

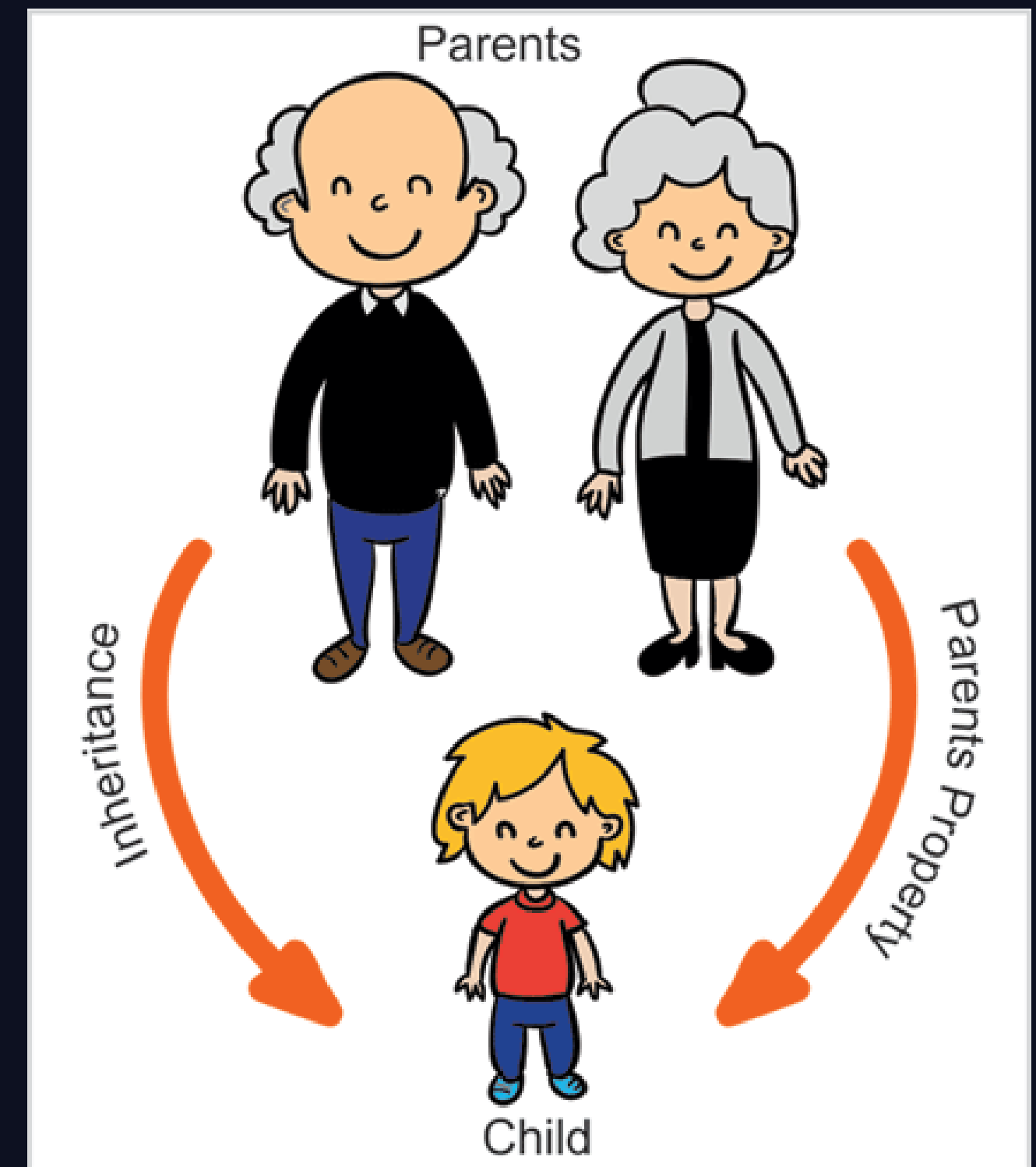
