

ภาษาคอมพิวเตอร์ computer language

ผู้ช่วยศาสตราจารย์สมเกียรติ ช่อเหมือน (tko@webmail.npru.ac.th)

สาขาวิชาวิศวกรรมซอฟต์แวร์ คณะวิทยาศาสตร์และเทคโนโลยี



บทนำ

- ประเภทของภาษาคอมพิวเตอร์
 - ภาษาชั้นสูง (High-level language)
 - ภาษาแอสเซมบลี (Assembly language)
- การเลือกใช้ภาษาโปรแกรม
 - คุณสมบัติของภาษาโปรแกรม
 - การเลือกใช้ภาษาโปรแกรม
- Frameworks & Libraries



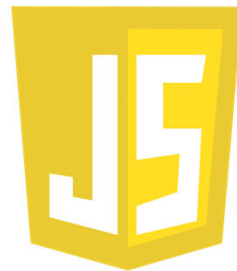
ภาษาชั้นสูง (High-level Language)

- ใกล้เคียงหรือคล้ายกับภาษามนุษย์ ทำให้เข้าใจได้ง่าย
- ใช้คอมไพเลอร์หรืออินเทอร์พรีเตอร์แปลงเป็นภาษาเครื่อง (machine language)



Java™

JavaScript



ภาษาอัสเซมบลี (Assembly Language)

- ภาษาที่มีความใกล้เคียงกับภาษาเครื่อง
- ภาษาอัสเซมบลีถูกแปลงเป็นภาษาเครื่องโดยใช้โปรแกรมที่เรียกว่า **assembler**

- ตัวอย่างโค้ดภาษาอัสเซมบลี:

```
mov eax, 5 ;
```

กำหนดค่า 5 ให้กับ register eax

```
add eax, 3 ;
```

บวก 3 กับค่าใน register eax

คุณสมบัติของภาษาโปรแกรม

ความปลอดภัย (Security)

- คุณสมบัติในการป้องกันความเสี่ยงทางความปลอดภัย
- ป้องกันการโจมตีจากภายนอกหรือจัดการความจำอย่างปลอดภัย

ความเป็นกลาง (Portability)

- การทำงานบนแพลตฟอร์มหรือระบบปฏิบัติการต่าง ๆ โดยไม่ต้องแก้ไขโค้ดมาก

ประสิทธิภาพ (Efficiency)

- ความสามารถในการประมวลผลและใช้ทรัพยากรของระบบ

ความยืดหยุ่น (Flexibility)

- การปรับแต่ง, ขยายหรือปรับปรุงโปรแกรมได้ง่าย



การเลือกภาษาคอมพิวเตอร์

- โปรเจกต์ที่ต้องการพัฒนา
- ประสบการณ์และความถนัดของทีมพัฒนา
- ความต้องการด้านประสิทธิภาพและทรัพยากร
- การรองรับของชุมชนและความพร้อมของทรัพยากร
- การรองรับของบริษัทและโครงการ
- การขยายขนาดและบำรุงรักษา
- ระยะเวลาการพัฒนา



การเลือกใช้ภาษา

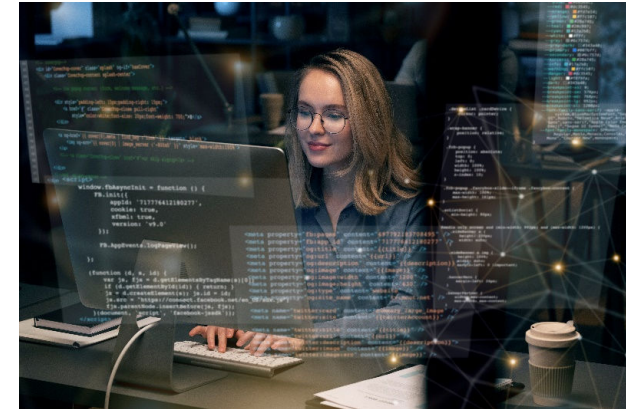
วิเคราะห์ความต้องการของโปรเจค

- ความต้องการทางฟังก์ชัน
- ประสิทธิภาพ
- ความปลอดภัยของโปรเจค

สิ่งแวดล้อมการพัฒนา (Development Environment)

