

IECS273/274 [1111 3670/3671] 物件導向設計與實習

01#1 這門課程上什麼？





01

▶ 課程概述

▶ 關於物件導向設計與實習這門課程

本課程著重在物件導向程式設計：

學習在物件導向設計原則下，使用Java程式語言來開發軟體。

- 本課程兼具程式設計的教學與實作演練。

透過程式設計的實作練習與指定作業，

讓大家了解物件導向程式設計的開發技巧。

- 本課程會再介紹物件導向設計語言UML的類別圖。

透過系統設計與程式開發的對應，

讓大家初步了解軟體工程的開發模式。

▶ 課程規劃內容(1/2)

周次	日期	規劃課程內容
1	2022.09.13	Object-Oriented Paradigm
2	2022.09.20	Java Fundamentals 1 - Simple Expression Statement
3	2022.09.27	Java Fundamentals 2 - Conditional & Iterative Statement
4	2022.10.04	OOP - Object & Class I
5	2022.10.11	OOP - Object & Class 2
6	2022.10.18	OOP - Inheritance
7	2022.10.25	OOP – Polymorphism
8	2022.11.01	TBD
9	2022.11.08	Midterm Exam

▶ 課程規劃內容(2/2)

周次	日期	規劃課程內容
10	2022.11.15	Java Programming 1 - Exception Handling, File I/O
11	2022.11.22	Java Programming 2 - Mutable and Immutable Objects
12	2022.11.29	Java Programming 3 - GUI Layout
13	2022.12.06	Java Programming 4 - Collections, Thread, Networking.
14	2022.12.13	Java Programming 5 - Interfaces, Generics Class
15	2022.12.20	Miscellaneous - UML Class Diagram
16	2022.12.27	TBD
17	2023.01.03	Final Exam
18	2022.01.10	-

▶ 參考資料

➤ 課程簡報與內容架構

- C. H. Huang, Introduction to Computer Programming (with Software Lab), FCU ISTM 2617/2618,

➤ Java 程式語言-參考書目與開發環境

- Y. D. Liang, *Introduction to Java Programming and Data Structures (11th Edition)*, Pearson
- Walter Savitch, *Absolute Java (6th Edition)*, Pearson
- Java SE Development Kit (18.0.1.1)
- Eclipse IDE (2022.3)

➤ 物件導向方法論-參考文獻與開發環境

- OMG Unified Modeling Language (OMG UML) Version 2.5.1
- StarUML (<https://staruml.io/>)



02

► 遊戲規則的確立

▶ 課程基本資訊

- Course Number: IECS 273/274 [1111 3670/3671]
- Course Title: Object Oriented Design (with Software Lab)
- Credit Hours: 3
- Room: Room 234 Information and Electrical Engineering Bldg.
(資電館 234)
- Teaching Assistants:
 - 粘敦焱(碩班) mail: owlmien9@gmail.com
 - 蘇唐為(專題生) mail: d0908831@o365.fcu.edu.tw

▶ 上課模式說明

■ Lecture Time : 18:20~20:00, Tuesday

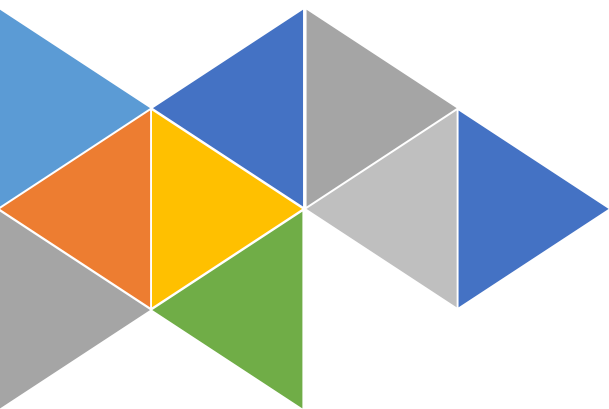
- ✓ iLearn 2.0 : 課程與實習重要公告
- ✓ Microsoft Teams : 課堂授課模式
 - ✓ 同學上課時段無法到課者，採線上參與(MS Teams)。
 - ✓ 老師因故無法到課，會先透過iLearn公告，以線上教學(MS Teams)為原則。

