Laporan Praktikum Pemrograman II

Modul II –OOP Constructor, Object Type, Inheritance, dan Overriding

Nellyana safitri/22103001011

Dosen : Achmad Arif Munaji, ST., M.Kom

Tanggal praktikum: senin, 27 Maret 2023

Nellyanasafitri2@gmail.com

Teknik Komputer

Institut Teknologi dan Sains Nahdlatul Ulama Kalimantan

***Abstrak***—Dalam dunia komputer, PHP pada dasarnya berarti PHP "hypertext preprocessor". PHP digunakan sebagai salah satu script untuk mempercantik tampilan website. Dalam prakteknya, PHP sering digunakan dengan bahasa pemrograman lain, seperti bahasa pemrograman HTML dan bahasa pemrograman JAVAscript. PHP adalah bahasa skrip sisi server yang dirancang untuk pengembangan web. Selain itu, PHP juga dapat digunakan sebagai bahasa pemrograman umum.

***Kata kunci : Bahasa pemrograman PHP, OOP***

1. Pendahuluan

Bahasa pemrograman berorientasi objek atau object-oriented programming (oop) adalah jenis pemrograman yang menggunakan objek dan kelas. Php awalnya hanya kumpulan skrip sederhana. Berbagai fitur pemrograman berorientasi objek ditambahkan selama pengembangan.

Pada praktikum ini mahasiswa membuat program atau coding pada visual studio code dengan Bahasa pemrograman php menggunakan object oriented programing (OOP) Constructor, Object Type, Inheritance dan Overiding. Hasil codingan tersebut dapat di lihat menggunakan local host XAMPP.

1. Tinjauan Pustaka
   1. Constructor

Konstruktor adalah metode atau fungsi yang dipanggil secara otomatis saat kelas dibuat instance-nya (objek dibuat). Di bagian konstruktor, kita bisa melakukan apa saja di dalam metode/fungsi kecuali mengembalikan nilai. Konstruktor memungkinkan properti objek diinisialisasi setelah objek dibuat. Jika Anda membuat fungsi \_\_construct(), PHP akan secara otomatis memanggil fungsi tersebut saat Anda membuat objek dari kelas..

* 1. Object Type

Tipe objek adalah tipe data yang tidak hanya menyimpan data, tetapi juga berisi informasi tentang manipulasi data. Artinya, dalam tipe data lain, seperti: Integer, float, string hanya berisi tipe datanya sendiri, sedangkan tipe data objek dapat berisi tipe data lainnya .

* 1. Inheritance

Warisan adalah kelas yang diturunkan dari kelas lain. Warisan dibuat dengan membuat kelas baru yang tujuannya adalah untuk menghubungkan ke kelas lain yang diinginkan untuk menerima informasi atau data dari kelas yang ditautkan. Kelas baru ini disebut kelas anak.

* 1. Overring

Overriding adalah teknik PHP dengan konsep OOP yang mengimplementasikan subclassing metode yang sama dengan metode induk. Mengapa Anda harus menggunakan Override? Terkadang perlu untuk membuat fungsi yang sama tetapi kondisinya berbeda. Prasyarat untuk menggunakan teknik override adalah Anda harus memiliki kelas induk dan kelas anak .

1. METODE PRAKTIKUM

Praktikum ini di mulai membuka aplikasi XAMPP control panel (local host) klik tombol start apache kemudian masuk pada file XAMPP htdock dan Buat folder bernama **praktikum1** lalu open with (Visual Studio Code) > lalu menambahkan file dengan nama Latihan1.php dan mulai membuat codingan sesuai dengan modul yang di bagikan sebagai berikut :

1. Terapkan constructor untuk class fruit pada modul.
2. Buat class baru bernama DetailFruit yang menerapkan konsep *Object Type class Fruit* pada *method* detail() dan kemudian *method* tersebut akan menampilkan output “I love $name and its $color.
3. Buat *Class Apple* dan *Class Banana* yang merupakan *Child* dari *Class Fruit*. Masing-masing *Child Class* mempunyai *property* $shape dan *method* detail() untuk menampilkan output: “I Love $name, its $color, and its $shape”. Terapkan juga konsep *Overriding* untuk membuat baris program menjadi lebih rapi dan tidak diulang-ulang

Buat *Class* Kalkulator yang memiliki tiga *Child Class*, yaitu Tambah, Kurang, dan Bagi. *Parent Class* mempunyai *property* $angka1 dan $angka2. Semua *Class* mempunyai *method* hasil(). *Method* hasil() pada *Parent Class* hanya menampilkan angka saja tanpa dilakukan proses aritmatika, sedangkan *Child Class* melakukan proses aritmatika sesuai dengan nama *Class*-nya.

