File Note

Buổi 3: Cách xây dựng đối tượng, thuộc tính, phương thức bằng ngôn ngữ Java

# 1, Cách khai báo đối tượng trong Java

**Cú pháp**

|  |
| --- |
| **public class** HelloWorld {  // Khai báo thuộc tính  // Khai báo và khởi tạo phương thức  } |

* Một file chỉ lưu một đối tượng
* Quy tắc đặt tên :

+ Viết hoa chữ cái đầu tiên của mỗi từ

+ Chỉ được sử dụng các chữ cái a-z, A-Z, 0-9 để đặt tên đối tượng

+ Không đặt số ở đầu

* Ví dụ: HocSinhGioi, StringCount, …

# 2, Cách khai báo thuộc tính java

**Cú pháp**

|  |
| --- |
| **int** thuocTinh; |
| <Kiểu dữ liệu> tenThuocTinh [= giá trị]; |

* Quy tắc đặt tên

+ Viết hoa chữ cái đầu tiên của mỗi từ, ngoại trừ chữ cái đầu tiên

+ Chỉ được sử dụng các chữ cái a-z, A-Z, 0-9 để đặt tên đối tượng

+ Không đặt số ở đầu

* Ví Dụ: hocSinh, sinhVien, giaoVien …

Thuộc tính hằng

Cú pháp

|  |
| --- |
| **final** <KDL> **tenThuocTinh** = <giá trị>; |
| **final float** PI\_VALUE = 3.14f; |

* Quy tắc đặt tên

+ Sử dụng các chữ cái A-Z, 0-9 để đặt tên thuộc tính

+ Các từ cách nhau bằng dấu gạch dưới '\_'

* Ví dụ:

**final double** PI = 3,141592653589793f;

**final double** FIRST\_CHARACTER = 'a';

# 3, Cách khai báo phương thức trong Java

**Cú pháp**

|  |
| --- |
| <Kiểu dữ liệu> tenPhuongThuc ([<KDL> tenThamSo]) {  // Nội dung phương thức  } |

* Quy tắc đặt tên

+ Viết hoa chữ cái đầu tiên của mỗi từ, ngoại trừ chữ cái đầu tiên

+ Chỉ được sử dụng các chữ cái a-z, A-Z, 0-9 để đặt tên phương thức

+ Không đặt số ở đầu

* Ví dụ: tinhTong, timKiem, …

**Phương thức không trả về**

|  |
| --- |
| **void** tenPhuongThuc ([<KDL> tenThamSo]) {  // Nội dung phương thức  } |

(Trong bài thi chỉ sử dụng phương thức **void,** trừ các phương thức **tìm kiếm, liệt kê**)

**Phương thức trả về**

|  |
| --- |
| **<Kiểu dữ liệu>** tenPhuongThuc ([<KDL> tenThamSo]) {  // Nội dung phương thức  **return** ketQua;  } |

**Tham số truyền vào**

* Là các dữ liệu được thực thi các công việc trong phương thức
* Một phương thức sử dụng tham số truyền vào khi bản thân nó không có đủ dữ liệu để thực hiện công việc

Từ khóa **return**

* Là một từ khóa (keyword) được java định nghĩa dùng để kế thúc một phương thức
* **return** có thể xuất hiện bất kỳ chỗ nào trong phương thức, khi gặp lệnh này sẽ ngay lập tức dừng lại tại vị trí đó
* Lệnh **return** là bắt buộc phải có trong phương thức trả về kết quả

BTVN

* Tương tự như 2 buổi trước
* Dùng Eclip, mỗi bài 1 project