

- 1) Destructor → In PHP, destructors are called
 - when it explicitly destroys an object
 - or when all the references to the object go out of scope
- 2) Destructors have special name

PHP -> `__destruct()`

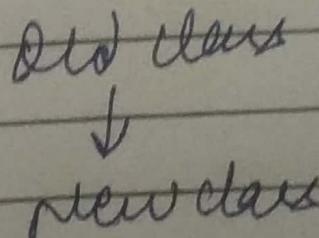
- ↳ Destructors don't have any arguments

3) Syntax: `function __destruct()`

```
echo "Destroyed";
```

Inheritance →

- ① The mechanism of defining a new class from old one is called inheritance or derivation



Old class → Super class, Base Class,
Parent Class

Child class → new class, Sub class,
Extended class

Parent Class



Child class

- Father
 - name
 - Money
 - business

Son

- Job
- access property
of father

Class A

1 public & id
public & name

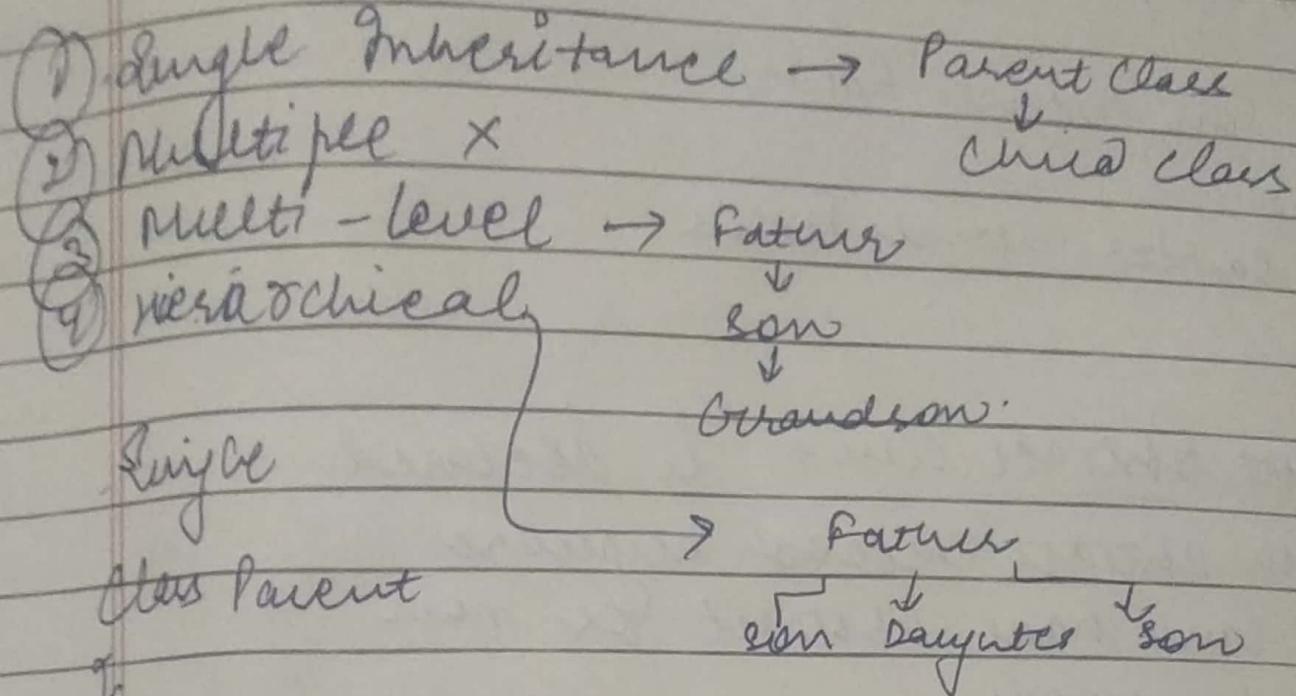
Declaration of child class

2
3
4;

Class child class extends parent
name class name

5
6;

Type of Inheritance



class Child extends Parent

multi-level Class Father
T G

Class Son extends Father
T G

Class Grandson extends Son
T G

Hierarchical

User Father
↳ ↳

Class Son extends Father
↳ ↳

Class Daughter extends Father
↳ ↳

Virtual Abstract class is declared

with abstract keyword.

↳ It can have abstract or non abstract methods.

↳ Object of abstract class can't be created

Abstract method →

Method that is declared as abstract & doesn't have implementation

abstract function sleep(); ^{if no body}

abstract class father → object of
2 abstract class
can't be created

4 abstract function disp(); created

Class Son extends Father

2 public function disp();

2 echo "Abstract defined";

4

<?php

Class Father {

function disp() {

echo "Method";

44

\$obj = new Father;

\$obj → disp();

?>

If we use abstract class
then shows error

∴ we can't create
the obj of abstract
class

<?php

abstract class Father {

function disp() {

echo "Method";

• abstract function ab();

4

class Son extends Father {

public function ab() {

echo "Abstract";

44

\$obj = new obj;

\$obj → ab();

?>

obj → Abstract

abstract method → mandatory
class to make class abstract

final →

final keyword is used to create final method or class

→ final method can't be overridden in child class

→ final class can't be inherited it means we can't create subclass of final class

Class A {

 final function disp() {
 echo "super class";
 }

Class B extends A {

 function disp() { → error → can't override
 echo "child class";
 } → final method.

final class →

final class A {

 function disp(); → Class B may not inherit
 from final class /

Class B extends A → error → ∵ we can't
create ∵ class is final