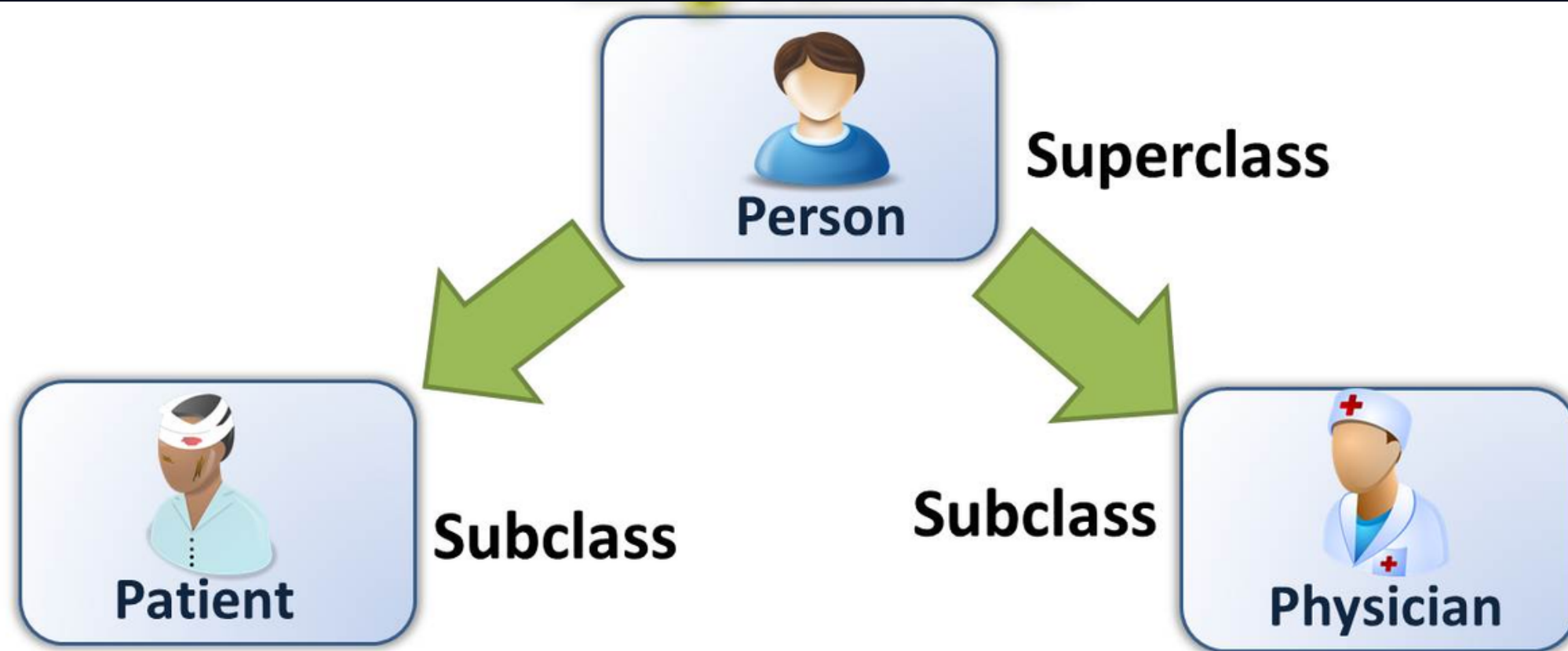
# OBJECT ORIENTED PROGRAMMING