■ Lab Time : 20:00~21:40 , Tuesday

- ✓ 實習過程模擬團隊實際開發方式，每三到五人推選小組長一人，帶領同組同學完成練習或作業。
- ✓ 實習以每位同學為單位，將指定練習或作業傳送至iLearn 2.0平台。
- ✓ 指定練習或作業有問題，由TA和老師協助輔導。
- ✓ 因應疫情變化，無法到資電234上課者，請自備電腦並安裝相關開發環境。

▶ 學習評量方式

- 平時成績 40%
 - ✓ 點名 4%
(期中考前後各兩次)
 - ✓ 練習與作業 36% - Practice & Assignment
(每周實習課程均有練習，另外預計6次作業)
(繳交練習與作業 要有程式碼 註解 看到結果展示)
- 期中考 30%
(Open Book & 上機 & LAN)
- 期末考 30%
(Open Book & 上機 & LAN)



IECS273/274 [1111 3670/3671]

物件導向設計與實習

01#2 物件導向開發設計





01

► 開發模式的典範轉移

▶ What Is Computational Thinking?

■ Three stages of information literacy development:

- Before the 1980s, almost only computer experts or engineers had access to computers; ordinary people were always out of reach for computers, and information literacy was completely blank for the general public.
- After the 1980s, with the advent of personal computers, the general public began to have the opportunity in accessing and using computers. At this stage, Microsoft's Office gradually became a tool software used in many daily work. The need for information literacy has shifted from learning Chinese and English keyboard input to the use of packaged software such as word processing, spreadsheets, and slideshows.
- Since the 1990s, Internet technology has risen rapidly until it is ubiquitous; the advent of the Internet era has also improved the information literacy of the general public. The popularity of smart phones and mobile Internet has made many innovative services change our way of life.
- Reference: <https://computationalthinkingcourse.withgoogle.com/unit>

► What Is Computational Thinking?

- In recent years, with the advancement of high technology, we often see some terms in the field of new technology in the mass media:
 - For example, cloud computing, Internet of Things, big data analysis, deep learning, and artificial intelligence.
- Not to mention the technical details of these fields, the point is that they will bring great changes to life and work in the future.
 - These changes include smart home, smart entertainment, smart manufacturing, smart business, smart agriculture, smart healthcare, smart city, etc.
- Faced with the intelligence of all aspects of life, advanced countries are actively investing resources to improve information literacy and cultivate the ability of people from all aspects of life to write programs, so as to integrate into various intelligent technology development and technology applications as soon as possible.

▶ What Is Computational Thinking?

- For students in various fields, the purpose of learning computer programming is not necessarily to become a programming master.
 - Instead, we hope that you who enter the workplace in the future will have the basic ability to engage in intelligent work. Learning programming involves two main concepts: programming languages and computational thinking.
 - To put it simply, a programming language is a language for communicating with a computer, that is, a "speech" that the computer "sees" or "listens" to understand.
 - In fact, the design of the programming language also reflects the basic architecture of computer hardware, which will be explained in detail in the following units.
- Regardless of the field, learning a programming language has several advantages:
 - Improve personal information literacy and competitiveness.
 - Improve personal work efficiency and productivity.
 - Enhance personal cross-domain digital innovation and intelligent application capabilities.
 - Improve personal international mobility and create personal value.

► Computational Thinking - Four Keys

- Computational thinking is the thinking mode of solving problems on computers. This mode of thinking is not only for computers, but also extremely important in daily life and work.
- Four keys to computational thinking
 - **Problem Decomposition**
 - **Pattern Recognition**
 - **Abstract Thinking**
 - **Algorithm Design**

► Computational Thinking - Four Keys

- **Problem Decomposition:** In the problem-solving method, "major things are reduced to small ones" is a means of reducing complexity to simplicity.
 - In other words, decomposing difficult or complex problems into smaller problems, and then solving these small problems separately, is the basic spirit of decomposition.
 - For example, the company wants to organize a product launch. Faced with this problem, companies need to clearly list:
 - ✓ published product items and forms
 - ✓ date and place
 - ✓ manpower and funding
 - ✓ expected results of the presentation, etc.
 - These problems are to divide further into many small problems, and then solve these small problems separately. Of course, it is possible for each small problem to be broken down into smaller problems; for example,
 - ✓ In terms of manpower, it can be further divided into personnel selection, work assignment, job scheduling, etc.

