**TRAITS**:Traits are used to declare methods that can be used in multiple classes. Traits can have methods and abstract methods that can be used in multiple classes, and the methods can have any access modifier (public, private, or protected).

Traits are declared with the trait keyword

Eg:

<?php

trait hello{

public function sayHello(){

echo “Hello everyone”;

}

public function sayhi(){

echo “Hi everyone”; //multiple functions

}

}

trait bye{

Public function sayBye(){

echo “Bye Bye everyone”; //multiple traits

}

}

class base{

use hello,bye;

}

class base2{

use hello;

}

$test =new base();

$test->sayHello();

$test->sayBye();

$test->sayhi();

?>

Output : **Hello everyoneBye Bye everyoneHi everyone**

**Namespaces:**Namespaces are qualifiers that solve two different problems:

1. They allow for better organization by grouping classes that work together to perform a task

**Namespaces are declared at the beginning of a file using the namespace keyword**

**Eg:**

<?php

namespace Html;

class Flower {

public $title = "";

public $numPetals = 5;

public function message() {

echo "<p>Flower '{$this->title}' has {$this->numPetals} petals.</p>";

}

}

$petals = new Flower();

$table->title = "My petals";

$table->numPetals = 5;

?>

<html>

<body>

<?php

$petals->message();

?>

</body>

</html>

**Output: Flower '' has 5 petals.**