Inheritance & Composition

# What is an Inheritance ?

- When a class derives from another class.
- Relation Between Two Classes.
- The child class will inherit all the public and protected properties and methods from the parent class
- **It's a Relation Called ' IS-A ' Relation**
- **Child Class can have its own properties and methods**
- Using the extends keyword to Make Inheritance Relation
- Parent : Who Give His Properties and Method To Child. Called Superclass
- Child : Who Take the Properties and Methods From the Parent. Called Subclass

# Example...



# Coding...

```
class Person {  
  
    public $name;  
    private $address;  
    protected $age;  
  
    public function sleep(){  
        echo "I'm Sleeping";  
    }  
  
    protected function walk(){  
        echo "i'm Walking";  
    }  
  
    private function eat(){  
        echo "I'm Eating";  
    }  
}
```

To Can Access The class in another File

```
require "Person.php";  
class Teacher extends Person  
{  
    // Hi I'm Empty Class Teacher  
    // Derived From My Parent Class Person  
}  
$teacher1 = new Teacher();  
$teacher1->name = "Ahmed";  
echo $teacher1->name; // Ahmed
```

# Coding...

Without Declare Any Properties or Method !!! WoW

```
class Teacher extends Person
{
    // Hi I'm Empty Class Teacher Derived From My Parent Class Person
    function setAge()
    {
        $this->|
    }
}
```

- f name
- m setAge() void
- f age
- m sleep() void
- m walk() void

Press Ctrl+. to choose the selected (or first) suggestion and insert a dot afterwards [Next Tip](#)

BUT hamm

Where Is address and eat()?

PRIVATE



# What About Constructor



```
class Person {  
  
    public $name;  
    private $address;  
    protected $age;  
  
    function __construct()  
    {  
        echo "Welcome"  
    }  
}
```



```
class Teacher extends Person  
{  
  
    // I Take The Constructor From My Parent  
}  
$teacher1 = new Teacher(); // Welcome
```



```
class Teacher extends Person  
{  
    // I Override My Parent Constructor  
    function __construct()  
    {  
        echo "Teacehr";  
    }  
}  
$teacher1 = new Teacher(); // Teacehr
```

# What is Overriding?

- The method must have **the same name** as in the parent class
- The method must have **the same parameter** as in the parent class.
- There must be an **IS-A** relationship (inheritance).

```
class Person
{
    function walk()
    {
        echo "I'm Walking is Walking";
    }
}
```

```
class Teacher extends Person
{
    function walk()
    {
        echo "Sara is Walking";
    }
}
```

# Overriding Con...

The method must have **the same parameter** as in the parent class.

## How Can I Prevent Overriding

```
class Person
{
    function walk($name)
    {
        echo "{$name} Walking is Walking";
    }
}
```

```
class Teacher extends Person
{
    function walk()
    {
        echo "Sara is Walking";
    }
}
```

