

# Chapter 0.

## About the Course



2020-2021

COMP2396 Object-Oriented Programming and Java

Dr. T.W. Chim (E-mail: [twchim@cs.hku.hk](mailto:twchim@cs.hku.hk))

Department of Computer Science, The University of Hong Kong



# About us

## Instructor:

Dr. T.W. Chim

E-mail: [twchim@cs.hku.hk](mailto:twchim@cs.hku.hk)

Office: HW519

Phone: 28578272

## Teaching Assistants:

Mr. Wong Kwan Ho Marco

E-mail: [khwong@cs.hku.hk](mailto:khwong@cs.hku.hk)

Office: CB319

Phone: 28578263

Miss Wen Jing

E-mail: [joywen@connect.hku.hk](mailto:joywen@connect.hku.hk)

Office: TBA

Phone: TBA

## Student Teaching Assistants:

Mr. Liu Yunhao

E-mail: [davidliu@connect.hku.hk](mailto:davidliu@connect.hku.hk)

Mr. King Min Hao, Martin

E-mail: [martin24@connect.hku.hk](mailto:martin24@connect.hku.hk)

Mr. Wang Hanlin

E-mail: [u3560396@connect.hku.hk](mailto:u3560396@connect.hku.hk)



# What is Java?

- Originally developed by James Gosling in 1994
  - He created the original design of Java and implemented the language original compiler and virtual machine





Java™

# Language Platforms

- Java Platform, Standard Edition (Java SE)
  - Defines everything from the basic types and objects of the Java programming language to high-level classes that are used for networking, security, database access, graphical user interface (GUI) development, and XML parsing. It is mostly used to develop client-side standalone applications or applets.
- Java Platform, Enterprise Edition (Java EE)
  - Built on top of the Java SE platform.
  - Provides an API and runtime environment for developing and running large-scale, multi-tiered, scalable, reliable, and secure network and server-side applications (including Java servlets and Java Server Pages).



Java™

# Language Platforms

- Java Platform, Micro Edition (Java ME)
  - Provides an API and a small-footprint virtual machine for running Java programming language applications on small devices, like mobile phones.
  - A subset of the Java SE API, along with special class libraries useful for small device application development.
- JavaFX
  - Use hardware-accelerated graphics and media engines to take advantage of higher-performance clients and a modern look-and-feel as well as high-level APIs for connecting to networked data sources.

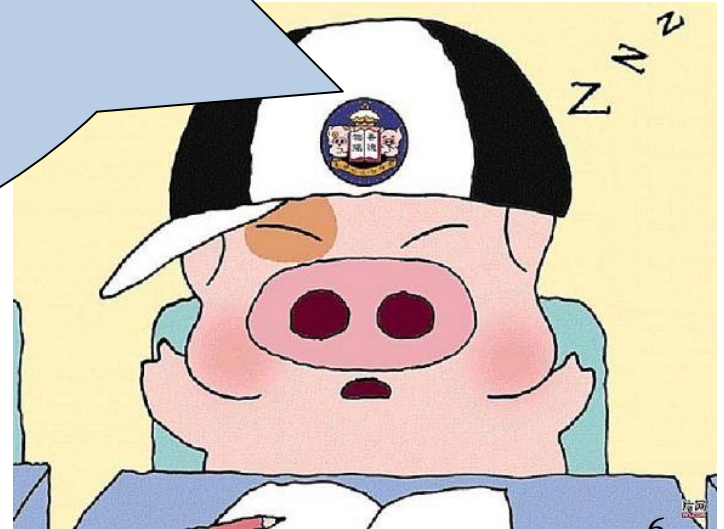
**This course will focus on Java SE!**



# Java™ Usefulness of COMP2396

**CS Graduate 2013**

The reason why I took this course is because of OOP. I think this is a very important concept in CS. So, it is very useful. I think it is useful to career too. For example, when I write the app, OOP concept takes a very important role in quality of programming.







# Java™ Usefulness of COMP2396

Of course, student should know Java before graduate: (1) when you master Java, it means that you have also learnt another language - C#. My opinion is: student should be able to handle at least C++ & Java first, most other languages are quite similar to them; (2) OOP concept is very important in a development project with other teammates, it really helps for developer's understanding about the codes from the others in a short time, and make the development project being more schedulable - since we can separate the project into several jobs easily.

