

## Objectives

At the end of this lab, you should be able to:

- What are boxing and unboxing
- Identify where boxing and unboxing takes place.
- Explain the meaning and use of *namespace* (*keyword*)
- There are two main types: *value type* and *reference type*.



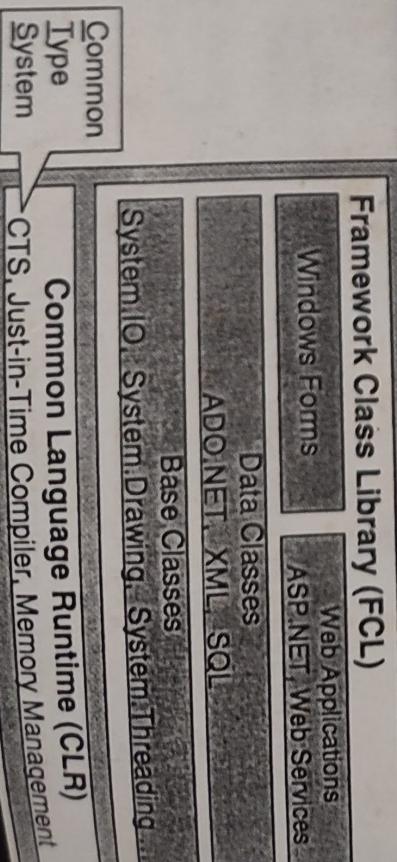
Side 1

© revised date: 31 Sep 2006

ACOOP - Advanced Object Oriented Programming

## .NET Platform = CLR + FCL

### Framework Class Library (FCL)





Duration: 2 hours

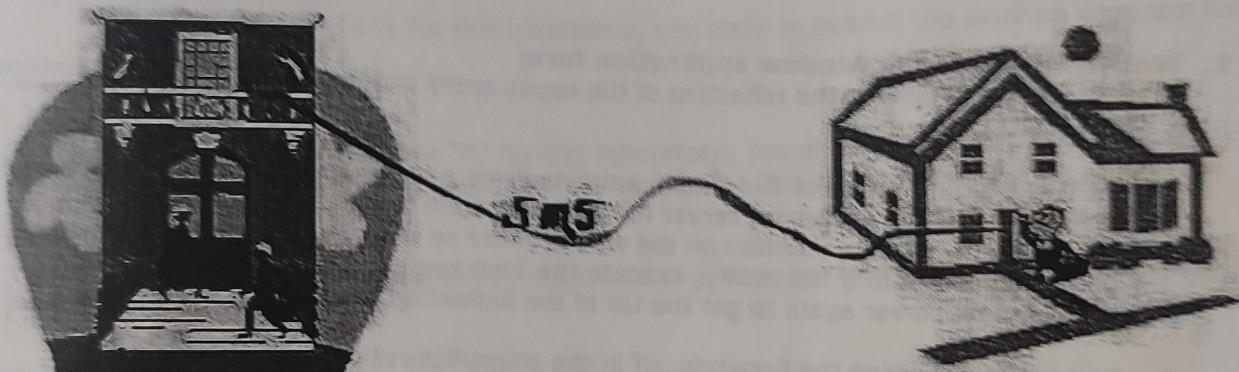
## Laboratory 1

### Objective

In this laboratory, you will

1. Reflect on the house entry window application (FOOP mini-project)
2. Create a simple animated web application
3. Perform web browsing in the house entry window application form and
4. Finally add to the simple animated web application a house entry web application to replace the house entry window application.
5. Implement 2 types of Validator controls.

### 1. A reflection of the house entry window application (Recall FOOP mini-project)

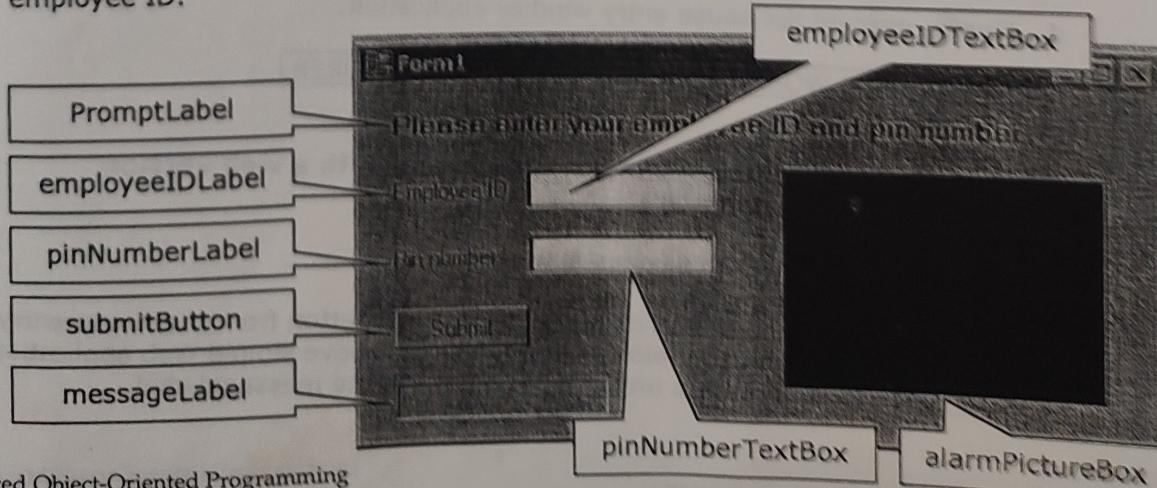


**Figure: A pictorial illustration of an application using pin number for house entry**

Don works in a company of 200 employees as a house salesman. In order to enter the house that he is selling, he needs to enter his pin number and his numerical employee ID. If he forgets his pin number, he could send a short message (SMS) to his main office. His main office would send him his pin number by SMS.