▶ Computational Thinking - Four Keys

- **Pattern Recognition:** imitation is an important way to solve problems, and computational thinking is no exception.
 - To imitate is to seek solutions based on other people's or one's own previous experience.
 - In the process of imitation, it is very important to be able to judge what experience is related or similar to the problem to be solved, that is to say, the pattern of the problem can be **effectively summarized, classified, and identified**.
 - For example, someone is going to make a stir-fried pork with kimchi.
 - ✓ Base on the ingredients and cooking methods of this dish based on past experience, such as stir-fried beef with green peppers is past experience.
 - ✓ First, the fried method has been identified. Although pork and beef are different, kimchi and green pepper are not the same, but applying the method of stir-fried beef with pepper is obviously helpful to make the dish of stir-fried pork with kimchi.
 - ✓ Pattern recognition is to find the similarity and context of problems, classify them, and then develop solutions.

▶ Computational Thinking - Four Keys

- **Abstract Thinking:** thinking about things or things in a conceptual way.
 - Concrete thinking is the opposite of abstract thinking.
 - ✓ Specific thinking is based on specific objects or events, such as John, a student who actually thinks about events that occur in the course of class.
 - ✓ Abstract thinking is based on the abstract concept of "students", thinking about events that may occur in the course of class.
 - ✓ From the point of view of computer digitization, the abstract concepts of students can be represented by student numbers, names, and classes, which are called attributes of students.
 - ✓ Of course, different application purposes can also add other attributes according to their needs, such as birthday, height, weight, etc., or add student photos, video records, etc.
 - ✓ To put it simply, abstract thinking is to "generalize" things, so that more general solutions can be found in the process of thinking.

▶ Computational Thinking - Four Keys

- **Algorithm Design:** Algorithms refer to stepwise procedures to solve a problem.
 - In general, algorithms must be logical, otherwise steps cannot be established.
 - For example: the cooking process is an algorithm
 - ✓ First, the ingredients and quantities to be prepared will be described before cooking, and then each step of cooking will be described.
 - ✓ From a computer program's point of view, ingredients and portion sizes are the initial conditions of the algorithm; cooking steps are the process of program execution.
 - ✓ Also, the algorithm must be logical.
 - For example, to make a dish of Kung Pao Chicken, you must first heat the oil, add the ingredients and sauté until fragrant, then add chicken and cooking wine to make it dry.
 - If someone puts chicken and all the ingredients into the pot and stir fry, then the resulting dish is obviously not Kung Pao chicken, and the taste will be much worse.
 - Algorithms can generally be expressed in **flow charts**.

► Understand Programming Language

- Usually the communication between people is carried out through language, and the language of communication between people is called **natural language**.
 - When two people are talking, if one of them only speaks Chinese and another one speaks English, then they cannot communicate with each other.
- If a person wants to communicate with a computer or make a computer "do things", he/she needs to use a language that the computer can understand. The language of the computer is called the **programming language**.
- The development of programming languages is divided into three programming paradigms:
 - **Imperative programming**: e.g., FORTRAN, COBOL, Algol68, PASCAL, C, C++, Java, etc.
 - ✓ **Object-oriented programming**: is an extension of imperative programming, e.g., smalltalk, Modula-2, C++, C#, Java
 - **Functional programming**: e.g., LISP, ML, Miranda, etc. °
 - **Logic programming**: e.g., Prolog, Datalog, etc.

► Understand Programming Language

- Different programming paradigms affect problem-solving thinking patterns:
 - Imperative programming: computer operations are similar to computer hardware executing instructions
 - ✓ Most high-level imperative programming languages: variables, arithmetic operations, logical operations, assignment statements, iterative/loop statements, conditional branch statements (selection/conditional statement), unconditional branching statement
 - Functional programming: computer operations are similar to mathematical function mapping operations
 - ✓ "Pure" functional programming languages: arithmetic operations, list operations, function definitions (especially recursive functions), function application
 - Logic programming: computer operations are similar to mathematical logical reasoning operations
 - ✓ Logical programming languages: facts, logical rules