**CS Graduate 2012**

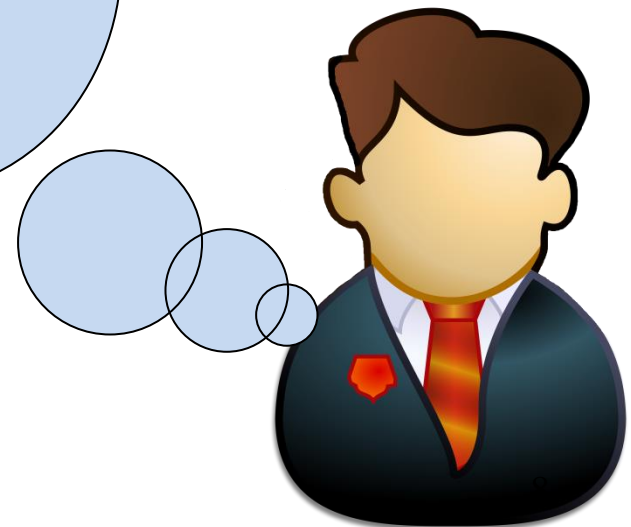




# Java™ Usefulness of COMP2396

**CS Graduate 2011**

I think 0396 is useful because I am using Java in my daily work. Strong foundation and understanding of OOP and Java (beginners and intermediate level) are important to my career. I am a developer of a web-based system. The UI is built by HTML5 and server is Java.







# Java™ Usefulness of COMP2396

**CS Graduate 2011**

Yes, very useful. I did J2EE job after graduate. Need to know Java. I think HKU graduates are weak in programming skills. They better take more programming courses.





# Java™ Usefulness of COMP2396

- **Relation with something in the market:**
  - Java language is similar to C# language which is used widely in game engine software like Unity.
  - Java is one of the languages supported by Android Studio, which is used to write Android applications.
  - Object-oriented programming concept is important if you want to write iOS applications (using Objective-C / Swift).

# Market share of Programming Languages

Worldwide, Aug 2020 compared to a year ago:

Rank	Change	Language	Share	Trend
1		Python	31.59 %	+3.3 %
2		Java	16.9 %	-2.7 %
3		Javascript	8.17 %	+0.0 %
4		C#	6.54 %	-0.7 %
5	↑	C/C++	5.88 %	+0.1 %
6	↓	PHP	5.78 %	-0.7 %
7		R	4.18 %	+0.3 %
8		Objective-C	2.6 %	-0.0 %
9		Swift	2.35 %	-0.0 %
10	↑	TypeScript	1.94 %	+0.2 %
11	↓	Matlab	1.63 %	-0.2 %
12		Kotlin	1.57 %	+0.1 %
13	↑↑	Go	1.39 %	+0.2 %
14	↓	Ruby	1.22 %	-0.2 %
15	↓	VBA	1.19 %	-0.1 %

16		Scala	0.97 %	-0.1 %
17	↑	Rust	0.91 %	+0.3 %
18	↓	Visual Basic	0.82 %	-0.2 %
19	↑↑↑↑↑	Dart	0.57 %	+0.2 %
20	↑↑↑	Ada	0.54 %	+0.2 %
21	↑	Lua	0.52 %	+0.1 %
22	↓↓↓	Perl	0.45 %	-0.1 %
23	↓↓↓	Abap	0.44 %	-0.1 %
24	↑↑↑	Julia	0.43 %	+0.2 %
25		Cobol	0.42 %	+0.1 %
26	↓↓↓↓↓	Groovy	0.41 %	-0.1 %
27	↓	Haskell	0.32 %	+0.0 %
28		Delphi	0.28 %	+0.0 %

© Pierre Carbonnelle, 2020

Reference: <http://pypl.github.io/PYPL.html>



# Learning Outcomes

## Object-oriented Programming

Be able to understand better the object-oriented approach in programming.  
Students should be able to analyze and design a computer program to solve real world problems based on object-oriented principles.

## Good Documentation Practice

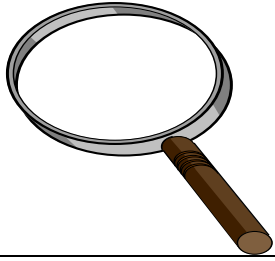
To learn and appreciate the importance and merits of proper comments in source code and API documentations

## Java Programming

Be able to write computer programs to solve real world problems in Java

## GUI Programming

Be able to write simple GUI interfaces for a computer program to interact with users, and to understand the event-based GUI handling principles.



