

Hàm khởi tạo trong class

Đc chạy ngay khi 1 object mới đc tạo ra.

```
class A {
```

```
    function __construct () {
```

```
        echo 1;
```

```
    }
```

```
}
```

```
$nom = new A();
```

!

~ - .

function -- construct (\$name, \$age){

\$this->name = \$name;

\$this->age = \$age

}

\$a = new A('thienth', 30);

=> \$a → name = 'thienth'.

→ age = 30

Hãng số Hong class

```
$a = 5;      const NAME = "thienth";  
$a = 6;      define("NAME", "thienth");
```

```
echo NAME;
```

Hãng & thẻ gán lại qtv:

class — mỗi class đc phép tạo ra h/số & phạm vi
tác dụng thuộc về class.

```
class A {  
    const NAME = "d hien th"; // h/xc  
                                // tính (static)
```

```
    echo A::NAME;
```

```
class B {
```

```
    const NAME = "nam ht";
```

```
    echo B::NAME;
```

```
}
```

B :: X() → Y()

→ Z()

→ var();

gọi hàm vô hạn

⇒ Static

Static X()

return \$model;

function Y()

return \$this

function Z()

return \$this

Base Model



where (\$1, \$2, \$3)

\$x = select * from

User

Product.

Category.

users

\$table products

categories.

\$model → table

where \$1, \$2, \$3

function get()

connect / execute
prepare / fetchAll

Product::where('price', >, 3000)

⇒ select * from products

where price > 3000.

\$prods = Product::where() → get()

+) Trừu tượng: cụ thể của mô tả về đối tượng
trong q/c của p/năm \rightarrow Class (tạo class)

+) Kế thừa: lớp B kế thừa (extends) lớp A.
thì các Obj của lớp B đ/c phép dùng $\left\{ \begin{array}{l} \text{thuộc tính} \\ \text{phương thức} \end{array} \right.$
đ/c tạo ra từ lớp A.

+) Đa hình (over write): các lớp con đ/c phép
đ/n lại q/t; /nội dung của các tính /phương thức đ/c đ/n
ở lớp cha.

```
class A {
```

```
function run() {
```

```
return "im running";
```

```
}
```

```
}
```

```
class B extends A {
```

```
function run() {
```

```
return $this->name . "running";
```

```
}
```

```
}
```

```
$x = new B();  
(x->name = "phat";  
$x->run());  
=> phat running
```


