
5          Motor motor = new Motor();
6          motor.printStatus();
7          motor.increaseSpeed();
8
9          motor.turnIgnitionOn();
10         motor.printStatus();
11
12         motor.increaseSpeed();
13         motor.printStatus();
14
15         motor.increaseSpeed();
16         motor.printStatus();
17
18         motor.increaseSpeed();
19         motor.printStatus();
20
21         motor.turnIgnitionOff();
22         motor.printStatus();
23     }
24 }
25

```

```

1  PS D:\Kuliah> d:; cd 'd:\Kuliah'; & 'C:\Program
   ↳ Files\Java\jdk-18.0.2.1\bin\java.exe'
   ↳ '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
   ↳ 'C:\Users\G4CE-PC\AppData\Roaming\Code\User\workspaceStorage\
   ↳ 80d97a47d24665dc0bce7ab1e048ecbd\redhat.java\jdt_ws\
   ↳ Kuliah_28156aa7\bin' 'Experiment2.MotorDemo'
2  Ignition Off
3  Velocity 0
4
5  Can't increase the velocity because the ignition is off
6
7  Ignition On
8  Velocity 0
9

```

```
10 Ignition On
11 Velocity 5
12
13 Ignition On
14 Velocity 10
15
16 Ignition On
17 Velocity 15
18
19 Ignition Off
20 Velocity 0
21
```

3.3 Question

1. Pada class TestMobil, saat kita menambah kecepatan untuk pertama kalinya, mengapa muncul peringatan “Kecepatan tidak bisa bertambah karena Mesin Off!”?

Answer: Because increasing velocity has the requirement for the ignition to be On before the velocity can be increase

2. Mengapa atribut kecepatan dan kontakOn diset private?

Answer: So as not to be improperly manipulate both variables

3. Ubah class Motor sehingga kecepatan maksimalnya adalah 100!

```
1  public void increaseSpeed() {
2      if (ignitionOn == true) {
3          if (velocity == 100) {
4              System.out.println("Maximum velocity has been
5                  ↪ reached");
6          } else {
7              velocity += 5;
8          }
9      } else {
10         System.out.println("Can't increase the velocity because
11             ↪ the ignition is off\n");
12     }
13 }
```

3.4 Experiment 3 - Getter and Setter

```
1 package Experiment3;
2
3 public class Member {
4     private String name;
5     private String address;
6     private float balance;
7
8     public void setName(String name) {
9         this.name = name;
10    }
11
12    public void setAddress(String address) {
13        this.address = address;
14    }
15
16    public String getName() {
17        return name;
18    }
19
20    public String getAddress() {
21        return address;
22    }
23
24    public float getBalance() {
25        return balance;
26    }
27
28    public void deposit(float cash) {
29        balance += cash;
30    }
31
32    public void withdraw(float cash) {
33        balance -= cash;
34    }
35 }
36
37
38 package Experiment3;
39
40 public class CooperativeDemo {
41     public static void main(String[] args) {
```

```

5      Member member1 = new Member();
6      member1.setName("Iwan Setiawan");
7      member1.setAddress("Jalan Sukarno Hatta no 10");
8      member1.deposit(100_000);
9      System.out.println("Balance " + member1.getName() + " : Rp " +
    ↪ member1.getBalance());
10
11     member1.withdraw(5_000);
12     System.out.println("Balance " + member1.getName() + " : Rp " +
    ↪ member1.getBalance());
13 }
14 }
15

```

```

1 PS D:\Kuliah> d:; cd 'd:\Kuliah'; & 'C:\Program
    ↪ Files\Java\jdk-18.0.2.1\bin\java.exe'
    ↪ '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
    ↪ 'C:\Users\G4CE-PC\AppData\Roaming\Code\User\workspaceStorage\
    ↪ 80d97a47d24665dc0bce7ab1e048ecbd\redhat.java\jdt_ws\
    ↪ Kuliah_28156aa7\bin' 'Experiment3.CooperativeDemo'
2 Balance Iwan Setiawan : Rp 100000.0
3 Balance Iwan Setiawan : Rp 95000.0

```

3.5 Experiment 4 - Constructor, Instantiation