# Syllabus

## Object-oriented Programming

- OOP overview
- Abstraction, encapsulation, and information hiding
- Inheritance (Interface) and polymorphism (Java array and array list)

## Good Documentation Practice

- Program documentation: JavaDoc

## Java Programming

- Java heap and garbage collection, Java math library, Java exception, Java package
- Java IO package and object serialization, Java network programming
- Collection class and iteration protocol

## GUI Programming

- GUI control and GUI libraries
- Java AWT and Java Swing: event handling, Layout manager, Java Applet



# Course Structure



Lectures  
(2 hrs per week)

Consultation



Tutorials (1 hr per week)

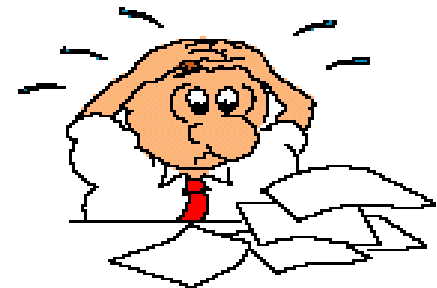


4 Programming  
Assignments  
(40%)



Programming Mid-term  
Examination (10%)

Written / Programming  
Final Examination  
(50%)







# Course Structure

Date	Tue (Online) 12:30pm – 1:20pm
19-Jan	Lecture 1
26-Jan	Tutorial 1
2-Feb	Tutorial 2
9-Feb	Tutorial 3
16-Feb	Chinese New Year
23-Feb	Tutorial 4
2-Mar	Tutorial 5
9-Mar	Reading Week
16-Mar	Tutorial 6
23-Mar	Tutorial 7
30-Mar	Tutorial 8
6-Apr	Easter Holiday
13-Apr	Tutorial 9
20-Apr	Tutorial 10
27-Apr	Tutorial 11

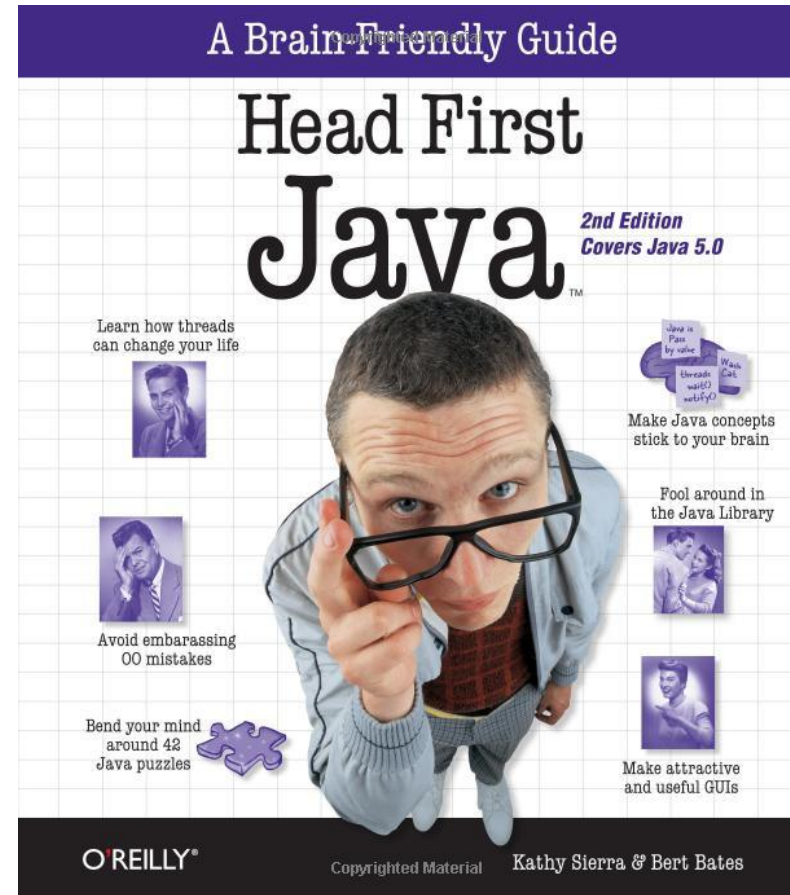
Date	Fri (Online) 12:30pm – 2:20pm
22-Jan	Lecture 2
29-Jan	Lecture 3
5-Feb	Lecture 4
12-Feb	Chinese New Year
19-Feb	Lecture 5
26-Feb	Lecture 6
5-Mar	Lecture 7
12-Mar	Reading Week
19-Mar	Mid-term Examination
26-Mar	Lecture 8
2-Apr	Easter Holiday
9-Apr	Lecture 9
16-Apr	Lecture 10
23-Apr	Lecture 11
30-Apr	Lecture 12



# Reference Book

- Head First Java: A Brain-Friendly Guide, 2<sup>nd</sup> Ed, by K. Sierra & Bert Bates, O'Reilly, 2005.
- Out of stock with no reprinting schedule at this moment.

*Total number of pages: 688*



# Chapter 0.

# End



2020-2021

COMP2396 Object-Oriented Programming and Java

Dr. T.W. Chim (E-mail: [twchim@cs.hku.hk](mailto:twchim@cs.hku.hk))

Department of Computer Science, The University of Hong Kong