**TUGAS PEMROGRAMAN BERORIENTASI OBJEK**



**Disusun oleh:**

**Nama: Haniza Kurnia Dwi Putri**

**NIM: 22091397070**

**Kelas: 2022B**

**UNIVERSITAS NEGERI SURABAYA**

**2023**

1. Implementasi inheritance bangun datar
2. Class induk bangun datar

* Dalam bentuk bahasa pemrograman c++

**Source code:**

#include <iostream>

// Kelas Induk (Base Class)

class BangunDatar {

public:

float luas;

float keliling;

void hitungLuas() {

std::cout << "Menghitung luas bangun datar." << std::endl;

}

void hitungKeliling() {

std::cout << "Menghitung keliling bangun datar." << std::endl;

}

};

// Kelas Anak (Derived Class)

class PersegiPanjang : public BangunDatar {

public:

float panjang;

float lebar;

void hitungLuas() {

luas = panjang \* lebar;

std::cout << "Luas persegi panjang: " << luas << std::endl;

}

void hitungKeliling() {

keliling = 2 \* (panjang + lebar);

std::cout << "Keliling persegi panjang: " << keliling << std::endl;

}

};

int main() {

PersegiPanjang pp;

pp.panjang = 5.0;

pp.lebar = 3.0;

pp.hitungLuas(); // Memanggil metode dalam PersegiPanjang

pp.hitungKeliling(); // Memanggil metode dalam PersegiPanjang

// Memanggil metode dari kelas induk

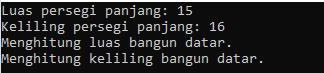
pp.BangunDatar::hitungLuas();

pp.BangunDatar::hitungKeliling();

return 0;

}

**Program’s run:**

****

* Dalam bentuk bahasa pemrograman php dan html (serta form)

**Source code:**

<!DOCTYPE html>

<html>

<head>

<title>Kalkulator Luas Bangun Datar</title>

</head>

<body>

<h1>Kalkulator Luas Bangun Datar</h1>

<form method="post">

<label>Pilih Bangun Datar:</label>

<select name="shape">

<option value="rectangle">Persegi Panjang</option>

<option value="circle">Lingkaran</option>

</select>

<br>

<label for="length">Panjang:</label>

<input type="text" name="length" id="length">

<br>

<label for="width">Lebar:</label>

<input type="text" name="width" id="width">

<br>

<label for="radius">Jari-Jari:</label>

<input type="text" name="radius" id="radius">

<br>

<input type="submit" name="calculate" value="Hitung">

</form>

<?php

// Class Shape sebagai class induk

class Shape {

protected $area;

public function calculateArea() {

return $this->area;

}

}

// Class Rectangle sebagai class anak

class Rectangle extends Shape {

private $length;

private $width;

public function \_\_construct($length, $width) {

$this->length = $length;

$this->width = $width;

$this->area = $this->length \* $this->width;

}

}

// Class Circle sebagai class anak

class Circle extends Shape {

private $radius;

public function \_\_construct($radius) {

$this->radius = $radius;

$this->area = 3.14 \* $this->radius \* $this->radius;

}

}

if (isset($\_POST['calculate'])) {

$shapeType = $\_POST['shape'];

if ($shapeType == "rectangle") {

$length = $\_POST['length'];

$width = $\_POST['width'];

$shape = new Rectangle($length, $width);

} elseif ($shapeType == "circle") {

$radius = $\_POST['radius'];

$shape = new Circle($radius);

}

echo "Luas Bangun Datar: " . $shape->calculateArea();

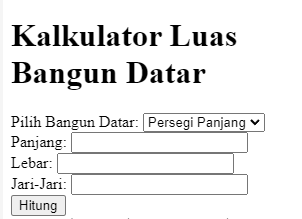
}

?>

</body>

</html>

**Program’s run:**

****

1. Class anak persegi, persegi Panjang, lingkaran, dan segitiga

* Dalam bahasa pemrograman c++

**Source code:**

#include <iostream>

#include <cmath>

class BangunDatar {

public:

virtual double hitungLuas() = 0;

};

class Persegi : public BangunDatar {

private:

double sisi;

public:

Persegi(double sisi) : sisi(sisi) {}

double hitungLuas() override {

return sisi \* sisi;

}

};

class PersegiPanjang : public BangunDatar {

private:

double panjang;

double lebar;

public:

PersegiPanjang(double panjang, double lebar) : panjang(panjang), lebar(lebar) {}

double hitungLuas() override {

return panjang \* lebar;

}

};

class Lingkaran : public BangunDatar {

private:

double jariJari;

public:

Lingkaran(double jariJari) : jariJari(jariJari) {}

double hitungLuas() override {

return M\_PI \* jariJari \* jariJari;

