STT-NF

Pengenalan Pola Desain Perangkat Lunak & Pemrograman Berorientasi Objek

Tifani Nabarian, S.Kom., M.T.I.
tifany.nabarian@dosen.nurulfikri.ac.id

Profil Dosen

Riwayat Pendidikan:

- ✓ S1 – Teknik Informatika: UNSRI
- ✓ S2 – Magister Teknologi Informasi: UI

Riwayat Perkerjaan:

PT. BANK NEGARA INDONESIA (Persero) Tbk.
5 thn 4 bln

Channel System & SOA Analyst
Nov 2015 – Feb 2018 · 2 thn 4 bln
IT Solution & Security Division, DKI Jakarta - Indonesia

Switching System & Middleware Programmer
Mar 2015 – Okt 2015 · 8 bln
IT Division, DKI Jakarta - Indonesia

Channel & Connectivity Programmer
Nov 2013 – Feb 2015 · 1 thn 4 bln
IT Division, DKI Jakarta - Indonesia

ODP Trainee Batch 108, Specific Banking
Nov 2012 – Okt 2013 · 1 thn
BNI Corporate University, DKI Jakarta - Indonesia

Kontrak Perkuliahan

Mata Kuliah : Pola Desain Perangkat Lunak
Kode : KMTI21112
SKS : 3 SKS

Rencana Perkuliahan

- ✓ Pemaparan Materi via Video
- ✓ Diskusi/Tanya Jawab via Forum
- ✓ Tugas (Individu/Kelompok)
- ✓ Praktikum Mandiri

Komponen Penilaian:

UTS	: 25%
UAS	: 25%
Quiz	: 20%
Tugas	: 25%
Kehadiran	: 5%

Adab Menuntut Ilmu

“Pelajarilah adab sebelum mempelajari suatu ilmu.”

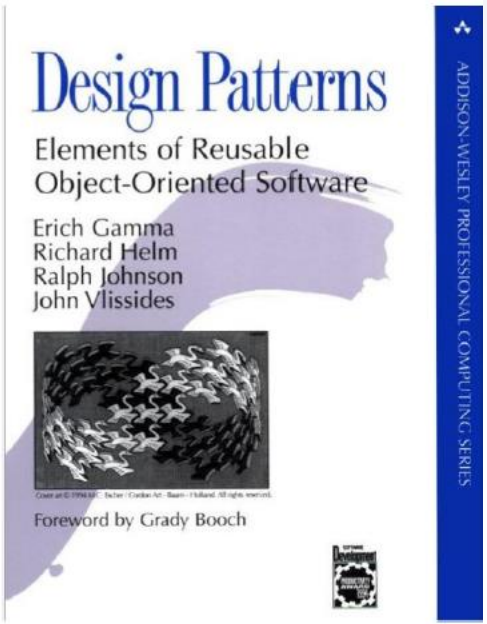
Kenapa sampai para ulama mendahulukan mempelajari adab? Sebagaimana Yusuf bin Al Husain berkata,

بالأدب تفهم العلم

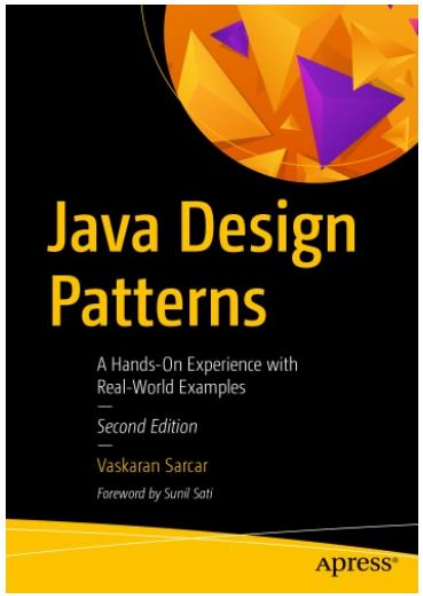
“Dengan mempelajari adab, maka engkau jadi mudah memahami ilmu.”