1. hasil dan analisis

Dalam praktikum ini akan membuat code dengan VSC menggunakan bahasa pemrograman PHP OOP, yaitu construktor, type object, inhiretance, dan Overidding. Praktikum dimulai setelah memahami modul dan setelah free test.

<?php

class Fruit {

    public $name;

    public $color;

    function \_\_construct($name, $color)

    {

        $this->name = $name;

        $this->color = $color;

    }

    function get\_name()

    {

        return $this->name;

    }

    function get\_color()

    {

        return $this->color;

    }

}

class DetailFruit {

    function detail(Fruit $fruit)

    {

        return "I love {$fruit->name} and ist {$fruit->color}";

    }

}

class Apple extends Fruit {

    public $shape;

    function \_\_construct($name, $color, $shape)

    {

        parent:: \_\_construct($name, $color);

        $this->shape = $shape;

    }

    function detail()

    {

        return "I Love $this->name and its $this->color and ist $this->shape";

    }

}

class Banana extends Fruit {

    public $shape;

    function \_\_construct($name, $color, $shape)

    {

        parent:: \_\_construct($name, $color);

        $this->shape = $shape;

    }

    function detail()

    {

        return "I Love $this->name and its $this->color and its $this->shape";

    }

}

$apple = new Apple("Apple", "Green", "Oval");

$banana = new Banana("Banana", "Yellow", "Long");

$detail = new Fruit("Apple & Banana", "Green & Yellow", "Oval & Long");

echo $apple->detail();

echo "<br>";

echo $banana->detail();

echo "<br>";

echo $detail->get\_name();

echo "<br>";

echo $detail->get\_color();

Codingan di atas merupakan constructor Kemudian object type pada detailFruit( ), dan child class di luar class fruit yaitu class Apple dan Banana mempunyai property $shapes dan method detail. Codingan pada vsc di atas menghasilkan “I Love Apple and its Green and its Oval” untuk child class Apple, dan “I Love Banana and its Yellow and its Long” untuk child class Banana.

Praktikum di lanjutkan dengan soal no 4 yaitu.

* + - 1. Buat *Class* Kalkulator yang memiliki tiga *Child Class*, yaitu Tambah, Kurang, dan Bagi. *Parent Class* mempunyai *property* $angka1 dan $angka2. Semua *Class* mempunyai *method* hasil(). *Method* hasil() pada *Parent Class* hanya menampilkan angka saja tanpa dilakukan proses aritmatika, sedangkan *Child Class* melakukan proses aritmatika sesuai dengan nama *Class*-nya.

Berikut hasil perintah no 4 pada modul :

<?php

class Kalkulator {

    public $angka1;

    public $angka2;

    function \_\_construct($angka1 = 0, $angka2 = 0)

    {

        $this->angka1 = $angka1;

        $this->angka2 = $angka2;

    }

    function getData()

    {

        return "$this->angka1, $this->angka2";

    }

}

class Tambah extends Kalkulator {

    function hasilTambah()

    {

        return $this->angka1 + $this->angka2;

    }

}

class Kurang extends Kalkulator {

    function hasilKurang()

    {

        return $this->angka1 - $this->angka2;

    }

}

class Bagi extends Kalkulator {

    function hasilBagi()

    {

        return $this->angka1 / $this->angka2;

    }

}

$kalkulator = new Kalkulator();

$tambah = new Tambah(1,1);

$kurang = new Kurang(10,5);

$bagi = new Bagi(10,2);

//echo $tambah->hasilTambah();

//echo $kurang->hasilKurang();

echo $bagi->hasilBagi();

Codingan di atas menunjukan Hasil dari no 4 dengan class Kalkulator dengan child class Tambah, Kurang dan Bagi pada setiap method nya terdapat hitungan matika sesuai dngan nama class yang telah di tentukan. Codingan di atas dapat melakukan proses perhitungsn sesuai dengan class nya yaitu , penjumlahan, pengurangan, pembagian.

1. Kesimpulan

Pada praktikum ini dapat di simpulkan :

Pemrograman Berorientasi Objek, atau lebih dikenal dengan Object Oriented Programming (OOP), adalah paradigma pemrograman yang memecahkan masalah pemrograman dengan menyediakan objek (terdiri dari beberapa atribut dan metode) yang terkait satu sama lain dan diorganisasikan ke dalam satu grup atau class. Nantinya, objek-objek tersebut berinteraksi satu sama lain untuk memecahkan masalah pemrograman.

VI. Daftar Pustaka

1. <https://ejournal.bsi.ac.id/ejurnal/index.php>
2. Aziz, M. Farid. 2005. Object Oriented Programing dengan PHP5, Jakarta: PT Elex Media Komputindo
3. <https://www.warungbelajar.com/tutorial-belajar-php-part-36-mengenal-fungsi-constructor>
4. <https://kodingin.com/overriding-pada-php-oop/>