



# Abstract Class dan Interface

---

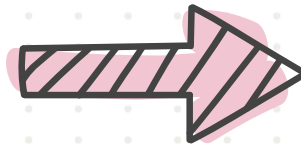




**01**

# **Abstract Class**

---



# Abstract Class

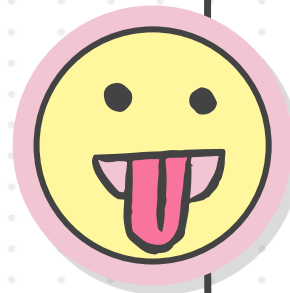


Abstract class adalah salah satu tipe class spesial yang ditandai dengan keyword **abstract** pada deklarasi class tersebut. Sebuah abstract class dibuat apabila class tersebut memang dimaksudkan untuk **tidak akan memiliki instance** (tetapi tetap bisa di-reference-kan). Abstract class dapat memiliki method abstract dan method non abstract.

Method abstract adalah sebuah method yang hanya menyediakan method *flag* (return type, parameter, dan nama method) saja tanpa menyediakan isi (body) dari method tersebut. Method abstract dibuat sengaja supaya method tersebut di-override pada class yang meng-inherit abstract class. Class yang meng-inherit abstract class harus meng-override semua abstract method yang ada

Peraturan abstract class:

- Tidak dapat membuat instance dari abstract class
- Abstract class dapat memiliki constructor
- Abstract class tanpa satupun abstract method diperbolehkan
- Tidak diperbolehkan menggabungkan keyword **final** dan **abstract** (karena kontradiktif)

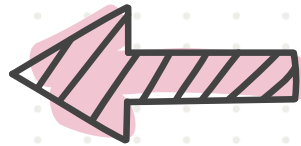




**02**

# Interface

---



# Interface

Interface adalah salah satu tipe class spesial yang ditandai dengan penggunaan keyword **interface** daripada **class** saat pendeklarasian. Penggunaan interface adalah sebuah cara untuk mencapai **total abstraction** di Java, di mana pada interface hanya diperbolehkan untuk membuat abstract method dan final public static attribute. Khusus untuk interface, seluruh method secara default bertipe **abstract**.

Interface hanya memberikan instruksi seperti apa yang harus dan tidak harus dilakukan kepada class yang meng-inheritnya. Class meng-inherit interface tidak dengan menggunakan keyword **extends** tetapi menggunakan keyword **implements**. Hanya dari interface lah sebuah class dapat meraih **multiple inheritance** (sebuah class dapat meng-implement lebih dari satu interface)

Peraturan interface:

- Tidak dapat membuat instance dari interface
- Interface tidak dapat memiliki constructor
- Hanya boleh ada abstract method dan public final static attribute





**03**

# **Perbedaan**





Abstract class	Interface
1) Abstract class can <b>have abstract and non-abstract</b> methods.	Interface can have <b>only abstract</b> methods. Since Java 8, it can have <b>default and static methods</b> also.
2) Abstract class <b>doesn't support multiple inheritance</b> .	Interface <b>supports multiple inheritance</b> .
3) Abstract class <b>can have final, non-final, static and non-static variables</b> .	Interface has <b>only static and final variables</b> .
4) Abstract class <b>can provide the implementation of interface</b> .	Interface <b>can't provide the implementation of abstract class</b> .
5) The <b>abstract keyword</b> is used to declare abstract class.	The <b>interface keyword</b> is used to declare interface.
6) An <b>abstract class</b> can extend another Java class and implement multiple Java interfaces.	An <b>interface</b> can extend another Java interface only.
7) An <b>abstract class</b> can be extended using keyword "extends".	An <b>interface</b> can be implemented using keyword "implements".
8) A Java <b>abstract class</b> can have class members like private, protected, etc.	Members of a Java interface are public by default.
9) <b>Example:</b> <pre>public abstract class Shape{ public abstract void draw(); }</pre>	<b>Example:</b> <pre>public interface Drawable{ void draw(); }</pre>