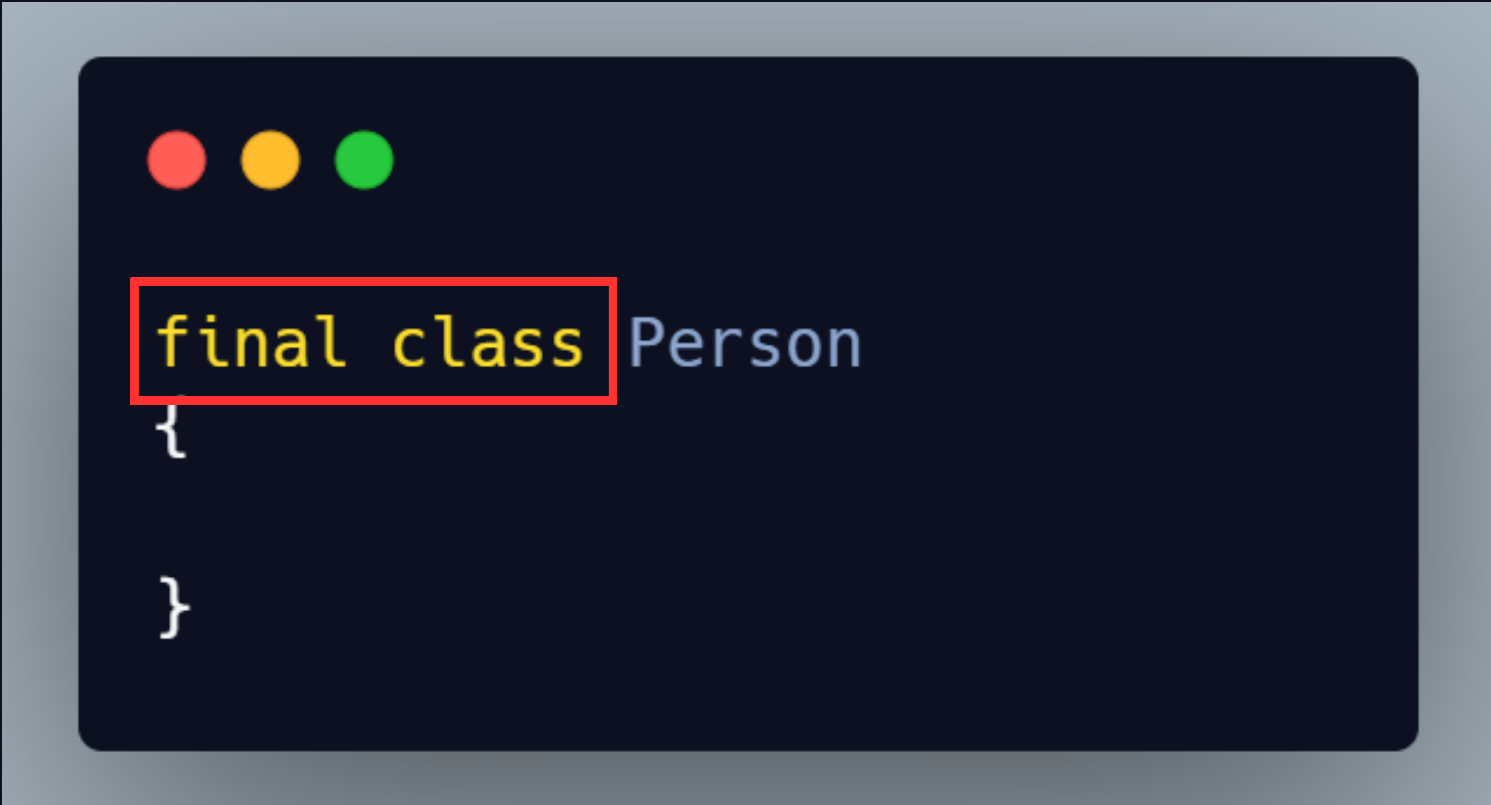
**Error**



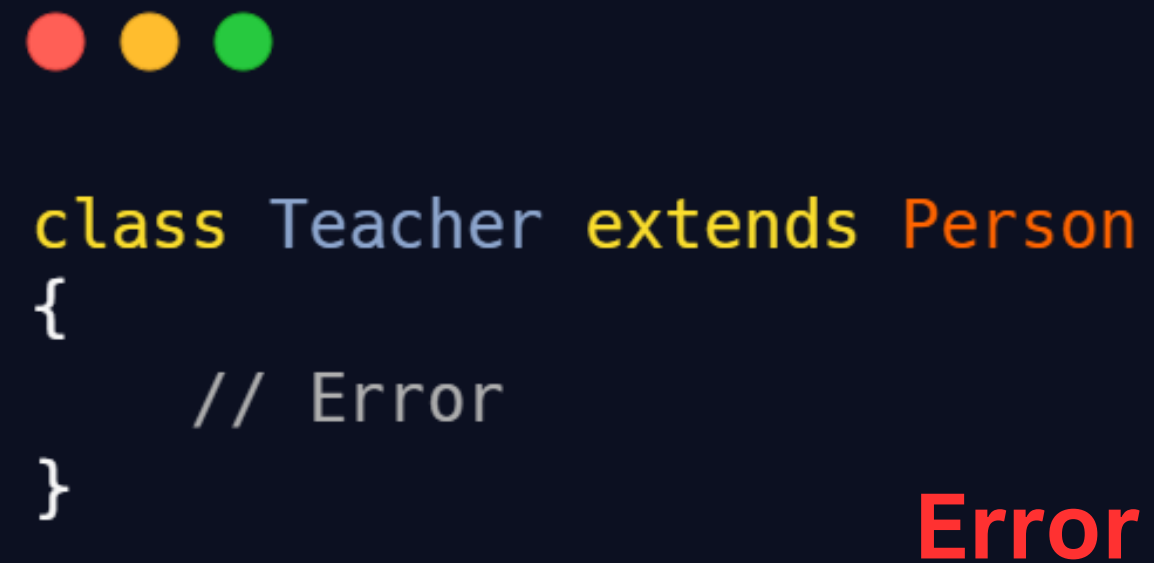
# Final Keyword

- By Using **Final Keyword** U can **Prevent Overriding and Inheritance**

## Prevent Inheritance



```
final class Person  
{  
  
}
```



```
class Teacher extends Person  
{  
    // Error  
}
```

**Error**

# Final Keyword

- By Using **Final Keyword** U can **Prevent Overriding and Inheritance**

## Prevent Overriding

```
class Person
{
    final function walk()
    {
        echo "Ahmed Walking is Walking";
    }
}
```

```
class Teacher extends Person
{
    function walk()
    {
        echo "Sara is Walking";
    }
}
```

**Error**

Composition

# Composition

- Composition is concept in object-oriented programming to implement **has-a** relationship between objects.
- Composition involves using other classes to build more complex classes and **there is no parent/child relationship exists in this case.**
- There are two types of composition : **association and aggregation.** Both types of composition include a **has-a** relationship

# Example

Consider the computer as an object. Computer **has-a** motherboard, keyboard, speaker, etc. Composed objects also can be composed of other objects. For example, **motherboard** **has-a** RAM, CPU, ect

Object



# Code

```
class Engine
{
    public function getType()
    {
        echo "Type IS German";
    }
}
```

As Soon As You Create Object  
From Car Make A composition  
With Engine

```
class Car
{
    // Car Has-An Engine
    public $engine;
    function __construct()
    {
        $this->engine = new Engine();
    }
}

$BMW = new Car();
$BMW->engine->getType(); // Type IS German
```

Object Of Type Engine



# Others Ways To Code

## Class Name

```
class Car
{
    public Engine $engine;
