



Phần 3

Lập Trình Hướng Đối Tượng với Java

Chương 1:

Lớp - Đối Tượng - Phương Thức Xây Dựng

TS. Phan Thượng Cang
Khoa CNTT&TT-Đại học Cần Thơ

Lớp (Class)

⌘ Khai báo:

```
<phạm vi> class <Tên_Lớp>{
```

```
    // các thuộc tính
```

```
    <phạm vi> <kiểu> <tên_thuộc_tính>;
```

```
    // các phương thức
```

```
    <phạm vi> <kiểu> <tên_phương_thức>(đối_số){
```

```
        // định nghĩa phương thức
```

```
    }
```

```
}
```

Lớp (Class)



⌘ Phạm vi truy cập:

-private: chỉ được truy cập trong cùng lớp

***default**: không giới hạn trong cùng lớp, gói

#protected: không giới hạn trong cùng
lớp, gói, và lớp con

+public: không giới hạn

Lớp (Class)

⌘ Phạm vi truy cập:

package goiX

```
class A {  
    private  
    protected  
    public  
}
```

a

b

c

d

```
class B {  
    câu lệnh  
}
```

package goiY

```
class D  
extends A {  
    câu lệnh  
}
```

```
class C {  
    câu lệnh  
}
```

import goiY.D

import goiX.A

Lớp (Class)

⌘ Ví dụ:

```
public class Diem{
```

```
    private int x, y ;
```

```
    public void gan(int x1, int y1){
```

```
        x = x1;
```

```
        y = y1;
```

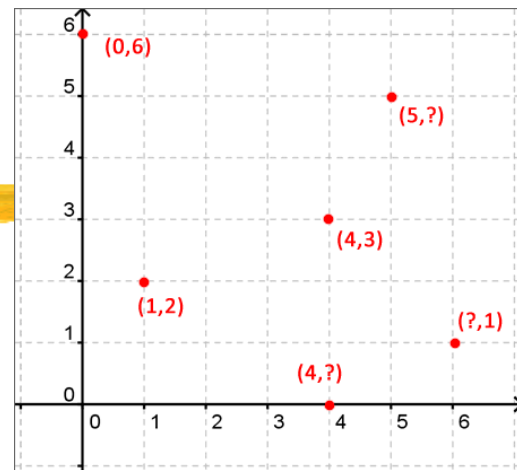
```
    }
```

```
    public void in(){
```

```
        System.out.println( "(" + x + "," + y + ")" );
```

```
    }
```

```
}
```



Đối tượng (Object)

⌘ Khai báo biến tham chiếu:

<tên_lớp> <tên_đối_tượng>;

⌘ Cấp phát bộ nhớ cho đối tượng:

<tên_đối_tượng> = new <tên_lớp>();

⌘ Truy cập các thành phần chung của đối tượng:

<tên_đối_tượng> . <thành_phần>[(doiso)];

⌘ Ví dụ:

Diem a = new Diem(); a.gan(2, 7); a.in() ;
a.x = 5 ; ????

Mảng Đối tượng (Array)

⌘ Khai báo biến tham chiếu mảng:

<tên_lớp> <tên_mảng>[];

⌘ Cấp phát số phần tử mảng (các biến tham chiếu):

<tên_mảng> = new <tên_lớp>[n];

⌘ Cấp phát bộ nhớ cho các đối tượng của mảng:

for (int i=0;i<n;i++)

<tên_mảng>[i] = new <tên_lớp>();

Mảng Đối tượng (Array)

⌘ Ví dụ:

```
Diem a[];
```

```
a = new Diem[100];
```

```
for (int i=0 ; i<a.length ; i++)  
    a[i] = new Diem();
```

```
for (int i=0 ; i<a.length ; i++)  
    a[i].nhap()
```


Phương thức xây dựng (Constructor)

⌘ Khai báo:

```
public class A{
```

```
    public A(đối số){
```

```
        // Cấp phát bộ nhớ cho các thuộc tính đối tượng
```

```
        // Khởi tạo giá trị ban đầu cho các thuộc tính
```

```
    }
```

```
}
```

⌘ Lưu ý:

- ☑ Hàm sẽ được gọi tự động ngay trước khi tạo đối tượng
- ☑ Tên hàm trùng với tên lớp và không có kiểu trả về
- ☑ Có thể tái định nghĩa hàm xây dựng. Có bao nhiêu hàm xây dựng thì có bấy nhiêu cách tạo đối tượng.