► Object-Oriented Programming

- Object-oriented programming is an extension of imperative programming
 - Every object-oriented programming language contains the language building blocks of imperative programming: variables, arithmetic operations, logical operations, assignment statements, selection/conditional statements, iterative/loop statement.
 - In object-oriented programming, a collection of elements of the same nature is called a **class**, and a specific element in a class is called an **object**.
 - ✓ Students are a class, and specific students such as John and Mary are individual objects ◦
 - Object-oriented programming languages have the following characteristics:
 - ✓ **Abstraction**: Turning complexity into simplicity, divided into **data abstraction** and **procedure abstraction**.
 - ✓ **Encapsulation**: information hiding of specific implementation details of data and methods; use of **public**, **private**, and **protected** members to distinguish access restrictions.

► Object-Oriented Programming

- ✓ **Dynamic binding:** object-oriented function (function) is called **method**, function call is called **message passing**; when the program is executed, the method changes with different **objects** do dynamic binding.
- ✓ **Inheritance:** A class can have an **inheritance relation** between a **parent class** and a **child class** of a **tree structure**.
 - Students form a parent class, and students of International School of Technology and Management (ISTM), and students of College of Information and Electrical Engineering (CIEE) are subclasses.
- ✓ **Polymorphism** refers to the related different subclasses generated by inheritance, whose objects will respond differently to the same message.
 - For example, Shape is a class that has three subclasses: Square, Rectangle, and Triangle. When you want to calculate the area of an object obj of a shape, obj.area(), Program execution will call the method of the relevant area method according to the actual subclass of the object to **pass the correct message**.

