

CHAPTER01

ความรู้เบื้องต้นเกี่ยวกับภาษาจาวา

ระบบคอมพิวเตอร์

- หมายถึงกระบวนการหรือขั้นตอนการทำงานของคอมพิวเตอร์
- ประกอบไปด้วยการรับข้อมูล (**input**) การประมวลผล (**process**) และ การแสดงผล (**output**)



บล็อกໄດ້ອະແກມຮບບຄອມພິວເຕອີ່

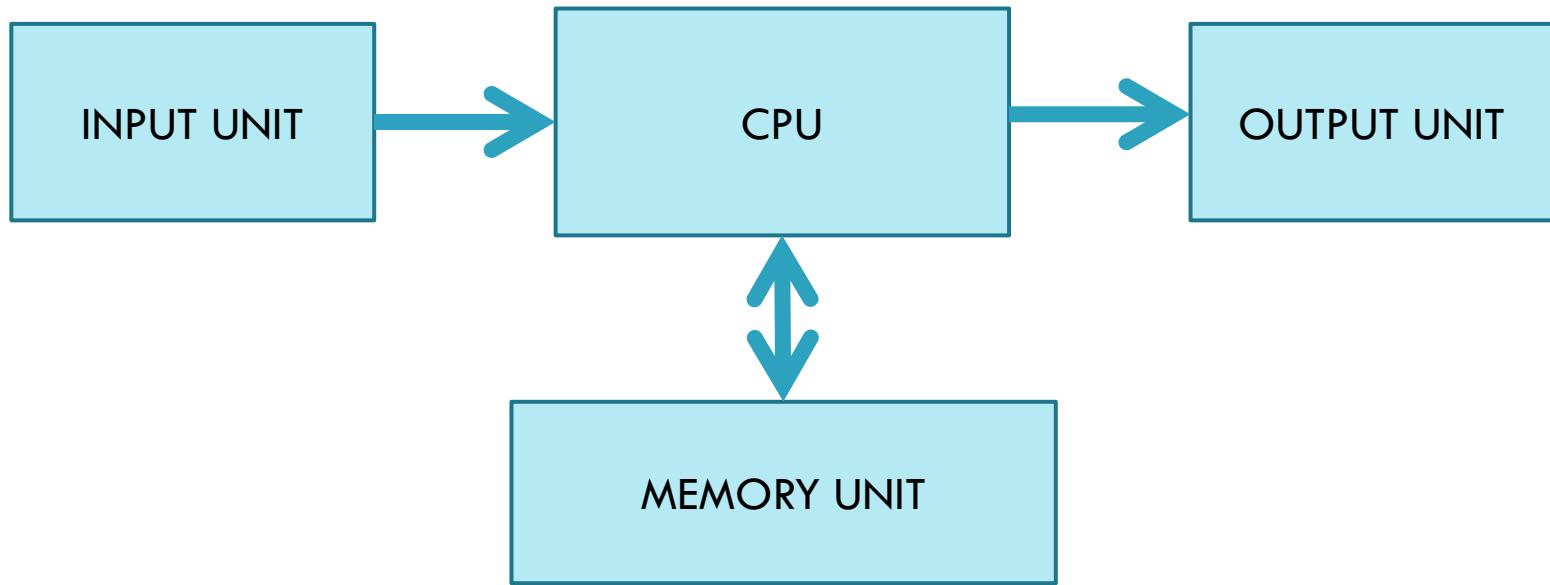
องค์ประกอบสำคัญของระบบคอมพิวเตอร์

- ระบบคอมพิวเตอร์มีองค์ประกอบที่สำคัญ 3 ส่วน คือ
- **Hardware**
- **Software**
- **Peopleware**

สถาปัตยกรรมคอมพิวเตอร์

- องค์ประกอบพื้นฐานของคอมพิวเตอร์ประกอบไปด้วย 4 ส่วนหลักที่สำคัญคือ
 - 1. Input Unit
 - 2. Output Unit
 - 3. CPU
 - 4. Memory Unit

สถาปัตยกรรมคอมพิวเตอร์ (ต่อ)



องค์ประกอบของระบบคอมพิวเตอร์

INPUT UNIT

- เป็นหน่วยที่ทำหน้าที่รับข้อมูลจากภายนอกเข้าไปในคอมพิวเตอร์
- Keyboard**
- Mouse**
- Barcode Reader**
- Light Pen** เป็นต้น