- พิจารณชอพต์แวร์ ฮาร์ดแวร์ และแพลตฟอร์มที่จะใช้ในการพัฒนา

ความเชี่ยวชาญของทีมพัฒนา (Development Team Expertise)



ภาษาคอมพิวเตอร์ที่นิยม (1)

- **JavaScript**

- การพัฒนาเว็บเนื่องจากสามารถใช้ทั้งในฝั่งไคลเอ็นต์และเซิร์ฟเวอร์ (Node.js)
- ชุมชนใหญ่และมีเฟรมเวิร์คมากมายเช่น React, Angular, Vue.js

- **Python**

- การพัฒนาเว็บ (Django, Flask)
- การวิเคราะห์ข้อมูล การเรียนรู้ของเครื่อง (TensorFlow, PyTorch) และการทำงานแบบอัตโนมัติ
- ไวยากรณ์ที่อ่านง่ายและเหมาะสำหรับการพัฒนาแบบรวดเร็ว

- **Java**

- การพัฒนาแอปพลิเคชันองค์กร แอนดรอยด์แอป และระบบที่มีความเสถียร
 - ระบบการจัดการหน่วยความจำและมีชุมชนขนาดใหญ่
-

ภาษาคอมพิวเตอร์ที่นิยม (2)

- **C#**

- การพัฒนาแอปพลิเคชันบน Windows
- เกมด้วย Unity
- เว็บด้วย .NET Framework
- โครงสร้างพื้นฐานจาก Microsoft

- **Ruby**

- ภาษาที่ใช้พัฒนาแบบรวดเร็ว มีเฟรมเวิร์ค Ruby on Rails ที่ช่วยในการพัฒนา

- **C++ / C**

- การพัฒนาซอฟต์แวร์ที่ต้องการประสิทธิภาพสูง
 - วิดีโอเกม
 - ระบบฝังตัว
 - การประมวลผลระดับล่าง
-

Framework

- ชุดโค้ดและเครื่องมือที่กำหนดโครงสร้างและการทำงานไว้ล่วงหน้า
- ง่ายต่อการสร้างแอปพลิเคชัน
- โครงสร้างที่มีแนวคิดในการควบคุม (**Inversion of Control - IoC**)
- การควบคุมโปรแกรมตามที่กำหนดไว้
- นักพัฒนาต้องเขียนโค้ดเพิ่มเติมตามโครงสร้างที่กำหนด
- ตัวอย่าง framework
 - Angular, React (มักถูกใช้เป็น framework) ของ Javascript
 - Django สำหรับ Python



Library

- ชุดของฟังก์ชันหรือโค้ดที่สามารถเรียกใช้เพื่อทำงานที่เฉพาะเจาะจงได้
- ไม่เหมือนกับ framework
- libraries ไม่กำหนดโครงสร้างหลักของโปรแกรม
- เครื่องมือที่นักพัฒนาสามารถเลือกใช้ตามต้องการ เพื่อเพิ่มฟีเจอร์หรือฟังก์ชัน
- ตัวอย่าง Javascript libraries
 - jQuery
 - TensorFlow.js



สรุปท้ายบท

- ภาษาคอมพิวเตอร์เป็นเครื่องมือหลักที่ใช้ในการสื่อสารกับเครื่องจักร
- ภาษาระดับสูงที่ใกล้เคียงกับภาษามนุษย์ ไปจนถึงภาษาระดับต่ำที่ใกล้เคียงกับรหัส ใช้การแปลหรือตีความภาษา
- ภาษาระดับสูงเช่น Python, Java และ C# ได้รับการออกแบบมาเพื่อปรับปรุงความเร็วในการพัฒนา