    function __construct($engine)
    {
        $this->engine = $engine;
    }
}
$BMW = new Car(new Engine);
$BMW->engine->getType(); // Type IS German
```

## Dependency Injection.

```
class Car
{
    public $engine;
    function __construct(Engine $engine)
    {
        $this->engine = $engine;
    }
}
$BMW = new Car(new Engine);
$BMW->engine->getType(); // Type IS German
```

```
class Car
{
    public $engine;
    function __construct($engine)
    {
        $this->engine = $engine;
    }
}
$Engine = new Engine(); // Engine Object
$BMW = new Car($Engine);
$BMW->engine->getType(); // Type IS German
```

# More Complex Example

```
class Ram
{
    function getRamSize()
    {
        echo "Ram Size 32G";
    }
}
```

```
class Motherboard
{
    public Ram $ram;
    function __construct()
    {
        $this->ram = new Ram();
    }
}
```

```
class PC
{
    public Motherboard $motherboard;
    function __construct()
    {
        $this->motherboard = new Motherboard();
    }
}

$myPc = new PC(); // Object Of Type PC

$myPc->motherboard->ram->getRamSize(); // Ram Size 32

// motherboard => Object Of Type Motherboard
// ram => Object Of Type Ram
```



# T a s k   S e a r c h

Favor Composition over Inheritance. Why ?

Dependency Injection.

Does PHP Has Multi Inheritance ?. How Can Do That ?

# Resources

- <https://medium.com/geekculture/oop-series-composition-6c67a19cabd1>
- <https://www.koderhq.com/tutorial/php/oop-composition/#how-to-associate-through-composition>
- <https://www.amitmerchant.com/reasons-use-composition-over-inheritance-php/>
- <https://www.digitalocean.com/community/tutorials/composition-vs-inheritance>

# *Quote*

The most disastrous thing  
that you can ever learn is  
your first programming  
language

Alan Kay



# Thank You

Abdelrahman Abdullah