}

};

class Segitiga : public BangunDatar {

private:

double alas;

double tinggi;

public:

Segitiga(double alas, double tinggi) : alas(alas), tinggi(tinggi) {}

double hitungLuas() override {

return 0.5 \* alas \* tinggi;

}

};

int main() {

// Contoh penggunaan

Persegi persegi(5);

PersegiPanjang persegiPanjang(4, 6);

Lingkaran lingkaran(3);

Segitiga segitiga(4, 5);

std::cout << "Luas Persegi: " << persegi.hitungLuas() << std::endl;

std::cout << "Luas Persegi Panjang: " << persegiPanjang.hitungLuas() << std::endl;

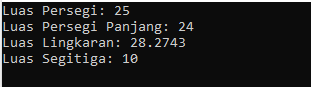
std::cout << "Luas Lingkaran: " << lingkaran.hitungLuas() << std::endl;

std::cout << "Luas Segitiga: " << segitiga.hitungLuas() << std::endl;

return 0;

}

**Program’s run:**

****

* Dalam bahasa pemrograman php dan html (disertai form)

**Source code:**

<!DOCTYPE html>

<html>

<head>

<title>Kalkulator Luas Bangun Datar</title>

</head>

<body>

<h1>Kalkulator Luas Bangun Datar</h1>

<form method="post">

<label for="bentuk">Pilih Bentuk:</label>

<select id="bentuk" name="bentuk">

<option value="Persegi">Persegi</option>

<option value="PersegiPanjang">Persegi Panjang</option>

<option value="Lingkaran">Lingkaran</option>

<option value="Segitiga">Segitiga</option>

</select>

<br><br>

<label for="param1">Parameter 1:</label>

<input type="text" id="param1" name="param1" placeholder="Panjang Sisi/Radius/Alas">

<br><br>

<label for="param2">Parameter 2 (hanya untuk Persegi Panjang dan Segitiga):</label>

<input type="text" id="param2" name="param2" placeholder="Lebar/Tinggi">

<br><br>

<input type="submit" value="Hitung Luas">

</form>

<?php

abstract class Bentuk {

abstract function hitungLuas();

}

class Persegi extends Bentuk {

private $sisi;

public function \_\_construct($sisi) {

$this->sisi = $sisi;

}

public function hitungLuas() {

return $this->sisi \* $this->sisi;

}

}

class PersegiPanjang extends Bentuk {

private $panjang;

private $lebar;

public function \_\_construct($panjang, $lebar) {

$this->panjang = $panjang;

$this->lebar = $lebar;

}

public function hitungLuas() {

return $this->panjang \* $this->lebar;

}

}

class Lingkaran extends Bentuk {

private $radius;

public function \_\_construct($radius) {

$this->radius = $radius;

}

public function hitungLuas() {

return M\_PI \* $this->radius \* $this->radius;

}

}

class Segitiga extends Bentuk {

private $alas;

private $tinggi;

public function \_\_construct($alas, $tinggi) {

$this->alas = $alas;

$this->tinggi = $tinggi;

}

public function hitungLuas() {

return 0.5 \* $this->alas \* $this->tinggi;

}

}

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$bentuk = $\_POST['bentuk'];

$param1 = floatval($\_POST['param1']);

$param2 = isset($\_POST['param2']) ? floatval($\_POST['param2']) : null;

switch ($bentuk) {

case 'Persegi':

$persegi = new Persegi($param1);

$luas = $persegi->hitungLuas();

echo "Luas Persegi: $luas";

break;

case 'PersegiPanjang':

$persegiPanjang = new PersegiPanjang($param1, $param2);

$luas = $persegiPanjang->hitungLuas();

echo "Luas Persegi Panjang: $luas";

break;

case 'Lingkaran':

$lingkaran = new Lingkaran($param1);

$luas = $lingkaran->hitungLuas();

echo "Luas Lingkaran: $luas";

break;

case 'Segitiga':

$segitiga = new Segitiga($param1, $param2);

$luas = $segitiga->hitungLuas();

echo "Luas Segitiga: $luas";

break;

default:

echo "Bentuk tidak valid.";

break;

}

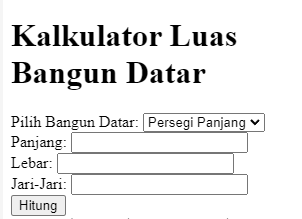
}

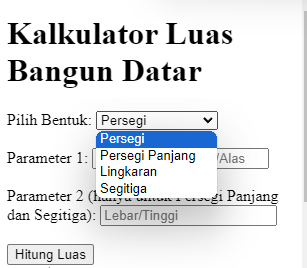
?>

</body>

</html>

**Program’s run:**

****

****