Source:
https://rumaysho.com/7199-banyak-ilmu-namun-lupa-belajar-adab-dan-akhlak.html

References



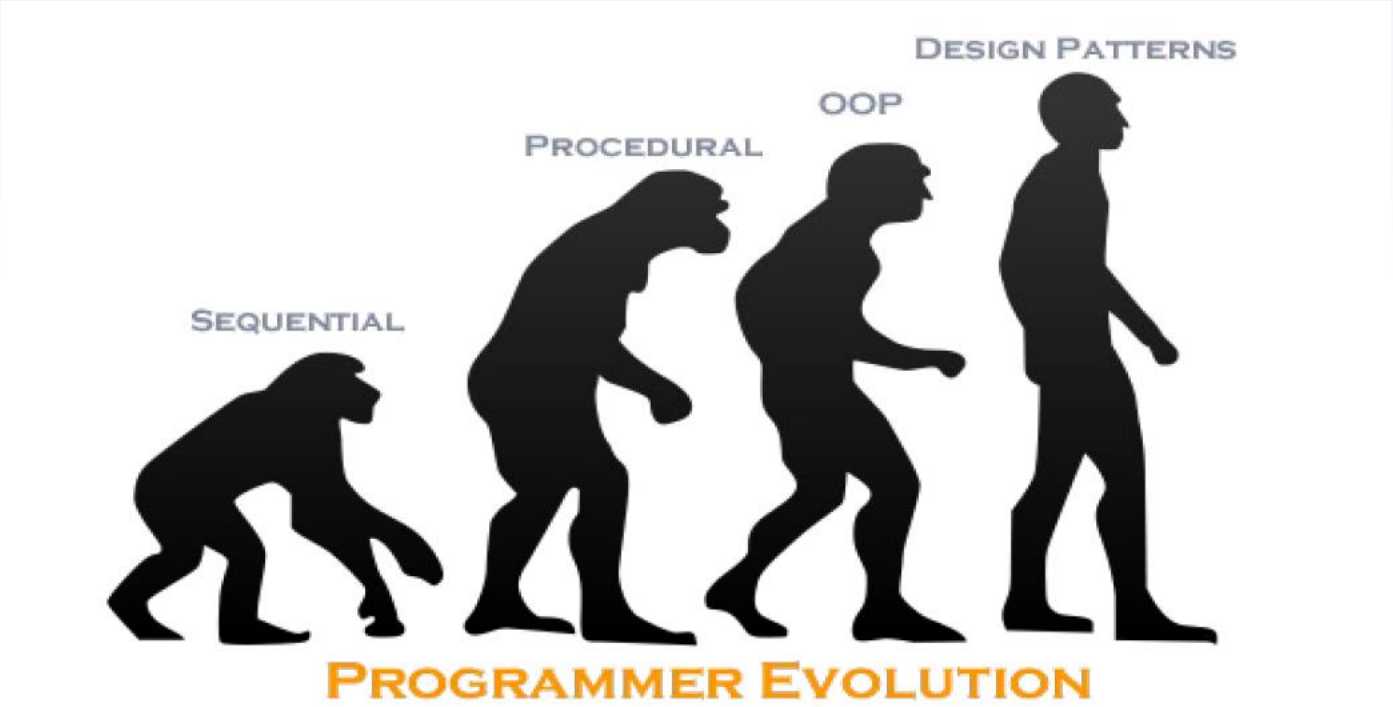
Erich Gamma, Design Pattern Elemen of Reusable Object Oriented Software, Addison-Wesley, 1994



Sarcar, Verkaran. Java Design Patterns: A Hands-On Experience with Real-World Examples, Apress, 2019

Pola Desain Perangkat Lunak

Latar Belakang



Latar Belakang (1)

What's wrong with sequential & procedural programming?

```php
<?php

$firstNumber=20;
$secondNumber=40;
$total= $firstNumber + $secondNumber;
echo $total;

?>
```

```php
<?php

function addEmUp($first,$second)
{
    $total=$first + $second;
    echo $total;
}
addEmUp(20,40);

?>
```