CPU : Central Processing Unit

- คือหน่วยประมวลผลกลาง
- ภายใน CPU ประกอบไปด้วย
- CU : Control Unit หน้าที่ควบคุมการทำงานของ CPU
- และ ALU : Arithmetic Logic Unit หน้าที่ประมวลผลข้อมูลทางด้านคณิตศาสตร์ และตรรกะ
- นอกจากนี้ยังมี Register ซึ่งเป็นหน่วยความจำขนาดเล็กภายใน CPU

OUTPUT UNIT

- คือหน่วยที่ทำหน้าที่นำข้อมูลที่ถูกประมวลผลแล้วไปแสดงผล อุปกรณ์ที่ใช้ในการแสดงผลได้แก่
- Monitor
- Speaker
- Printer
- Projector เป็นต้น

MEMORY UNIT

- แบ่งได้ 2 ประเภท คือ
- **Main Memory** : หน่วยความจำหลัก ได้แก่ RAM , ROM ทำหน้าที่เก็บข้อมูลที่รับมาจากการ **Input Unit** หรือเก็บโปรแกรมที่อ่านมาจากการ **Secondary Memory** เพื่อส่งต่อไปยัง **CPU** ทำการประมวลผลต่อไป
- และ **Secondary Memory** : หน่วยความจำสำรอง ได้แก่ ฮาร์ดดิสก์ , แผ่น CD, แผ่น DVD , Flash Drive เป็นต้น

ภาษาคอมพิวเตอร์

- หมายถึง ภาษาที่ใช้สำหรับเขียนโปรแกรม หรือชุดคำสั่งที่ใช้ควบคุมการทำงานของคอมพิวเตอร์ ให้ปฏิบัติตามความต้องการของผู้ใช้งาน
- ภาษาคอมพิวเตอร์แบ่งออกได้เป็น 2 ระดับ คือ
 - 1. ภาษาระดับต่ำ (**Low Level Language**)
 - 2. ภาษาระดับสูง (**High Level Language**)

Low Level Language

- เป็นภาษาคอมพิวเตอร์ที่ใช้กันในยุคแรกๆ
- มีความยุ่งยากในการเขียนมาก แบ่งได้เป็น 2 ประเภท คือ
- 1. Machine Language (ภาษาเครื่อง)
- 2. Assembly Language (ภาษาแอสเซมบลี)

Machine Language

- เป็นภาษาหรือคำสั่งที่ใช้ในการสั่งงาน หรือติดต่อกับเครื่องคอมพิวเตอร์โดยตรง
- ภาษาเครื่องประกอบไปด้วยรหัสเลขฐานสองซึ่งเทียบได้กับลักษณะของสัญญาณทางไฟฟ้า เช่นกับหลักการทำงานของเครื่องคอมพิวเตอร์พอดีทำให้คอมพิวเตอร์ทำงานตามคำสั่งได้ทันที

Assembly Language

- ภาษาแอสเซมบลี่ จัดเป็นภาษาสัญลักษณ์ (Symbolic Language)
- เป็นภาษาที่พัฒนาต่อจากภาษาเครื่อง
- ใช้สัญลักษณ์ข้อความแทนกลุ่มของเลขฐานสอง
- ทำให้การเขียนโปรแกรมสะดวกมากขึ้นกว่าภาษาเครื่อง
- แต่ก็ยังคงต้องจำความหมายของสัญลักษณ์ที่ใช้แทนภาษาเครื่อง

High Level Language

- เป็นภาษาคอมพิวเตอร์ที่ได้รับการพัฒนาให้สามารถใช้งานได้ง่ายและสะดวกมากขึ้น
- การเขียนภาษาไม่ขึ้นกับฮาร์ดแวร์หรือลักษณะการทำงานภายใต้เครื่องคอมพิวเตอร์
- โปรแกรมเมอร์ไม่จำเป็นต้องเข้าใจระบบการทำงานของเครื่องคอมพิวเตอร์มากนัก เพียงแต่เข้าใจกฎเกณฑ์ในการเขียนแต่ละภาษาให้ดีตัวภาษาคล้ายกับภาษาอังกฤษ

High Level Language (Continue)

- ภาษาระดับสูงเครื่องคอมพิวเตอร์จะยังไม่เข้าใจ จึงต้องมีการแปลให้เป็นภาษาเครื่องเสียก่อน
- โปรแกรมที่ใช้แปลภาษาระดับสูง แบ่งออกได้เป็น 2 ประเภท คือ
- 1. interpreter
- 2. complier