▶ 從Object World, SF, CA, 1994談起

Jacobson, *OOSE*



Requirement
Elicitation

Rumbaugh, *OMT*



Object-oriented
Analysis

Booch, *OOD*



Object-oriented
Design













02

► 程式語言的江湖

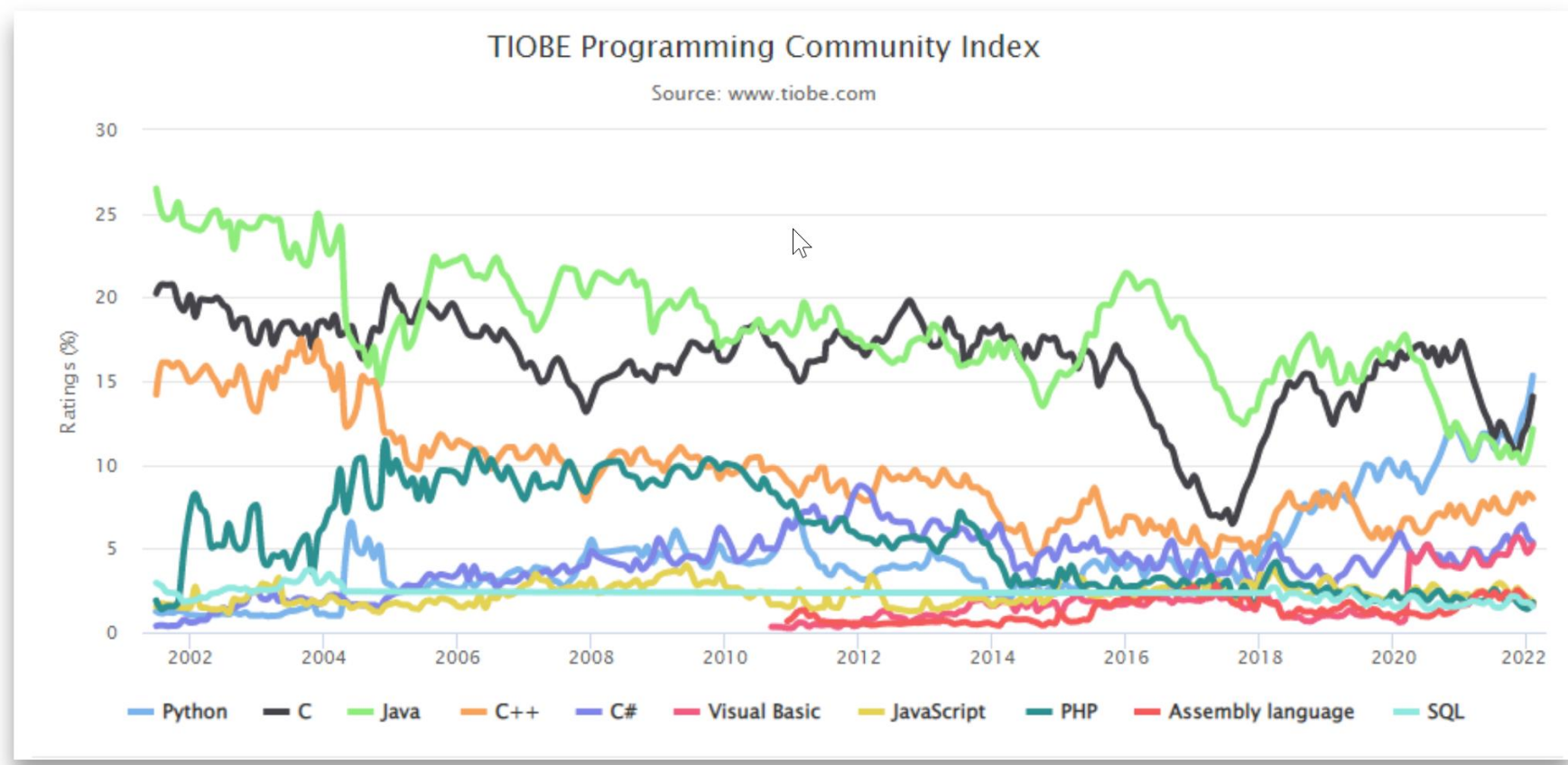
► Programming Languages



► 2022 TIOBE 程式語言排名

Feb 2022	Feb 2021	Change	Programming Language		Ratings	Change
1	3	▲		Python	15.33%	+4.47%
2	1	▼		C	14.08%	-2.26%
3	2	▼		Java	12.13%	+0.84%
4	4			C++	8.01%	+1.13%
5	5			C#	5.37%	+0.93%
6	6			Visual Basic	5.23%	+0.90%
7	7			JavaScript	1.83%	-0.45%
8	8			PHP	1.79%	+0.04%
9	10	▲		Assembly language	1.60%	-0.06%
10	9	▼		SQL	1.55%	-0.18%

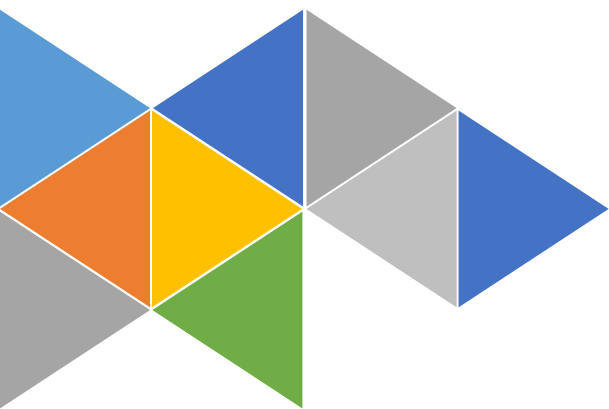
▶ 程式語言使用趨勢



▶ 程式語言-Java

- Java 是一個優秀的語言，它最大的特點就是跨平台，不論 Mac、Windows、Linux，程式碼長的都一樣。
- Java的開發者對於C++非常熟悉，所以改善了C++的缺點，但是結構上還是非常相似。
- 優點
 - ✓ 跨平台、相容性高、安全性高
 - ✓ 可以做各式各樣的應用，像是架設網站、網頁前後端、GUI開發、開發 Android APP、物聯網應用程式、大數據分析、金融業的帳務處理、桌面應用程式等等
- 缺點
 - ✓ 為了改善 C++ 的便利性，造成執行速度緩慢
 - ✓ 編譯過程繁瑣
 - ✓ 停止更新免費版本





IECS273/274 [1111 3670/3671] 物件導向設計與實習

01#3 Java程式開發初探





01

► Java程式初步剖析

► The First Java Program: HelloWorld

Every Java program is a **class**.

Class name: **HelloWorld**,
The file name should be the same as the class name HelloWorld.java, and the file extension must be **"*.java"**.

There must be a static main program with the method name **"main"**.

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

The main program parameter is a string array "String[]".

Java string output statement.

The name of the execution file after compilation is **"HelloWorld.class"**.