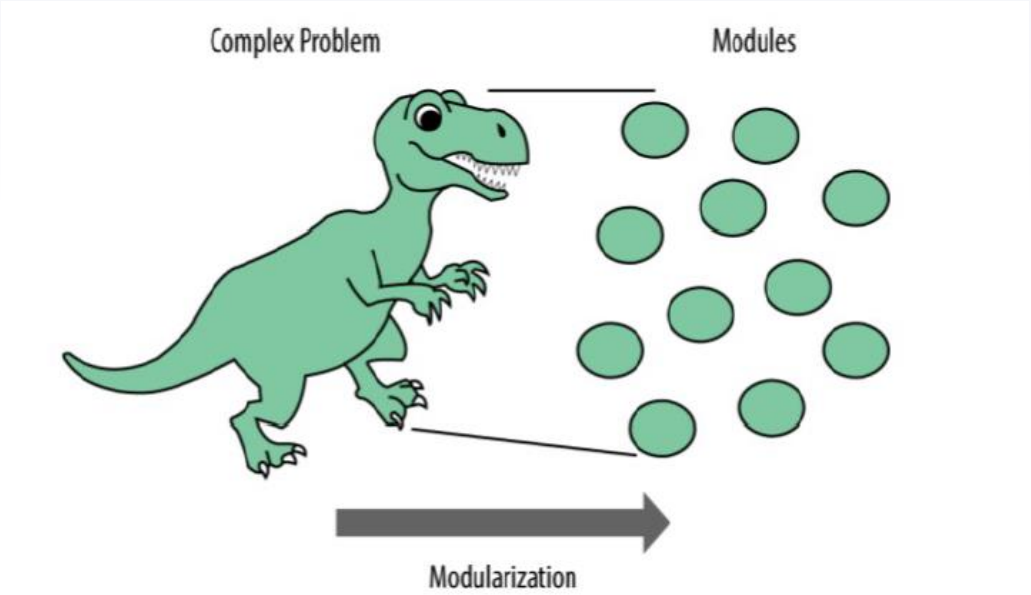
Latar Belakang (2)

*What's wrong with **sequential** & **procedural** programming?*

- As programmers began to write longer and longer programs with more complex tasks, the sequences began to be entangled into what was called spaghetti code.
- Like OOP, procedural programming uses modularity and reuse. However, **procedural programming does not provide for classes** where programming tasks can be bundled into objects. Class objects (instances of classes) can operate on their own data structures, and that cannot be done by functions alone.

Latar Belakang (3)

Why Object Oriented Programming?



Latar Belakang (4)

- Designing object-oriented software is **hard**
- Designing reusable object-oriented software is even harder
- Your design should be specific to the problem at hand but also general enough to address future problems and requirements

Tujuan Mata Kuliah

"Mata kuliah ini dimaksudkan untuk memberi kemampuan pada mahasiswa untuk dapat memahami **pola desain** serta **arsiktektur** dari **struktur kode program** pada pengembangan ***software***."

Design Thinking

- Memecahkan masalah menggunakan pendekatan desain.
- Melihat masalah secara komprehensif (keseluruhan)
- Mengambil keputusan yang sesuai untuk masalah yang dialami

What is Design Patterns?

"descriptions of **communicating objects and classes** that are customized to solve **a general design problem** in a particular context."

Why Design Patterns?

Why Design Patterns?