Interpreter

- การทำงานของอินเตอร์พรีตเตอร์จะแปลภาษาที่ละเอียดสั้ง แปลเสร็จก็จะปฏิบัติตามคำสั่งที่แปลโดยทันที หลังจากนั้นจึงไปแปลคำสั่งถัดไป
- ถ้าหากในขณะที่แปลคำสั่งเกิดข้อผิดพลาด ก็จะหยุดให้โปรแกรมเมอร์แก้ไข ข้อผิดพลาดคำสั่งนั้นก่อน
- ภาษาที่ใช้อินเตอร์พรีตเตอร์ ได้แก่ **BASICA , GWBASIC** เป็นต้น

Complier

- คอมไพล์เรอร์จะทำการแปลทั้งโปรแกรม และเก็บโปรแกรมที่แปลได้ในรูปแบบของออบเจ็คโปรแกรม (**object program**)
- ถ้าโปรแกรมที่แปลได้ไม่มีข้อผิดพลาด ก็จะปฏิบัติตามคำสั่งนั้นๆ ทันที
- ถ้าโปรแกรมที่แปลมีข้อผิดพลาด คอมไпал์เรอร์ก็จะบอกข้อผิดพลาดทั้งหมดให้ทราบ
- ภาษาที่ใช้คอมไпал์เรอร์ได้แก่ **C, Pascal, FORTRAN** เป็นต้น



วิวัฒนาการภาษาจาวา

- ปี คศ. 1991 ชันไมโครซิสเต็ม (Sun Microsystems) ได้ทำการวิจัย เพื่อพัฒนาซอฟต์แวร์ที่ใช้ควบคุมอุปกรณ์อิเล็กทรอนิกส์ขนาดเล็ก ซึ่งได้ให้ กำเนิดภาษาโอ๊ค (Oak)
- ปี คศ. 1993 ภาษาโอ๊คได้ถูกปรับปรุงใหม่เพื่อใช้ในการสร้างเว็บแอพลิ เคชั่น (Web Application) พร้อมกับสร้างเว็บбраузอร์ (Web Browser) ที่รองรับ ชื่อว่าเว็บรันเนอร์ (Web Runner)

วิวัฒนาการภาษาจาวา (ต่อ)

- คศ. 1995 บริษัทชั้นไมโครซิสเต็มได้เปิดตัวภาษาจาวา (**Java**) โดยเปลี่ยนชื่อมาจากภาษาโอลีค (เนื่องจากไปซ้ำกับภาษาที่มีอยู่แล้ว) พร้อมกับเว็บбраузอร์ ที่รองรับภาษาจาวา ชื่อว่า **HotJava (Web Runner เดิม)**
- บริษัทชั้นไมโครซิสเต็มได้มีการแจกจ่าย **Java Development Kit (JDK)** ซึ่งเป็นชุดพัฒนาโปรแกรมภาษาจาวาในอินเทอร์เน็ต



ข้อดีของภาษาจาวา

- ทำงานบนเว็บбраузอร์ได้
- มีความปลอดภัยสูง
- สนับสนุนการทำงานหลากหลายระดับ
- สามารถทำงานบนเครื่องคอมพิวเตอร์ต่างระบบได้
- เป็นภาษาเชิงวัตถุ
- มีความเรียบง่าย

ข้อดีของภาษาจาวา (ต่อ)

- มีระบบจัดการคืนพื้นที่ในหน่วยความจำอัตโนมัติ (**automatic garbage collection**)
- มีคลาสจำนวนมากมายให้ใช้งาน

พรี

จาแวแพล็ตฟอร์ม

- โปรแกรมที่เขียนด้วยภาษาจาแวสามารถนำไปทำงานบนระบบปฏิบัติการต่างๆ กัน เช่น **Windows** หรือ **Linux** หรือ **MacOS**
- สามารถทำงานบนคอมพิวเตอร์ที่ใช้ซีพียูคนละแบบ เช่น ซีพียูของ **intel** หรือซีพียูของ **IBM**
- **Write Once Run Anywhere** นั่นเอง
- นั่นเพราะว่าจาแวมีระบบที่เรียกว่า “**จาแวแพล็ตฟอร์ม**”

จาว่าแพล็ตฟอร์ม (ต่อ)

- เราเรียกระบบที่โปรแกรมอาศัยทำงานว่า “แพล็ตฟอร์ม (platform)”
- แพล็ตฟอร์มตีความหมายรวมไปถึงฮาร์ดแวร์และซอฟต์แวร์ด้วย หรือ
- แพล็ตฟอร์มอาจจะหมายถึงเฉพาะส่วนที่เป็นซอฟต์แวร์เพียงอย่างเดียว เช่นระบบปฏิบัติการ และชุดของโปรแกรมย่อยที่ใช้ในการพัฒนาซอฟต์แวร์หรือไลบรารี (**library**)
- จาว่าแพล็ตฟอร์มนั้นเป็นซอฟต์แวร์

