

## Java Keyword Part I

**literal** – value ที่พิมพ์ขึ้นมาเฉยๆ ในโปรแกรม เช่น `x = 12`, `s = "hello"` เรียก 12 และ "hello" ว่า literal

**Primitive data type** – ชนิดข้อมูลพื้นฐาน มี 8 ชนิด แบ่งเป็นจำนวนเต็ม 4 จำนวนจริง 2 อักขระ 1 ค่าความจริง 1 data type แบบนี้เก็บเฉพาะ data ไม่มี method

**Type Wrapper** – เป็นคลาสพิเศษที่เกิดจากการแปลง primitive type ให้เป็นคลาส เพื่อที่จะเก็บทั้ง data และ method เช่น `int -> Integer`

กฎการตั้งชื่อ ฝ่าฝืนไม่ได้

- ชื่อตัวแปรจะประกอบด้วย ตัวอักษร, ตัวเลข, underscore(\_), dollar sign(\$)
- ห้ามขึ้นต้นด้วยตัวเลข
- ห้ามมีช่องว่าง และสัญลักษณ์อื่นๆ
- ห้ามใช้คำสงวน เช่น `public`, `new`, `void` (คำที่พิมพ์ไปละขั้นสี่อะ)

มารยาทในการตั้งชื่อ ฝ่าฝืนได้

- การตั้งชื่อตัวแปรควรให้สื่อความหมายและสัมพันธ์กับข้อมูลที่จะเก็บ
- ชื่อ variable และ method ต้องขึ้นต้นด้วยตัวเล็ก และใช้ camel case (ex. `getName`, `setId`)
- ชื่อ class ต้องขึ้นต้นด้วยตัวใหญ่
- ไม่ควรมี \$ ในชื่อ

**obj.** – เป็นค่า value ชนิดหนึ่งที่เก็บทั้ง data และ method

**class** – ทำหน้าที่เป็น type ของ obj แต่ละตัว และเป็นตัวกำหนดว่า obj มี attribute และ method อะไรบ้าง

**public interface** – เมธอดและข้อมูลส่วนที่สามารถเรียกได้จากภายนอกคลาส (พวกที่ประกาศด้วย `public`)

**private implement** – เมธอดและข้อมูลส่วนที่ไม่สามารถเรียกจากภายนอกคลาสได้ (พวกที่ประกาศด้วย `private`)

**System.out.println()** – System เป็นชื่อคลาสที่อยู่ใน package **java.lang**

- out เป็น obj. ของคลาส `PrintStream` ที่อยู่ในคลาส `System`
- `println()` เป็นเมธอดที่ประกาศไว้ในคลาส `PrintStream`

**java.lang** – เป็น package ที่จะ import มาให้อัตโนมัติทุกโปรแกรม

**overloaded method** – คือ method ที่ชื่อเดียวกัน ประกาศไว้ในคลาสเดียวกัน แต่รับพารามิเตอร์ต่างกัน

**override method** – คือ method ที่ประกาศในคลาสลูก แต่ชื่อและการรับพารามิเตอร์เหมือนของคลาสแม่ จะเป็นการเขียนทับเมธอดเดิมของคลาสแม่

**constructor** – เมธอดที่ทำหน้าที่ในการสร้าง obj. โดยจะมีชื่อเหมือนคลาส และไม่ระบุ return type

**default constructor** – ค่าพื้นฐานกรณีที่ไม่สร้าง constructor ระบบจะสร้างมาให้

โดยจะเป็น constructor ที่ไม่รับ parameter

**constant** – ค่าคงที่ จะไม่สามารถแก้ไขค่าใหม่ได้ ประกาศด้วย **final**

- การตั้งชื่อ constant จะตั้งด้วยตัวใหญ่ทุกตัว และใช้ \_ คั่นระหว่างคำ(snake case)

**static** – method หรือ variable โดยปกติจะต้องสร้าง obj. ก่อน จากนั้นจึงเรียกใช้ผ่าน obj. ตัวนั้น

แต่หากต้องการเรียกโดยตรงจากคลาสจะต้องประกาศตัวแปร/เมธอดตัวนั้นเป็น static