02

► 整合開發環境的建立

► Integrated Development Environment

- Developing a program in a programming language requires appropriate tools, the most basic being a **text editor** and a **programming language compiler**.
- Many software developers design and develop various convenient and multi-functional software systems for users to easily write, compile, and test various programs. Such software systems are collectively referred to as **Integrated Development Environment (IDE)**.
- This course will use Eclipse as the base of the IDE, please follow the steps below to set up the IDE on your computer.
 - Visit [Java SE Development Kit Downloads](#) to download and install the development kit for the Java programming language.
 - ✓ Choose the appropriate version for your computer: Linux, macOS, or Windows version.
 - Visit [Eclipse IDE for Java Developers](#) to download and install the Eclipse development kit.
 - ✓ Choose the appropriate version for your computer: Linux, macOS, or Windows version.



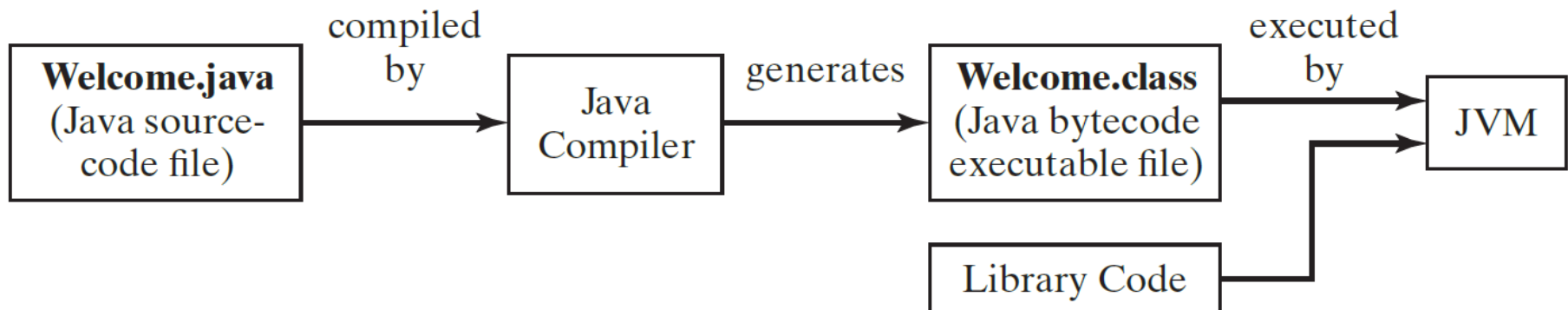
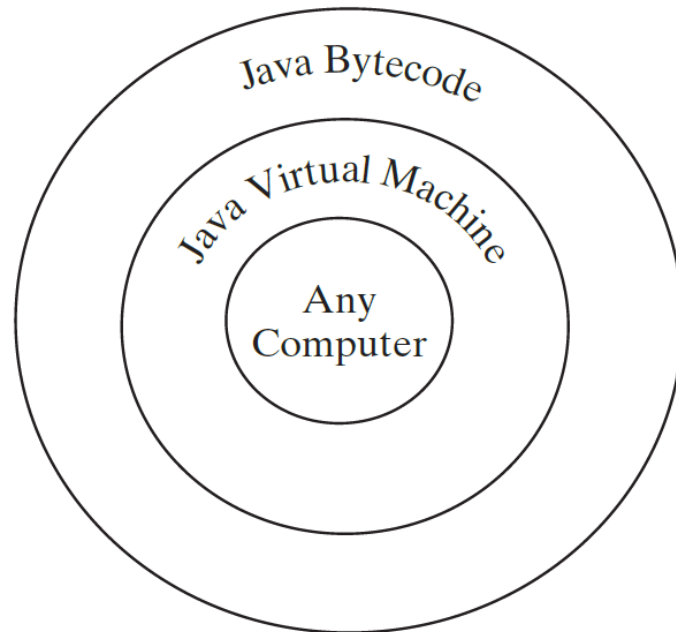
03

► 寫下你的第一支Java程式

▶ Java Development Procedure

■ Java programs normally go through the following phases

- edit
- compile
- load
- execute



Developing Java Program using Eclipse

- Creating a Java Project
- Creating a Java Class
- Editing a Java Program
- Compiling and Running a Class