JAVA แพลตฟอร์ม (ต่อ)

- Java แพลตฟอร์มประกอบไปด้วย ตัวแปลภาษา โปรแกรมเครื่องจักรเสมือน (JVM) และคลาสที่เป็นแกนหลักของภาษา Java
- จาłamี 3 แพลตฟอร์ม คือ
 - 1. **Java 2 Platform, Standard Edition (J2SE)** ใช้สำหรับเขียนโปรแกรมทั่วไปหรือที่เรียกว่า “Java อ�플ิเคชัน (java application)” และโปรแกรมที่ทำงานบนเว็บбраузอร์ หรือที่เรียกว่า “แอปเพล็ต (java applet)”

JAVAแพล็ตฟอร์ม (ต่อ)

- 2. Java 2 Platform, Enterprise Edition (J2EE) ใช้สำหรับเขียนโปรแกรมแบบมัลติเทียร์ (Multitiered) พัฒนามาจาก J2SE และเพิ่มคุณสมบัติสำหรับการพัฒนาโปรแกรมในระดับองค์กร
- 3. Java 2 Platform, Micro Edition (J2ME) เป็นแพล็ตฟอร์มที่ทำงานบนสินค้าอิเล็กทรอนิกส์ เช่น โทรศัพท์มือถือ รุ่นเก่า พีดีเอ และกล่องเคเบิลทีวี (TV set-top box)

เครื่องมือสำหรับเขียนโปรแกรมภาษาจาวา

- 1. ชุดพัฒนาภาษาจาวา (Java Development Kit : JDK)
 - โปรแกรมแปลงภาษาจาวา (Java Complier)
 - โปรแกรมรันภาษาจาวา (Java Runtime Environment : JRE)
- 2. โปรแกรมที่ใช้เขียนต้นฉบับโปรแกรม (Text Editor)

ชุดพัฒนาภาษาจาวา

- ชุดพัฒนาโปรแกรมภาษาจาวา (**Java Development Kit : JDK**) มีส่วนประกอบที่สำคัญคือ
- คอมไพล์เลอร์ และสภาพแวดล้อมสำหรับรันโปรแกรมจาวา
- ชื่ง **jdk** สามารถดาวน์โหลดได้จากเว็บไซต์

<http://www.oracle.com>

Java complier

- ทำหน้าที่แปลภาษาจาวาให้เป็นไบต์โค๊ด (byte code) ซึ่งเป็นคำสั่งที่ไม่ อิงกับชีพิญของเครื่องคอมพิวเตอร์ยี่ห้อใดยี่ห้อหนึ่ง
- เครื่องคอมพิวเตอร์ที่ประมวลผลไบต์โค๊ดเป็น “เครื่องคอมพิวเตอร์เสมือน” หรือที่เราเรียกว่า เครื่องจักรเสมือนจาวา Java Virtual Machine



Java Runtime Environment : JRE

- เป็นโปรแกรมที่ใช้รับไฟล์โค้ด สภาพแวดล้อมนี้ประกอบไปด้วยเครื่องจักรเสมือน Java (**Java Virtual Machine – JVM**) และคลาสที่เป็นแกนหลักของภาษา Java คอมพิวเตอร์ใด ๆ ที่มี JRE ติดตั้งอยู่จะสามารถรันโปรแกรมที่เขียนขึ้นด้วยภาษา Java ได้