```
1 package Experiment4;
2
3 public class CooperativeDemo {
4     public static void main(String[] args) {
5         Member member1 = new Member();
6         System.out.println("Balance " + member1.getName() + " : Rp " +
7             ↳ member1.getBalance());
8
9         member1.setName("Iwan Setiawan");
10        member1.setAddress("Jalan Sukarno Hatta no 10");
11        member1.deposit(100_000);
12        System.out.println("Balance " + member1.getName() + " : Rp " +
13            ↳ member1.getBalance());
14
15        member1.withdraw(5_000);
16        System.out.println("Balance " + member1.getName() + " : Rp " +
17            ↳ member1.getBalance());
18    }
19 }
```

```
1 PS D:\Kuliah> d:; cd 'd:\Kuliah'; & 'C:\Program
   ↳ Files\Java\jdk-18.0.2.1\bin\java.exe'
   ↳ '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
   ↳ 'C:\Users\G4CE-PC\AppData\Roaming\Code\User\workspaceStorage\
   ↳ 80d97a47d24665dc0bce7ab1e048ecbd\redhat.java\jdt_ws\
   ↳ Kuliah_28156aa7\bin' 'Experiment4.CooperativeDemo'
2 Balance null : Rp 0.0
3 Balance Iwan Setiawan : Rp 100000.0
4 Balance Iwan Setiawan : Rp 95000.0
```

```
1 package Experiment4;
2
3 public class Member {
4     private String name;
5     private String address;
6     private float balance;
7
8     Member(String name, String address) {
9         this.name = name;
10        this.address = address;
```

```

11         this.balance = 0;
12     }
13
14     public void setName(String name) {
15         this.name = name;
16     }
17
18     public void setAddress(String address) {
19         this.address = address;
20     }
21
22     public String getName() {
23         return name;
24     }
25
26     public String getAddress() {
27         return address;
28     }
29
30     public float getBalance() {
31         return balance;
32     }
33
34     public void deposit(float cash) {
35         balance += cash;
36     }
37
38     public void withdraw(float cash) {
39         balance -= cash;
40     }
41 }
42
43 package Experiment4;
44
45 public class CooperativeDemo {
46     public static void main(String[] args) {
47         Member member1 = new Member("Iwan", "Jalan Mawar");
48         System.out.println("Balance " + member1.getName() + " : Rp " +
49             ↪ member1.getBalance());
50
51         member1.setName("Iwan Setiawan");
52         member1.setAddress("Jalan Sukarno Hatta no 10");
53     }
54 }

```

```

10     member1.deposit(100_000);
11     System.out.println("Balance " + member1.getName() + " : Rp " +
        ↳ member1.getBalance());
12
13     member1.withdraw(5_000);
14     System.out.println("Balance " + member1.getName() + " : Rp " +
        ↳ member1.getBalance());
15 }
16 }
17

```

```

1 PS D:\Kuliah> d:; cd 'd:\Kuliah'; & 'C:\Program
    ↳ Files\Java\jdk-18.0.2.1\bin\java.exe'
    ↳ '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'
    ↳ 'C:\Users\G4CE-PC\AppData\Roaming\Code\User\workspaceStorage\
    ↳ 80d97a47d24665dc0bce7ab1e048ecbd\redhat.java\jdt_ws\
    ↳ Kuliah_28156aa7\bin' 'Experiment4.CooperativeDemo'
2 Balance Iwan : Rp 0.0
3 Balance Iwan Setiawan : Rp 100000.0
4 Balance Iwan Setiawan : Rp 95000.0

```

3.6 Question - Experiment 3 and 4

1. Apa yang dimaksud getter dan setter?
Answer: getter and setter are methods inside of a class use to access an attribute of a class.
2. Apa kegunaan dari method getSimpanan()?
Answer: the method is used to get the value of balance attribute inside of the object.
3. Method apa yang digunakan untk menambah saldo?
Answer: the deposit()/setor() is used to add the balance of the object
4. Apa yang dimaksud konstruktor?
Answer: constructor is a way to set the value of attributes inside of an object while instantiating it.
5. Sebutkan aturan dalam membuat konstruktor?
Answer: the name of the constructor must be the same as its class name, constructor method can't have a return type
6. Apakah boleh konstruktor bertipe private?
Answer: in theory a constructor could have private modifier as it is technically a method specialized to instantiate an object.
7. Kapan menggunakan parameter dengan passing parameter?
Answer: it is used when the object attributes need to be set right before it is being used.
8. Apa perbedaan atribut class dan instansiasi atribut?
Answer: class attribute can be used among other class, while attribute instantiation is an attribute that can be use by the instance of an object.
9. Apa perbedaan class method dan instansiasi method?
Answer: same as the previous question, the former can be used class wide while the later only can be use by the instance of the object.

5 Assignment

1. Cobalah program dibawah ini dan tuliskan hasil outputnya