①

Creating a Java Project

The screenshot shows the 'New Java Project' dialog box in the Eclipse IDE. The title bar reads 'New Java Project'. Below the title, it says 'Create a Java Project' and 'Create a Java project in the workspace or in an external location.' with a folder icon. The 'Project name' field contains 'FCUI ECS'. The 'Use default location' checkbox is checked, and the 'Location' is 'C:\Users\TTW\eclipse-workspace\FCUI ECS' with a 'Browse...' button. The 'JRE' section has three options: 'Use an execution environment JRE:' (selected) with a dropdown showing 'JavaSE-17', 'Use a project specific JRE:' with a dropdown showing 'jdk-17.0.1', and 'Use default JRE 'jdk-17.0.1' and workspace compiler preferences' with a 'Configure JREs...' link. The 'Project layout' section has two options: 'Use project folder as root for sources and class files' and 'Create separate folders for sources and class files' (selected) with a 'Configure default...' link. The 'Working sets' section has an 'Add project to working sets' checkbox, a 'Working sets:' dropdown, and 'New...' and 'Select...' buttons. The 'Module' section has a 'Create module-info.java file' checkbox which is checked. At the bottom, there are buttons for '< Back', 'Next >', 'Finish', and 'Cancel', along with a help icon.

New Java Project

Create a Java Project

Create a Java project in the workspace or in an external location.

Project name: FCUI ECS

☒ Use default location

Location: C:\Users\TTW\eclipse-workspace\FCUI ECS [Browse...](#)

JRE

☒ Use an execution environment JRE: JavaSE-17

☐ Use a project specific JRE: jdk-17.0.1

☐ Use default JRE 'jdk-17.0.1' and workspace compiler preferences [Configure JREs...](#)

Project layout

☐ Use project folder as root for sources and class files

☒ Create separate folders for sources and class files [Configure default...](#)

Working sets

☐ Add project to working sets [New...](#)

Working sets: [Select...](#)

Module

☒ Create module-info.java file

[?](#) < Back Next > Finish Cancel

Choose *File, New, Java Project* to display the New Project wizard.



② Creating a Java Class

New Java Class

Java Class

⚠ The use of the default package is discouraged.

Source folder: FCUI ECS/src Browse...

Package: (default) Browse...

☐ Enclosing type: Browse...

Name: welcome

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static
☒ none ☐ sealed ☐ non-sealed ☐ final

Superclass: java.lang.Object Browse...

Interfaces: Add...
Remove

Which method stubs would you like to create?

☐ public static void main(String[] args)
☐ Constructors from superclass
☒ Inherited abstract methods

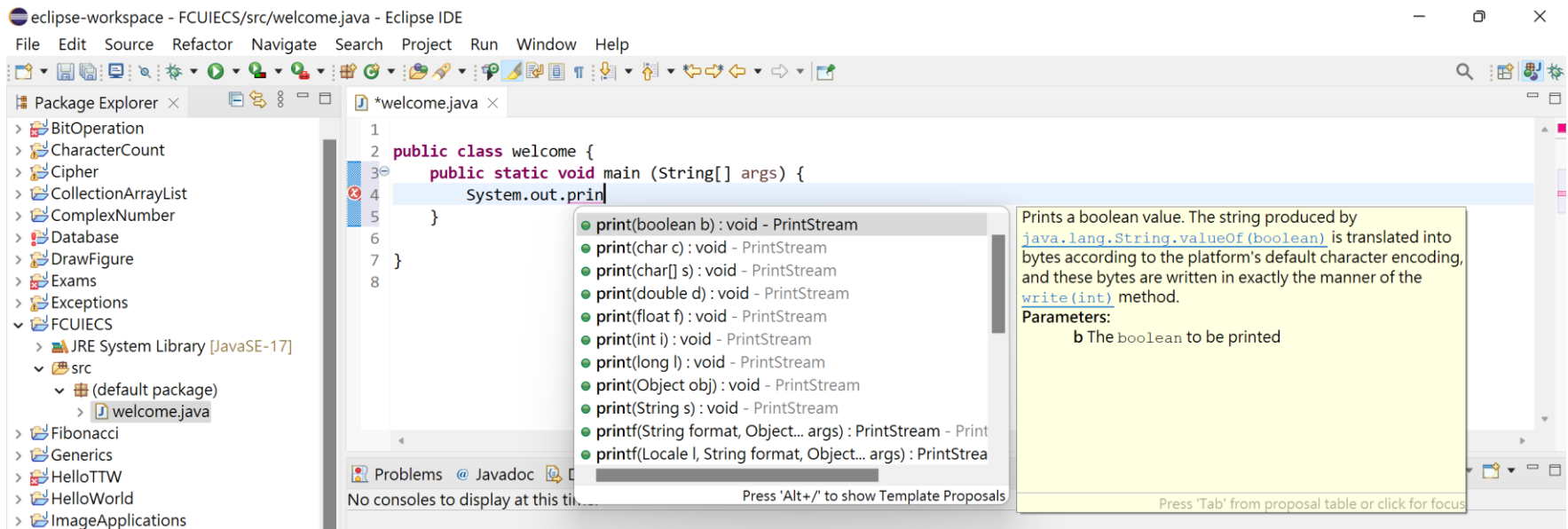
Do you want to add comments? (Configure templates and default value [here](#))
☐ Generate comments