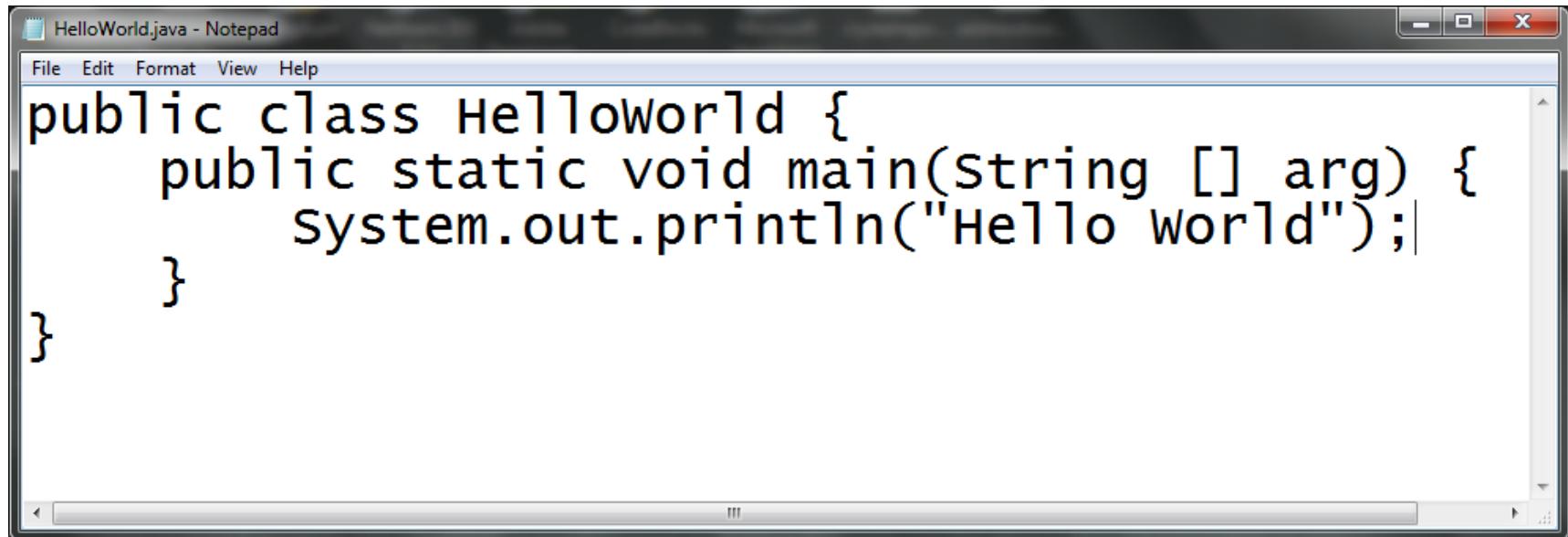
โปรแกรมที่ใช้เขียนต้นฉบับโปรแกรม (Text Editor)

- ชุดพัฒนาโปรแกรมภาษาจาวา มีโปรแกรมสำหรับคอมไฟล์และรันไฟล์โค้ดแต่ไม่มี **Text Editor** ที่ใช้สำหรับเขียนไฟล์นามสกุล **java**
- แต่เราสามารถใช้ **Text Editor** ตัวใดก็ได้เพื่อใช้เขียนโปรแกรม
- **Text Editor** ได้แก่ **Notepad, J-Lab, Netbeans, Eclipse, Visual Studio Code, IntelliJ**

โปรแกรม HelloWorld

- โปรแกรมที่ผู้เขียนส่วนใหญ่เขียนเป็นโปรแกรมแรก ส่วนใหญ่ต้องเขียนโปรแกรม HelloWorld
- เริ่มต้นทดลองเขียนโดยเปิด Notepad ขึ้นมา
- สร้างไฟล์ชื่อว่า **HelloWorld.java**
- บันทึกไฟล์ไว้ที่ D:\javaCode (บันทึกในไดร์ฟอะไรก็ได้)

ໂປຣແກຣມ HelloWorld (ຕ່ອ)



