

หลักสูตร Basic Java Programming

ภาพรวมของการเรียน

Java เป็นภาษาโปรแกรมเชิงวัตถุ (Object Oriented Programming) พัฒนาโดยซัน ไมโครซิสเต็มส์ แม้ว่าจะมีชื่อคล้ายกัน แต่ ภาษา Java ไม่มีความเกี่ยวข้องใดๆกับภาษา JavaScript จุดมุ่งหมายหลักในการพัฒนา Java คือใช้ภาษาโปรแกรมเชิงวัตถุ ไม่ขึ้นกับแพลตฟอร์ม (สถาปัตยกรรม และ ระบบปฏิบัติการ) เหมาะกับการใช้ในระบบเครือข่าย มีไลบรารีสนับสนุนมากมาย และเรียกใช้งานจากระยะไกลได้อย่างปลอดภัย สามารถสร้างแอปพลิเคชันบนโทรศัพท์มือถือ และ Graphical User Interface ด้วยภาษา Java หลักสูตรนี้สอนการเขียนโปรแกรมภาษา Java พื้นฐานเพื่อให้ผู้เรียนสามารถต่อยอดไปเรียนการเขียนโปรแกรม Java ขั้นสูงและทำแอปพลิเคชันต่อไป หลักสูตรนี้เหมาะสำหรับนักศึกษาที่กำลังเรียนในมหาวิทยาลัย นักเรียนที่เตรียมตัวสอบเข้ามหาวิทยาลัยในสายคอมพิวเตอร์ วิศวกรรมและโปรแกรมเมอร์

ระยะเวลาที่ใช้ในการเรียน

หลักสูตร 24 ชั่วโมง เรียนทั้งสิ้น 8 ครั้ง (ครั้งละ 3 ชั่วโมง)

โปรแกรมที่ใช้ในการเรียน

IntelliJ IDEA

หลักสูตรนี้เหมาะสำหรับ

- นักเรียนที่ต้องการปูพื้นฐาน Java Programming ก่อนเริ่มเรียนในระดับมหาวิทยาลัยหรืออาชีวศึกษา
- นักศึกษาที่กำลังศึกษาในระดับมหาวิทยาลัย
- วิศวกร
- โปรแกรมเมอร์

วิทยากร

อาจารย์ กฤษณ์ อุดมจารุมณี

อาจารย์ประจำหลักสูตร Computer Programming

การศึกษา .:

- Microsoft Certified Professional - .NET Framework 3.5 Windows Forms Application.
- ปริญญาตรี วิศวกรรมไฟฟ้า มหาวิทยาลัยสงขลานครินทร์
- ปริญญาโท M.B.A, Drexel University, Pennsylvania, USA

Java Programming Basic Course Outline

ครั้งที่ 1	Syntax ของภาษา Java 1. ทำความรู้จักกับภาษา Java 2. วิธีติดตั้ง Java Development Kit (JDK) 3. วิธีตั้งค่าใน IntelliJ IDEA สำหรับเขียนโค้ดภาษา Java 4. เริ่มต้นเขียนโปรแกรมภาษา Java 5. วิธีเขียนคำสั่งเพื่อแสดงผลเอาต์พุต (System.out) 6. วิธีเขียน Comment 7. วิธีกำหนดตัวแปร 8. วิธีเขียนคำสั่งป้อนข้อมูลของผู้ใช้ (Scanner class) 9. รู้จักกับประเภทของข้อมูล 10. ตัวดำเนินการของ Java 11. ตัวดำเนินการคณิตศาสตร์, ตัวดำเนินการเปรียบเทียบ, ตัวดำเนินการด้านตรรกะ 12. Exercises (แบบฝึกหัด)
ครั้งที่ 2	คำสั่งของ Java 13. คำสั่งในการจัดการกับข้อความ (String functions) 14. คำสั่งด้านคณิตศาสตร์ (Math functions) 15. คำสั่งด้านวันที่และเวลา (Date and Time functions) 16. คำสั่งด้านตรรกะ (Boolean) 17. Exercises (แบบฝึกหัด)
ครั้งที่ 3	คำสั่งของ Java 18. คำสั่งในการวนรอบ (for loop) 19. คำสั่งในการวนรอบ (while loop) 20. คำสั่งในการตัดสินใจโดยมีเงื่อนไข (conditional statement) 21. คำสั่ง switch 22. คำสั่ง break และ continue 23. Exercises (แบบฝึกหัด)
ครั้งที่ 4	Array กับ ArrayList วิธีสร้างข้อมูลชนิดอาร์เรย์ (Array) 24. คำสั่งของอาร์เรย์ 25. วิธีสร้างข้อมูลชนิดอาร์เรย์ลิสต์ (ArrayList) 26. คำสั่งของอาร์เรย์ลิสต์ 27. Exercises (แบบฝึกหัด)

ครั้งที่ 5	Function ที่กำหนดเอง 28. วิธีสร้างเมธอด (Method) 29. วิธีกำหนดพารามิเตอร์ (Parameters) ในเมธอด 30. วิธีเขียน Method Overloading 31. ขอบเขตของตัวแปร (Scope of Variables) 32. วิธีเขียนคำสั่งในการเรียกซ้ำ (Recursion) 33. Exercises (แบบฝึกหัด)
ครั้งที่ 6	ไฟล์อินพุต/เอาต์พุต กับการจัดการกับ Error 34. วิธีเขียนคำสั่งจัดการกับไฟล์ (Files) 35. วิธีจัดการกับข้อผิดพลาด (Exception Handling) 36. การแก้จุดบกพร่องต่าง ๆ ในโปรแกรม (Debugging) 37. Exercises (แบบฝึกหัด)
ครั้งที่ 7	วิธีเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming) 38. ทำความรู้จักกับวิธีเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming) 39. วิธีสร้างคลาส Classes 40. วิธีเขียนคลาสหรือพเพอร์ติ (Class Attributes) 41. วิธีสร้างอ็อบเจ็กต์จากคลาส (Objects/Class) 42. วิธีสร้างคอนสตรัคเตอร์ Constructors 43. วิธีสร้างคลาสเมธอด (Class Methods) 44. ทำความเข้าใจ Access Modifier
ครั้งที่ 8	วิธีเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming) (ต่อ) 45. วิธีเขียนโค้ดสำหรับคุณสมบัติ Encapsulation 46. วิธีสืบทอดคลาส (Inheritance) 47. วิธีเขียนโค้ดสำหรับคุณสมบัติ Polymorphism 48. ทำความเข้าใจเกี่ยวกับ Package 49. วิธีสร้างข้อมูลชนิด Enum 50. วิธีสร้าง Abstract Class 51. วิธีสร้าง interface 52. วิธีสร้าง Static Class และ Static Method 53. Exercises (แบบฝึกหัด)
***หมายเหตุ เนื้อหาอาจมีการเปลี่ยนแปลงตามความเหมาะสม	