? Finish Cancel

Choose *File, New, Class* to display the New Java Class wizard.

③ Editing a Java Program

- Appropriate Comments and Comment Styles
- Proper Indentation and Spacing
- Block Styles
 - The next-line style aligns braces vertically and makes programs easy to read, whereas the end-of-line style saves space and may help avoid some subtle programming errors.



▶ ④ **Compiling and Running a Class**

■ Syntax Errors

- Errors that are detected by the compiler are called syntax errors or compile errors.
- Syntax errors result from errors in code construction.

■ Runtime Errors

- Runtime errors are errors that cause a program to terminate abnormally.

■ Logic Errors (Semantic Errors)

- Logic errors occur when a program does not perform the way it was intended to.



eclipse-workspace - FCUI ECS/src/welcome.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Package Explorer

- > BitOperation
- > CharacterCount
- > Cipher
- > CollectionArrayList
- > ComplexNumber
- > Database
- > DrawFigure
- > Exams
- > Exceptions
- ✓ FCUI ECS
 - > JRE System Library [Java]
 - ✓ src
 - ✓ (default package)
 - > welcome.java
- > Fibonacci
- > Generics
- > HelloTTW
- > HelloWorld
- > ImageApplications
- > Images
- > ImmutableObject
- > InnerClass

welcome.java

```
1
2 public class welcome {
3     public static void main (String[] args) {
4         System.out.println("Welcome to FCU IECS 273");
5     }
6 }
7
```

Problems @ Javadoc Declaration Console

<terminated> welcome [Java Application] C:\Program Files\Java\jdk-17.0.1\bin\javaw.exe (2022年9月
Welcome to FCU IECS 273

Writable Smart Insert



04 ▶ Programming Practice

▶ 練習：程式設計是一種解決問題的工程與藝術

```
FFFFFFFF CCCCCCCC UU    UU    II  EEEEEEEEE CCCCCCCC SSSSSSSS
FF        CC          UU    UU    II  EE          CC          SS      SS
FF        CC          UU    UU    II  EE          CC          SS
FFFFFFF   CC          UU    UU    II  EEEEEEE   CC          SSSSSSSS
FF        CC          UU    UU    II  EE          CC          SS
FF        CC          UU    UU    II  EE          CC          SS
FF        CC          UU    UU    II  EE          CC          SS      SS
FF        CCCCCCCC  UU UUUU    II  EEEEEEEEE CCCCCCCC SSSSSSSS
```

```
DDDDDD 11 0000000 2222222 11 0000000 9999999
DD DD 11 00 00      22 11 00 00 99 99
DD DD 11 00 00      22 11 00 00 99 99
DD DD 11 00 00      222 11 00 00 9999999
DD DD 11 00 00      22 11 00 00 99
DD DD 11 00 00      22 11 00 00 99
DD DD 11 00 00      22 11 00 00 99
DDDDDD 11 0000000 2222222 11 0000000 99999
```

逢甲資工
學號
姓名
其他創作……

[請熟悉整合開發環境 先傻傻地用複製貼上的暴力法 解決印出上述字樣的問題]
[這個結果將是期末以前一個OOP練習的題目]