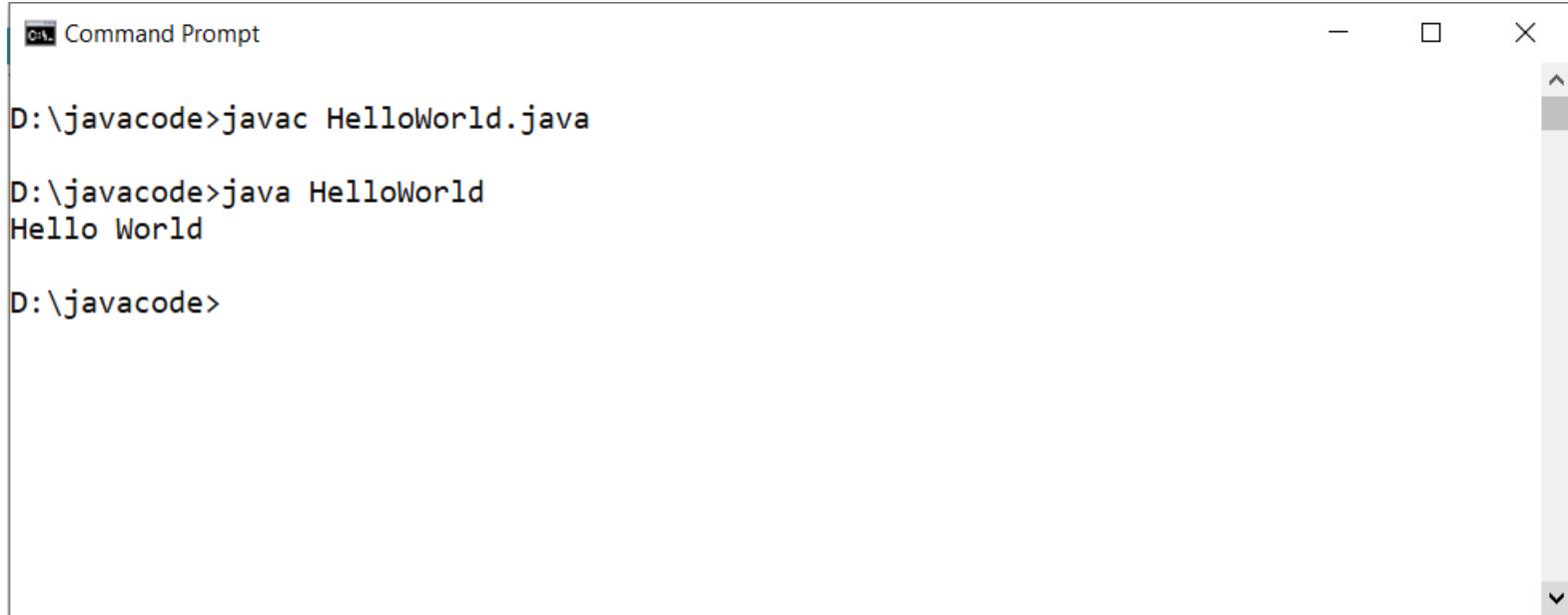
A screenshot of a Windows-style Notepad window titled "HelloWorld.java - Notepad". The window contains the following Java code:

```
public class HelloWorld {  
    public static void main(String [] arg) {  
        System.out.println("Hello world");  
    }  
}
```

โปรแกรม HelloWorld (ต่อ)

- เมื่อเขียนโปรแกรมเสร็จแล้ว เราต้องคอมไพล์โปรแกรมก่อน จึงจะสามารถรันโปรแกรมได้
- การคอมไพล์และรันโปรแกรม **HelloWorld** ทำได้โดยการเปิดโปรแกรม **command prompt** จากนั้นก็พิมพ์คำสั่ง
- **javac HelloWorld.java** (คอมไпал์โปรแกรม จะต้องไปยังโฟล์เดอร์ที่มีไฟล์นี้อยู่)
- **java HelloWorld** (รันโปรแกรม จะต้องไปยังโฟล์เดอร์ที่มีไฟล์นี้อยู่)

ໂປຣແກຣມ HelloWorld (ຕ່ອ)



A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the following text output:

```
D:\javacode>javac HelloWorld.java
D:\javacode>java HelloWorld
Hello World
D:\javacode>
```

การตั้งค่า PATH

- โดยปกติแล้วโปรแกรม **command prompt** จะไม่รู้จักคำสั่ง **javac**
- วิธีการจะทำให้โปรแกรม **command prompt** รู้จักคำสั่งที่ใช้สำหรับพัฒนาโปรแกรมภาษาจาวาทำได้โดยการตั้งค่า **PATH** ให้แก่เครื่องคอมพิวเตอร์ มีขั้นตอนดังนี้
 - 1. copy path ของ **jdk** ที่ติดตั้งในเครื่องคอมพิวเตอร์ เช่น

