

หลักสูตร C# Programming for Desktop Application Development

ภาพรวมของการเรียน

Desktop หรือ Windows Form Application เป็นแอปพลิเคชันที่ทำงานบนคอมพิวเตอร์เดสก์ท็อป โดยปกติ Desktop Application จะประกอบด้วย Controls เช่น Form, Label, Textbox, Button, ListBox, ComboBox ฯลฯ ภาษา C# เป็นภาษาหนึ่งของ Microsoft.NET ที่สามารถใช้พัฒนาโปรแกรม Desktop Application ฉะนั้นผู้ที่สร้าง Desktop Application บนแพลตฟอร์ม .NET ต้องเขียนภาษา C# หลักสูตรนี้จะปูพื้นฐานวิธีเขียนโปรแกรมภาษา C# ตั้งแต่เริ่มต้นจนถึงการเขียนโปรแกรมเชิงวัตถุ (Object-Oriented Programming) เพื่อให้ผู้เรียนสามารถพัฒนา Desktop Application และ ASP.NET ต่อไป หลักสูตรนี้เหมาะสำหรับโปรแกรมเมอร์ วิศวกร และนักศึกษาที่กำลังศึกษาในมหาวิทยาลัย

ระยะเวลาในการเรียน

หลักสูตร 30 ชั่วโมง เรียนทั้งหมด 10 ครั้ง (ครั้งละ 3 ชั่วโมง)

พื้นฐานของผู้เรียน

สามารถใช้ Windows และมีความรู้คอมพิวเตอร์เบื้องต้น

โปรแกรมที่ใช้

Visual Studio Community Edition กับ SQL Server Express Edition

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

ประโยชน์ที่จะได้รับจากหลักสูตร

- สามารถเข้าใจคำสั่งและเขียนโปรแกรม C# ได้อย่างถูกต้อง
- เข้าใจหลักวิธีเขียนโปรแกรมเชิงวัตถุหรือ Object-Oriented Programming
- สามารถสร้าง Windows Form Application ด้วย Windows Form Control ชนิดต่างๆ

- สามารถเขียนโปรแกรมติดต่อกับฐานข้อมูลด้วย ADO.NET

C# Programming Course Outline	
ครั้งที่ 1	<ol style="list-style-type: none"> 1. ทำความรู้จัก .NET Framework และ Common Language Runtime (CLR) 2. วิธีตั้งค่า Visual Studio Community Edition 3. เริ่มต้นเขียนโปรแกรม Console Application 4. สร้างตัวแปรและค่าคงที่ (Variables and Constant) 5. ชนิดของข้อมูล (Data Types) 6. วิธีเขียน Comment 7. ตัวดำเนินการ (Operators) <ul style="list-style-type: none"> • ตัวดำเนินการคณิตศาสตร์ (Arithmetic) • ตัวดำเนินการเปรียบเทียบ (Comparison) • ตัวดำเนินการตรรกะ (Logical) 8. วิธีแปลงชนิดของข้อมูล (Data Type Conversion)
ครั้งที่ 2	<ol style="list-style-type: none"> 9. วิธีเขียนโปรแกรมแบบวนลูป <ul style="list-style-type: none"> • วิธีใช้คำสั่ง for loop • วิธีใช้คำสั่ง while loop • วิธีใช้คำสั่ง do while loop • วิธีเขียนคำสั่ง break, continue, return 10. วิธีเขียนโปรแกรมแบบมีเงื่อนไข (Conditional Statement) <ul style="list-style-type: none"> • วิธีใช้คำสั่ง if...else if • วิธีใช้คำสั่ง Switch Case 11. คำสั่งของ C# <ul style="list-style-type: none"> • คำสั่งจัดการข้อความ (String) • คำสั่งด้านคณิตศาสตร์ (Math) • คำสั่งจัดการวันที่และเวลา (Date and Time) 12. วิธีใช้คำสั่ง foreach 13. วิธีสร้าง StringBuilder 14. การสร้างเมธอด (Method)

	<p>15. วิธีสร้างและใช้ข้อมูลชนิด Enum</p> <p>16. การตรวจหาและการจัดการข้อผิดพลาดในโปรแกรม (Debugging & Error Handling)</p>
<p>ครั้งที่ 3</p>	<p>17. วิธีเก็บข้อมูลชนิดอาเรย์ (Array)</p> <p>18. วิธีเก็บข้อมูลชนิดอาเรย์ลิสต์ (ArrayList)</p> <p>19. วิธีจัดการกับไฟล์และไดเรกทอรี (File, Path & Directory)</p> <p>20. ขอบเขตของตัวแปร (Scope of variable)</p> <p>21. public, private, protected</p> <p>22. อีเวนต์ (Events)</p> <ul style="list-style-type: none"> ● คีย์บอร์ดอีเวนต์ (Keyboard Events) ● เมาส์อีเวนต์ (Mouse Events)
<p>ครั้งที่ 4</p>	<p>วิธีวิธีเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming)</p> <p>23. คลาสและออบเจ็ค (Class, Object)</p> <p>24. ฟیلด์, เมธอด, พร็อพเพอร์ตี้ (Field, Method, Property)</p> <p>25. คอนสตรัคเตอร์ และโอเวอร์โหลด (Constructor, Overloading)</p> <p>26. this กับ Partial class</p> <p>27. คุณสมบัติของโปรแกรมเชิงวัตถุ</p> <p>28. การกำจัดข้อมูลที่ไม่เกี่ยวข้อง (Abstraction)</p> <p>29. การซ่อนข้อมูลในคลาส (Encapsulation)</p> <p>30. การสืบทอดคลาส (Inheritance)</p> <p>31. ความหลากหลายรูปแบบ (Polymorphism)</p> <p>32. วิธีสร้างเนมสเปซ (Namespace)</p>
<p>ครั้งที่ 5-6</p>	<p>33. วิธีสร้างและใช้ฟอร์มกับคอนโทรล (Windows Form Controls (1))</p> <ul style="list-style-type: none"> ● Message Box, Label, TextBox ● Button, CheckBox, RadioButton ● ListBox, ComboBox, NumericUpDown ● DateTimePicker, PictureBox, ProgressBar ● OpenFileDialog, FontDialog, ColorDialog ● MenuStrip, TabControl

	<ul style="list-style-type: none"> ● Panel, FlowLayoutPanel, TablePanel
ครั้งที่ 7-8	<p>34. วิธีเชื่อมฐานข้อมูลด้วย ADO.NET (1)</p> <ul style="list-style-type: none"> ● คำสั่ง SqlConnection ● คำสั่ง SqlCommand ● คำสั่ง SqlDataReader ● คำสั่ง SqlDataAdapter & DataSet ● คำสั่ง DataTable & DataRow <p>35. วิธีแสดงข้อมูลใน DataGridView</p> <p>36. คำสั่ง Data Binding</p> <ul style="list-style-type: none"> ● Simple DataBinding ● Complex DataBinding ● Binding Navigator
ครั้งที่ 9-10	<p>37. วิธีเชื่อมฐานข้อมูลด้วย ADO.NET (2)</p> <ul style="list-style-type: none"> ● วิธีบันทึกข้อมูลลงฐานข้อมูล (Insert data to database) ● วิธีอ่านข้อมูลจากฐานข้อมูล (Read data from database) ● วิธีแก้ไขข้อมูลในฐานข้อมูล (Edit data) ● วิธีลบข้อมูลในฐานข้อมูล (Delete data) <p>38. วิธีสร้างแอปพลิเคชัน (Deploying Windows Form Application)</p>