- Experienced designers reuse solutions that have worked in the past
- Well-structured object-oriented systems have recurring patterns of classes and objects
- Knowledge of the patterns that have worked in the past allows a designer to be more productive and the resulting designs to be more flexible and reusable

Pemrograman Berorientasi Objek

Java VS Python

https://www.coursera.org/articles/python-vs-java

Java:

- **It's simple**
- **It’s platform independent**
- **It’s object oriented**
- **It has a large global community**
- **It supports multithreading**
- **It’s secure**

Python:

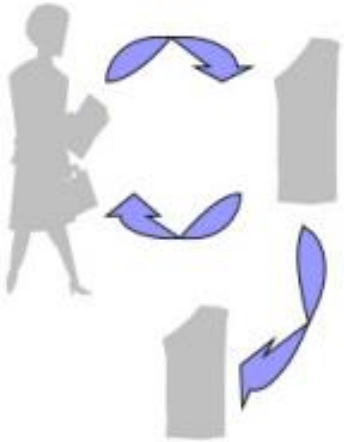
- **It’s platform-independent**
- **It allows for fast development**
- **It offers extensive libraries**
- **It offers a more flexible approach to programming**
- **It offers a more flexible approach to programming**
- **It may boost productivity**

Why OOP

- Object-oriented programming (OOP) is a programming language model organized around objects rather than "actions" and data rather than logic.
- Historically, a program has been viewed as a logical procedure that takes input data, processes it, and produces output data.
- Object-oriented programming takes the view that what we really care about are the objects we want to manipulate rather than the logic required to manipulate them.

Why OOP

- Procedural



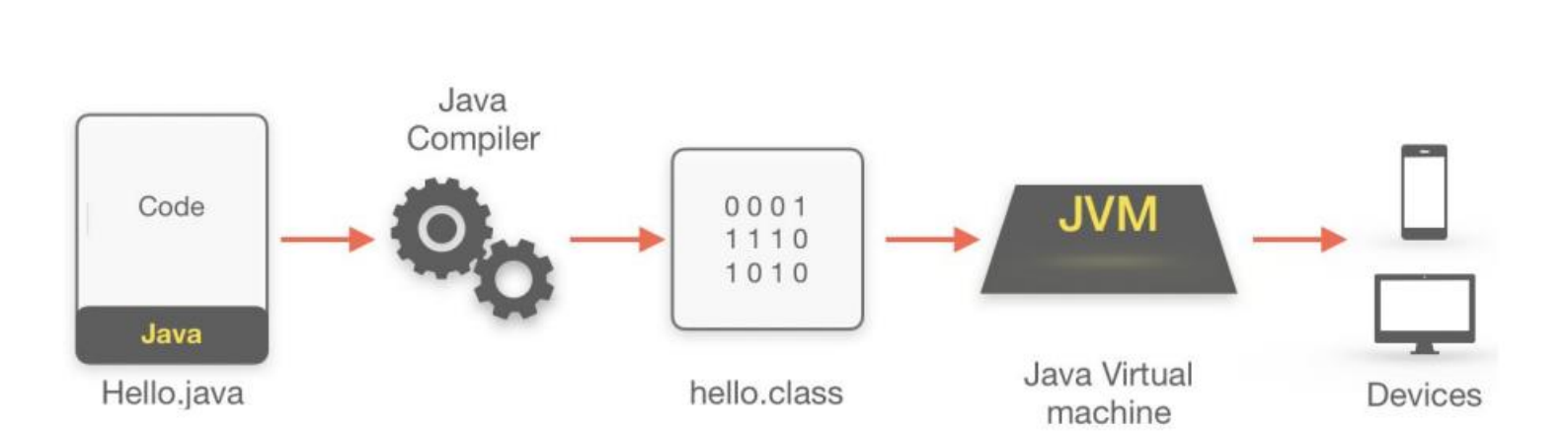
Withdraw, deposit, transfer

- Object Oriented

Customer, money, account

How Java Works

tutorial.eyehunts.com



Differences between Interpreter and Compiler	
Interpreter translates just one statement of the program at a time into machine code.	Compiler scans the entire program and translates the whole of it into machine code at once.
An interpreter takes very less time to analyze the source code. However, the overall time to execute the process is much slower.	A compiler takes a lot of time to analyze the source code. However, the overall time taken to execute the process is much faster.
An interpreter does not generate an intermediary code. Hence, an interpreter is highly efficient in terms of its memory.	A compiler always generates an intermediary object code. It will need further linking. Hence more memory is needed.
Keeps translating the program continuously till the first error is confronted. If any error is spotted, it stops working and hence debugging becomes easy.	A compiler generates the error message only after it scans the complete program and hence debugging is relatively harder while working with a compiler.
Interpreters are used by programming languages like Ruby and Python for example.	Compliers are used by programming languages like C and C++ for example.

