

讀書會

- 組員：龍品勳、張耀文、時顯文
- 討論時間：2024/10/28 10:00 AM
- 地點：圖書館



- 照片：

心得報告

- 心得：陣列的宣告方式還有排序的方法和 C++ 稍有不同，需要多一點的時間去理解。
- ChatGPT：
 1. 請比較 Java 及 C++ 的區別。

Java 和 C++ 是兩種強大的編程語言，各有特點和用途，這裡是它們的主要區別：

➤ 語言設計和應用領域

- Java：是一種面向對象的語言，設計目的是跨平台。Java 通過 JVM（Java Virtual Machine）運行，使其可以在多個平台上無縫執行。因此，Java 在企業應用、移動開發（如 Android）、網頁應用和服務器端開發中使用較廣泛。

- C++：是一種多重範式語言，支持面向對象和過程式編程，並允許低層次

的系統控制。它的執行速度快，因而常用於系統軟件、遊戲引擎、嵌入式系統和效能要求較高的應用。

➤ 編譯與執行

- Java：源代碼被編譯為字節碼（Bytecode），然後通過 JVM 執行。這種虛擬機機制使 Java 有較好的跨平台性，但運行效率較低。

- C++：源代碼直接編譯成機器碼，並直接在硬體上執行，因此速度較快，但不同平台可能需要重新編譯。

➤ 記憶體管理

- Java：採用自動垃圾回收機制（Garbage Collection），在某些特定條件下自動回收未使用的內存，減少了內存洩漏的風險。

- C++：需要手動管理記憶體，使用 *new* 和 *delete* 來分配和釋放內存。這雖然增加了控制靈活性，但也更容易出現內存洩漏和崩潰的風險。

➤ 語法與特性

- Java：不支持指針，並且不允許直接進行內存操作。Java 中所有數據都是通過引用來操作的，因此比較安全，但靈活性有限。

- C++：支持指針和指針運算，允許直接操作內存。C++ 支持多重繼承和函數指針，使得其靈活性更強，但也增加了複雜度和潛在錯誤。

➤ 繼承與多重繼承

- Java：支持單一繼承，即每個類只能繼承一個超類，但可以通過實現多個接口來達到類似效果。

- C++：支持多重繼承，可以繼承多個基類，雖然靈活性更高，但會導致“菱形繼承”等複雜問題。

➤ 函數庫與支持

- Java：擁有豐富的標準庫和第三方庫支持，適合快速開發，並且有強大的框架支持（如 Spring、Hibernate）。

- C++：雖然 C++ 標準庫相對有限，但可以訪問強大的 C 標準庫，並且擁有 STL（標準模板庫），方便操作容器、算法等。

➤ 例外處理（Exception Handling）

- Java：有內建的例外處理機制，並強制處理某些檢查型例外（如 IOException），使得代碼更安全。

- C++：支持例外處理，但不強制使用，也沒有檢查型例外，因此開發者需要自行負責處理。

➤ 多執行緒處理

- Java：內建多執行緒支持且擁有豐富的多執行緒類庫，便於開發多執行緒應用。

- C++：雖然 C++11 開始支持多執行緒標準，但相比 Java，開發多執行緒程式的複雜度較高。

總結

Java 更注重跨平台、簡單易用和安全性，適合快速開發企業應用和網路應用。而 C++ 側重於靈活性和高效能，更適合系統級別的應用和資源受限的環境。

2. 請問 Java 是由誰發明的？

Java 是由 *詹姆斯·高斯林* (James Gosling) 和他的團隊在 1990 年代初期於 *Sun Microsystems* 公司所發明的。詹姆斯·高斯林當時的目標是開發一種適用於家電設備的程式語言，能在不同的硬體設備上執行。這個專案一開始稱為 "Green Project"，最早的語言原型被稱為 *Oak*。由於商標問題，後來才更名為 *Java*。

Java 的「一次撰寫，到處執行」(Write Once, Run Anywhere, WORA) 特性，讓它迅速成為網頁應用和企業級應用程式的熱門語言。

3. 請問 Java 和 JavaScript 的區別。

Java 和 JavaScript 雖然名稱相似，但它們是兩種完全不同的程式語言，用途、語法和執行方式各不相同。以下是它們的主要區別：

➤ 語言類型

- Java 是一種靜態類型語言，必須在編譯時指定變量的類型，像是 *int*、*String* 等。

- JavaScript 是一種動態語言，變量的類型是在運行時確定的，可以改變變量的類型。

➤ 用途

- Java 通常用於後端開發、桌面應用程序和行動裝置開發（例如 Android 應用）。它也經常用於大型系統和企業級應用程式。

- JavaScript 主要用於前端開發，用於製作動態的網頁介面。近年來，隨著 Node.js 的出現，JavaScript 也被用於後端開發。

➤ 執行環境

- Java 需要 Java 虛擬機（JVM）來執行，Java 程式碼在編譯後生成平台無關的 bytecode，能夠在支持 JVM 的各種操作系統上運行。

- JavaScript 主要在瀏覽器中運行，由瀏覽器的 JavaScript 引擎（如 Chrome 的 V8 引擎）執行；Node.js 則是將 JavaScript 帶到了伺服器端。

➤ 語法差異

- Java 是物件導向的，並且強調類與繼承的概念；程式碼必須包裝在類中，並且以 *main* 方法作為程序入口。

- JavaScript 是基於物件的，但具有更靈活的原型繼承系統，允許更簡單的函式程式設計風格。

➤ 編譯與執行

- Java 是編譯型語言，需將程式碼編譯成 bytecode，然後由 JVM 執行。

- JavaScript 是解釋型語言，瀏覽器或 JavaScript 引擎會即時解釋並執行程式碼。

總結

雖然 Java 和 JavaScript 在名稱上相似，但它們的語法結構、用途和執行方式上差異很大。Java 偏向於強類型、企業應用的後端開發，而 JavaScript 則更靈活，主要用於網頁的前端開發。

- GitHub 程式連結：<https://github.com/WoemTheCat/CH2>
- GitHub 的截圖：

