**Object Oriented Programming Structure**

**Class**

A PHP class is a group of values with a set of operations to manipulate these values. Classes facilitate modularity and information hiding. Classes are used to define a new data type.

**Rules**

* **The class name can be any valid label.**
* It can’t be PHP reserved word.
* A valid class name starts with a letter or underscores, followed by any number of letter, numbers or underscores.

**Syntax: -**

Class Classname

{

var $variable\_name; // Data Member / Properties

var $variable\_name; // Data Member / Properties

function Method\_name() // Methods/ Member Function

{

Body of Method;

}

function Method\_name(parameter\_list)

{

Body of Method;

}

}

**Example: -**

**class** Student

{

var $subject; //global variable

function subjectName($math)

{

global $subject;

$subject = $math;

echo “Subject is $subject”;

}

}

**Example 2: -**

**class** Mobile

{

var $model; //global variable

function showModel($number)

{

global $model;

$model = $number;

echo “Model Number: $model”;

}

}

**Note: - after php version 4.X, we don’t write var before variable in class . Now in new method or after php 4 we use public, private,protected … etc instead of var.**

**New Example:**

**class** Mobile

{

public $model; //global variable

function showModel($number)

{

$this->model = $number;

echo “Model Number: $this->model”;

}

}

**Key Point: -**

* **You can’t assign computed value inside a class.**

**Example: -**

public $price = 10 + 20;

public $name = “Rahul”. “Jaiswal”;

* **You can’t begin the name of method with a double underscore \_\_**

**Example: -** function \_\_setName()

**Object**

Object is class type variable. Each time you create an object of a class a copy of each variable defined in the class is created. In other words yoy can say that each object of a class has its own copy of data member defined in the class. Member functions have only one copy and shared by all the objects of that class. All the objects may have their own value of variables.

**Syntax: - $object\_name = new class\_name;**

**Creating Object**

**class** Mobile

**{**

public $model;

function showModel($number)

{

$this->model = $number;

echo “Model Number: $this->model;

}

**}**

$Samsung = **new**  Mobile; // object created

**Accessing class member using object**

**->** operator is used to access class member using object.

Object\_name->variable\_name;

$Samsung->model;

Object\_name->method\_name();

$Samsung->showModel();

Object\_name->method\_name(parameter\_list)

$Samsung->showModel(‘A8’);

**$this Keyword**

The $this Keyword points to the current object. You use $this followed by the -> operator. In Addition, you omit the $ in front of the property.

Ex: - $this -> model;

**class** Mobile

**{**

public $model;

function showModel($number)

{

$this->model = $number;

echo “Model Number: $this->model;

}

**}**

$Samsung = **new**  Mobile;

$Samsung->showModel(‘J7’);