https://www.businessinsider.in/difference-between-compiler-and-interpreter

Praktikum 1

Install Java (for linux)

Execute the following command to install the default Java Runtime Environment (JRE), which will install the JRE from OpenJDK 11: `sudo apt install default-jre`

Verify the installation with: `java -version`

You'll see the following output:

```
vindi@pop:~$ java -version
openjdk version "11.0.8" 2020-07-14
OpenJDK Runtime Environment (build 11.0.8+10-
OpenJDK 64-Bit Server VM (build 11.0.8+10-pos
```

Install Java (for linux)

Install Java Development Kit (JDK) selain JRE untuk mengkompilasi dan menjalankan beberapa perangkat lunak berbasis Java tertentu. Untuk menginstal JDK, jalankan perintah berikut, yang juga akan menginstal JRE: `sudo apt install default-jdk`

Pastikan JDK diinstal dengan memeriksa versi javac, kompiler Java: `javac -version`

output :

```
vindi@pop:~$ javac -version
javac 11.0.8
```

Install Java (for windows)

Ikuti langkah tutorial dibawah :
https://www.journaldev.com/476/java-windows-10-download-install

Install Java (for Mac OS)

ikuti langkah tutorial dibawah ini :
https://mkyong.com/java/how-to-install-java-on-mac-osx/

Let's create simple program..

- buat file Hello.java
- buka file tersebut menggunakan text editor dan tulislah code seperti

```
public class Hello {
    public static void main(String[] args) {
        System.out.println("halo vindi, selamat belajar JAVA");
    }
}
```

- simpan, dan ikuti perintah dibawah untuk menjalankan program

```
vindi@pop:~/Documents/campus/asdos oop/practice1$ ls
Hello.java
vindi@pop:~/Documents/campus/asdos oop/practice1$ javac Hello.java
vindi@pop:~/Documents/campus/asdos oop/practice1$ ls
Hello.class  Hello.java
vindi@pop:~/Documents/campus/asdos oop/practice1$ java Hello
halo vindi, selamat belajar JAVA
```

Let's create simple program..

- buat file Hello.java
- buka file tersebut menggunakan text editor dan tulislah code seperti

```
public class Hello {
    public static void main(String[] args) {
        System.out.println("halo vindi, selamat belajar JAVA");
    }
}
```

- simpan, dan ikuti perintah dibawah untuk menjalankan program

```
vindi@pop:~/Documents/campus/asdos oop/practice1$ ls
Hello.java
vindi@pop:~/Documents/campus/asdos oop/practice1$ javac Hello.java
vindi@pop:~/Documents/campus/asdos oop/practice1$ ls
Hello.class  Hello.java
vindi@pop:~/Documents/campus/asdos oop/practice1$ java Hello
halo vindi, selamat belajar JAVA
```

IDE Tools

- Agar lebih mudah dalam execute program java, silahkan gunakan salah satu IDE dibawah ini :
- Eclipse: https://www.eclipse.org/downloads/packages/release/kepler/sr1/eclipse-id%20e-java-developers
- VSCode : https://code.visualstudio.com/
  - Untuk VSCode diperlukan setting dan install plugin tambahan (https://code.visualstudio.com/docs/languages/java)
- IntelliJ : https://www.jetbrains.com/idea/download/#section=linux
- NetBeans : https://netbeans.org/

Praktikum 1

- Buatlah program JAVA yang dapat menghasilkan output data diri Anda!
- Screenshot tampilannya dan upload pada kantung tugas di Elena.

Thank You!

Subhaanakallohumma wa bihamdika, asy-hadu alla ilaha illa anta, as-tagh-firuka wa atuubu ilaik