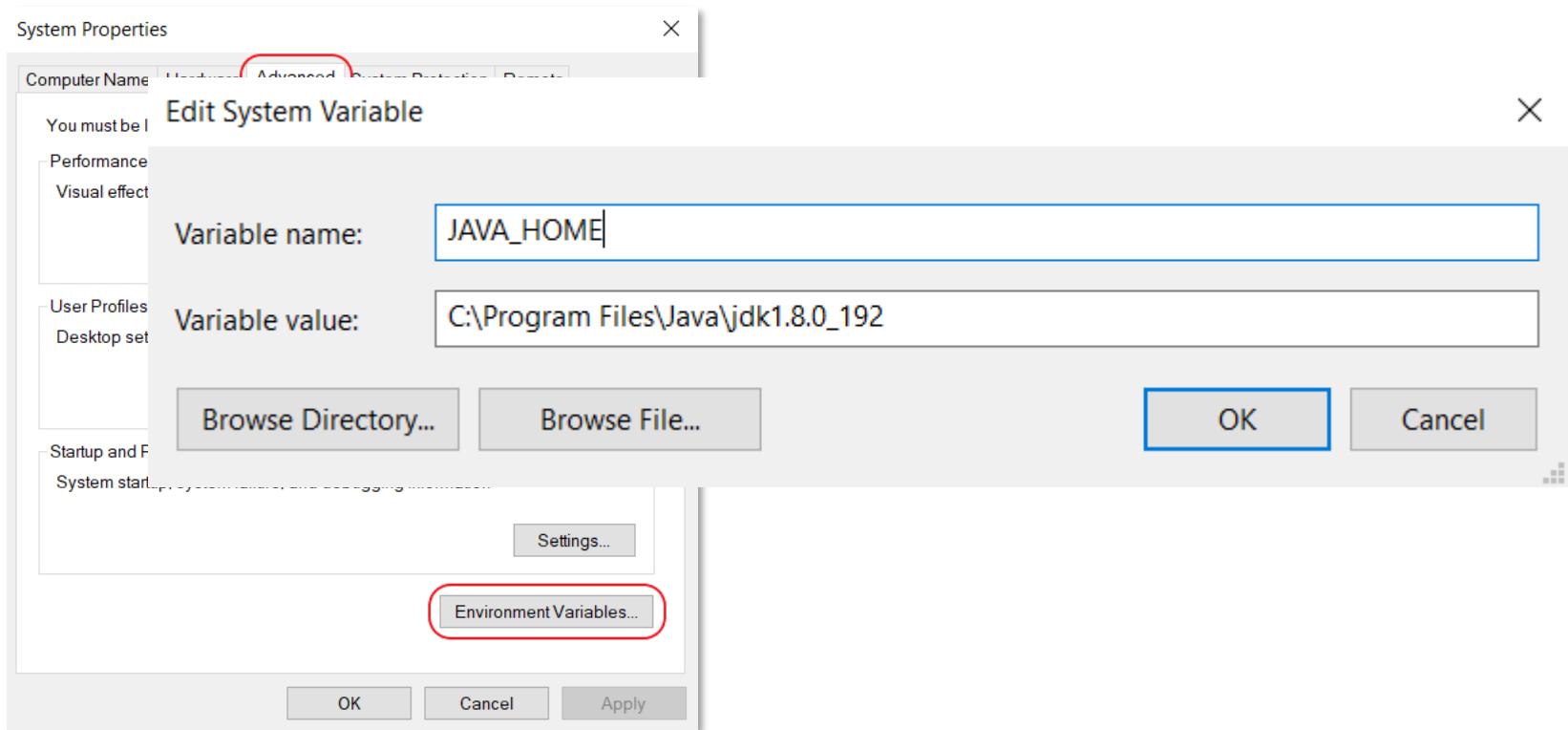


> This PC > Windows 10 (C:) > Program Files > Java > jdk1.8.0_192

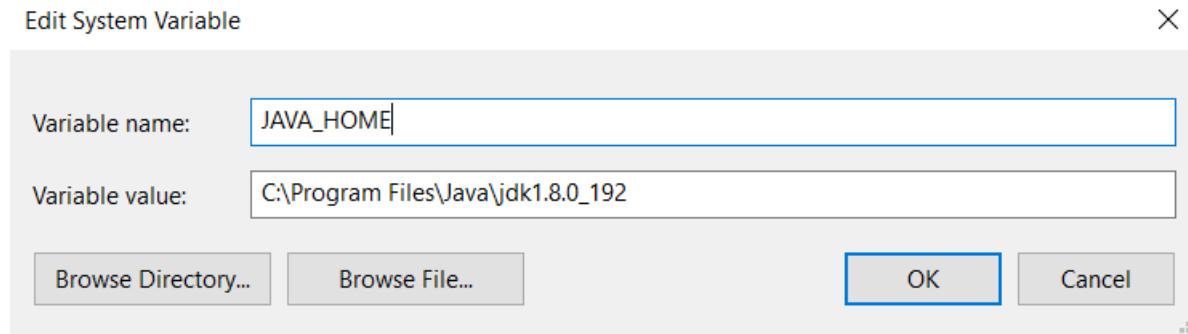
การตั้งค่า PATH (ต่อ)

- 2. กำหนดค่า path ที่ได้จากข้อ 1 ให้แก่ตัวแปร path ของระบบ
 - ▣ คลิกขวาที่ My Computer > properties > Advanced system settings เพื่อเปิด system properties

การตั้งค่า PATH (ต่อ)