For this reflection exercise, the house entry window application would be significantly simplified. Only Don could make the house entry. His employee ID is 200 and pin number is 1234. You have to implement the following tasks for the house entry window application.

- a. Create the user interface (as shown below) for Don to submit his pin number and his employee ID.



# Module Overview

If you are amazed by the rich graphic and interaction of computer game, in FOOP we will lay down a good foundation of Object-Oriented Programming (OOP) for you to embark into the exciting world of programming.

This practice-oriented module equips you with the fundamental skills required to develop Windows applications using object-oriented programming language. You will develop conceptual understanding to design and develop applications to solve business and engineering problems. Main topics include class design, defining properties and methods and bitwise operations.

## Syllabus for Fundamentals of Object-Oriented Programming - 56 hours

### 1. Introduction to Object-Oriented Programming

- 1.1. Introduction to Microsoft .NET
- 1.2. Visual Studio .NET development environment.
- 1.3. Object-Oriented Programming
  - 1.3.1. Object model
  - 1.3.2. Structured Programming to Event Driven programming
  - 1.3.3. Moving from C to C#
- 1.4. Simple C# programming

### 2. Using Control Classes

- 2.1. Using TextBox control
- 2.2. Using Label Control
- 2.3. Using CheckBox Control
- 2.4. Using RadioButton Control
- 2.5. Using GroupBox control
- 2.6. Using PictureBox Control

### 3. Variables, Constants and Calculations

- 3.1. Predefined Types
- 3.2. Variables and scope
  - 3.2.1. Variable declaration
  - 3.2.2. Coding block
  - 3.2.3. Namespaces
  - 3.2.4. Scope
  - 3.2.5. Variable initialization
- 3.3. Implicit Type conversions and Explicit Casting

### 4. Decisions and Conditions

- 4.1. Flowchart
- 4.2. *if/else* Selection structure
- 4.3. *while* selection structure
- 4.4. *for* Repetition Structure
- 4.5. *switch* Multi-selection Structure
- 4.6. *do/while* Repetition Structure
- 4.7. Logical Operators

**Ngee Ann Polytechnic**  
**Electronic & Computer Engineering Division**  
**TEACHING PLAN (APRIL 2006 SEMESTER)**

**Module Title** : Fundamentals of Object-Oriented Programming

**Level** : 2

**Module Leader** : Ho Jen Chan/6166

**Module Lecturers** : Teo Tat Lee, Tan Sim Bee, Ma Ngok Sang, Auyong Lin Song, Yeo Meliana, Hee Juay Jiunn, Nordin Salleh, Ker Siong Nai, Soo Weng Kiong, Ho Jen Chan, Azman Johari, Wong Siew Cheong, Tan Boon Lee

Week No.	Week Begins	Lecture	Hr	Tutorial	Hr	Laboratory	Hr	Remarks Tests
1	24 Apr	Chp 1: Introduction to Object-Oriented Programming	4	Tut 1 OOP		Lab 1 OOP		
2*	1 May	Chp 2:Using Control Classes	3	Tut 2 Using Control Classes	1		0	
3**	8 May	Chp 3:Variables, Constants and Calculations	2		0	Lab 2 Using Control Classes	2	
4	15 May	Chp3 Con't	1	Tut 3 Variables, Constant and Calculations	1	Lab 3 Variables, Constant and Calculations	2	
5	22 May	Chp 4:Decisions and Conditions	3	Tut 4 Decisions and Conditions	1		0	
6	29 May	Revision	2		0	Lab 4 Decisions and Conditions	2	Theory Test 1
7	5 Jun	Common Test Week						
8	12 Jun	Term Break						
9	19 Jun	Term Break						
10	26 Jun	Chp 5:Methods	2		0	Trial test	2	
11	3 Jul	Chp 5 Con't	2	Tut 5 Methods	1	Lab Test	1	
12	10 Jul	Chp 6:Arrays	2		0	Lab 5 Methods	2	Lab retest (MC)
13	17 Jul	Chp 6 Con't	1	Tut 6 Arrays	1	Lab 6 Arrays	2	
14	24 Jul	Chp 7:Creating user-defined Classes	2		0	Lab 7 Mini-Project	2	
15	31 Jul	Chp 7 Con't	1	Tut 7 Classes	1	Lab 7 Con't	2	
16#	7 Aug	Revision	4					Theory Test 2
17	14 Aug	Study & Examination Week						
18	21 Aug	Examination Week						
19-25	28 Aug	Vacation (28 Aug – 15 Oct 06)						

\*Labour Day, Monday, 1 May 06

\*\*Vesak Day, Friday, 12 May 06

#National Day, Wednesday, 9 Aug 06

• Assessment weightage

Theory Test 1 = 10 % Theory Test 2 = 30% Lab = 40% Lab Test = 20 %

04/21/06