```
1  package Assignment;
2
3  public class EncapDemo {
4      private String name;
5      private int age;
6
7      public String getName() {
8          return name;
9      }
10
11     public void setName(String newName) {
12         name = newName;
13     }
14
15     public int getAge() {
16         return age;
17     }
18
19     public void setAge(int newAge) {
20         if (newAge > 30) {
21             age = 30;
22         } else {
23             age = newAge;
24         }
25     }
26 }
27
28
29 package Assignment;
30
31 public class EncapTest {
32     public static void main(String[] args) {
33         EncapDemo encap = new EncapDemo();
34         encap.setName("James");
35         encap.setAge(35);
36
37         System.out.println("Name : " + encap.getName());
38         System.out.println("Age : " + encap.getAge());
39     }
40 }
```

```
12 }  
13
```

Answer:

```
1 PS D:\Kuliah> & 'C:\Program  
  ↪ Files\Java\jdk-18.0.2.1\bin\java.exe'  
  ↪ '-XX:+ShowCodeDetailsInExceptionMessages' '-cp'  
  ↪ 'C:\Users\G4CE-PC\AppData\Roaming\Code\User\workspaceStorage\  
  ↪ 80d97a47d24665dc0bce7ab1e048ecbd\redhat.java\jdt_ws\  
  ↪ Kuliah_28156aa7\bin' 'Assignment.EncapTest'  
2 Name : James  
3 Age : 30
```

2. Pada program diatas, pada class EncapTest kita mengeset age dengan nilai 35, namun pada saat ditampilkan ke layar nilainya 30, jelaskan mengapa.

Answer: because in the setAge method in the if else condition, the age set into 30 if the newAge value is over 30.

3. Ubah program diatas agar atribut age dapat diberi nilai maksimal 30 dan minimal 18.

Answer:

```
1 public void setAge(int newAge) {  
2     if (newAge > 30) {  
3         age = 30;  
4     } else if (newAge < 18) {  
5         age = 18;  
6     } else {  
7         age = newAge;  
8     }  
9 }
```

4. Pada sebuah sistem informasi koperasi simpan pinjam, terdapat class Anggota yang memiliki atribut antara lain nomor KTP, nama, limit peminjaman, dan jumlah pinjaman. Anggota dapat meminjam uang dengan batas limit peminjaman yang ditentukan. Anggota juga dapat mengangsur pinjaman. Ketika Anggota tersebut mengangsur pinjaman, maka jumlah pinjaman akan berkurang sesuai dengan nominal yang diangsur. Buatlah class Anggota tersebut, berikan atribut, method dan konstruktor sesuai dengan kebutuhan. Uji dengan TestKoperasi berikut ini untuk memeriksa apakah class Anggota yang anda buat telah sesuai dengan yang diharapkan.

```
1 package Assignment;  
2
```

```

3  public class TestKoperasi{
4      public static void main(String[] args) {
5          Anggota donny = new Anggota("111333444", "Donny",
6              ↳ 5_000_000);
7
8          System.out.println("Nama Anggota: " + donny.getNama());
9          System.out.println("Limit Pinjaman: " +
10             ↳ donny.getLimitPinjaman());
11
12         System.out.println("\nMeminjam uang 10.000.000...");
13         donny.pinjam(10_000_000);
14         System.out.println("Jumlah pinjaman saat ini: " +
15             ↳ donny.getJumlahPinjaman());
16
17         System.out.println("\nMeminjam uang 4.000.000...");
18         donny.pinjam(4_000_000);
19         System.out.println("Jumlah pinjaman saat ini: " +
20             ↳ donny.getJumlahPinjaman());
21
22         System.out.println("\nMembayar angsuran 1.000.000");
23         donny.angsur(1_000_000);
24         System.out.println("Jumlah pinjaman saat ini: " +
25             ↳ donny.getJumlahPinjaman());
26
27         System.out.println("\nMembayar angsuran 3.000.000");
28         donny.angsur(3000000);
29         System.out.println("Jumlah pinjaman saat ini: " +
30             ↳ donny.getJumlahPinjaman());
31     }
32 }

```

Answer:

```

1  package Assignment;
2
3  public class Anggota {
4      private String noKTP;
5      private String nama;
6      private int limitPinjaman;
7      private int jumlahPinjaman;
8
9      Anggota(String noKTP, String nama, int limitPinjaman) {

```

```

10         this.noKTP = noKTP;
11         this.nama = nama;
12         this.limitPinjaman = limitPinjaman;
13         this.jumlahPinjaman = 0;
14     }
15
16     public String getNoKTP() {
17         return noKTP;
18     }
19
20     public String getNama() {
21         return nama;
22     }
23
24     public int getLimitPinjaman() {
25         return limitPinjaman;
26     }
27
28     public int getJumlahPinjaman() {
29         return jumlahPinjaman;
30     }
31
32     public void pinjam(int uang) {
33         if (uang > limitPinjaman) {
34             System.out.println("Maaf, jumlah pinjaman melebihi
35                 ↳ limit");
36         } else {
37             jumlahPinjaman += uang;
38         }
39     }
40
41     public void angsur(int uang) {
42         jumlahPinjaman -= uang;
43     }
44

```

5. Modifikasi soal no. 4 agar nominal yang dapat diangsur minimal adalah 10% dari jumlah pinjaman saat ini. Jika mengangsur kurang dari itu, maka muncul peringatan “Maaf, angsuran harus 10% dari jumlah pinjaman”.

Answer:

```

1 public void angsur(int uang) {

```

```

2      if (uang < jumlahPinjaman * 0.1) {
3          System.out.println("Maaf, angsuran harus 10% dari jumlah
           ↳ pinjaman");
4      } else {
5          jumlahPinjaman -= uang;
6      }
7  }

```

6. Modifikasi class TestKoperasi, agar jumlah pinjaman dan angsuran dapat menerima input dari console.

Answer:

```

1  package Assignment;
2
3  import java.util.Scanner;
4
5  public class TestKoperasi{
6      static Scanner input = new Scanner(System.in);
7      static Anggota donny = new Anggota("111333444", "Donny",
           ↳ 5_000_000);
8      public static void main(String[] args) {
9
10         System.out.println("Nama Anggota: " + donny.getNama());
11         System.out.println("Limit Pinjaman: " +
           ↳ donny.getLimitPinjaman());
12
13         System.out.println("\nMeminjam uang 10.000.000...");
14         donny.pinjam(10_000_000);
15         System.out.println("Jumlah pinjaman saat ini: " +
           ↳ donny.getJumlahPinjaman());
16
17         System.out.println("\nMeminjam uang 4.000.000...");
18         donny.pinjam(4_000_000);
19         System.out.println("Jumlah pinjaman saat ini: " +
           ↳ donny.getJumlahPinjaman());
20
21         System.out.println("\nMembayar angsuran 1.000.000");
22         donny.angsur(1_000_000);
23         System.out.println("Jumlah pinjaman saat ini: " +
           ↳ donny.getJumlahPinjaman());
24
25         System.out.println("\nMembayar angsuran 3.000.000");
26         donny.angsur(3000000);

```

```

27         System.out.println("Jumlah pinjaman saat ini: " +
    ↪     donny.getJumlahPinjaman());
28
29         goToMenu();
30
31         input.close();
32     }
33
34     public static void goToMenu() {
35         System.out.println("Menu Transaksi Koperasi");
36         System.out.println("1. Pinjam");
37         System.out.println("2. Angsur");
38         System.out.print("Menu: ");
39         int menu = input.nextInt();
40         switch (menu) {
41             case 1:
42                 goToPinjam();
43                 break;
44
45             case 2:
46                 goToAngsur();
47                 break;
48
49             default:
50                 System.out.println("Tolong masukan nilai yang
    ↪     valid");
51                 goToMenu();
52                 break;
53         }
54     }
55
56     public static void goToPinjam() {
57         System.out.print("\nMasukkan jumlah yang ingin dipinjam :
    ↪     ");
58         donny.pinjam(input.nextInt());
59         System.out.println("Jumlah pinjaman saat ini: " +
    ↪     donny.getJumlahPinjaman());
60         goToMenu();
61     }
62
63     public static void goToAngsur() {

```

```
64         System.out.print("\nMasukkan jumlah yang ingin diangsur :  
        ↳ ");  
65         donny.angsur(input.nextInt());  
66         System.out.println("Jumlah pinjaman saat ini: " +  
        ↳ donny.getJumlahPinjaman());  
67         goToMenu();  
68     }  
69 }  
70
```