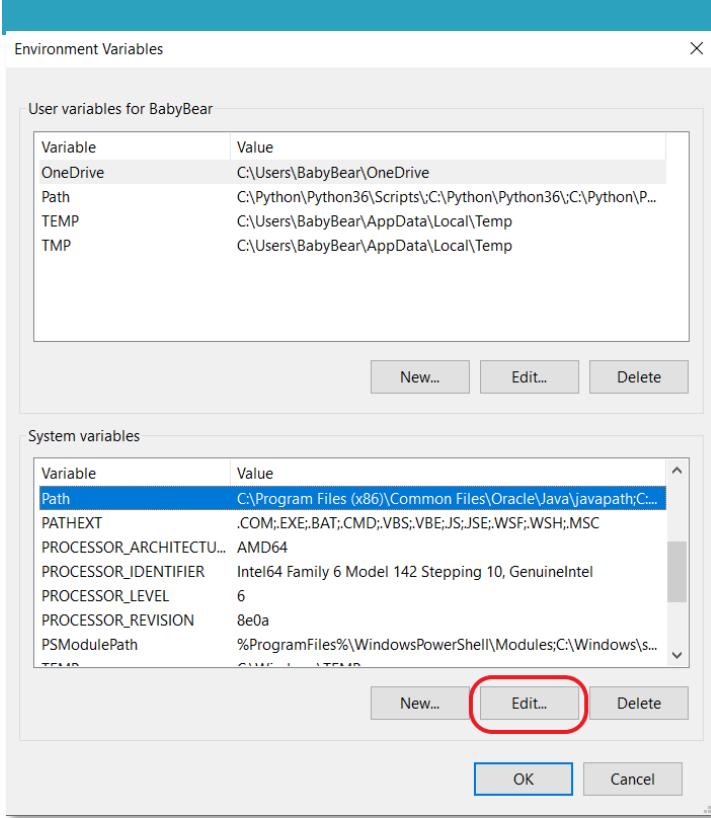


การตั้งค่า PATH (ต่อ)



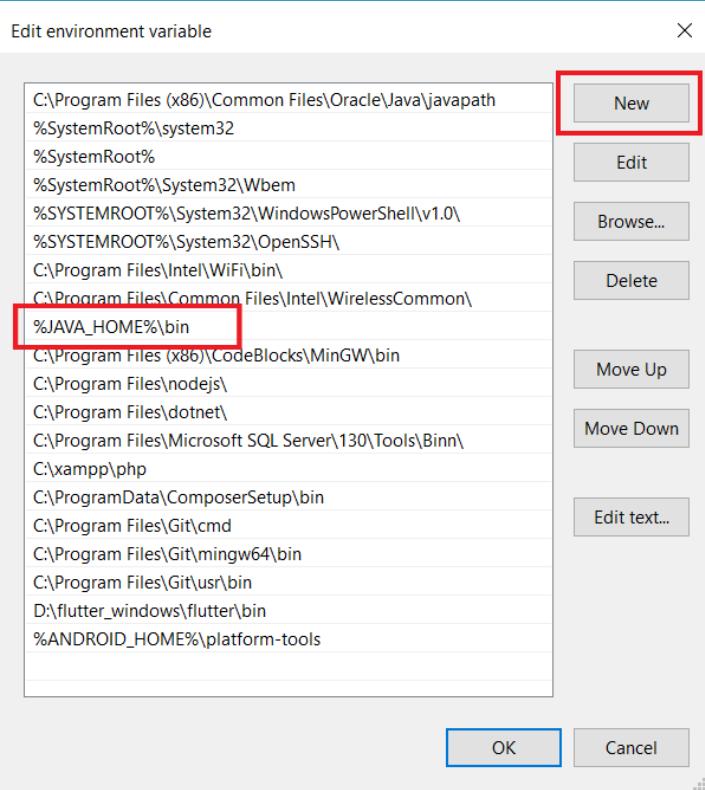
New Variable name

การตั้งค่า PATH (ต่อ)



เลือกตัวแปร Path และคลิกที่ปุ่ม Edit...

การตั้งค่า PATH (ต่อ)



Paste พาธที่ก็อปปี้ไว้ในข้อที่ 1 วางใน
ตำแหน่ง **Variable value** ก่อน paste
ให้ใส่เครื่องหมายเซมิโคลอน (;) คันระหว่าง
ค่าของตัวแปรตัวอื่นก่อน หลังจากนั้นทำการ
คลิกที่ปุ่ม **OK**

เอกสารอ้างอิง

- วรเชษฐ์ สุวรรณิก. เขียนโปรแกรม Java เป็นต้น 2nd edition.
กรุงเทพฯ : ซีเอ็ดยูเคชั่น, 2556.

End.

