**Soal Pertama**

1. Dalam memulai menerapkan konsep OOP, hal yang paling utama yang harus diketahui adalah Class, Atribut, dan Method. Coba jelaskan ketiga hal tersebut berdasarkan pemahaman kalian !

**Jawaban**

* Class adalah cetakan untuk membuat sebuah object, Class akan mempresentasikan objek yang dibuat.
* Atribut adalah bagian dari Class yang digunakan untuk mendeklarasikan atribut yang kita buat dengan type data tertentu.
* Method itu kata lain dari fungsi ada dua method yaitu diawali dengan type data dan dengan Void dan nanti akan dipanggil di Class main**.**

**Soal Kedua**

1. Dalam membuat sebuah Class, ada ketentuan untuk tidak membuat nama Class yang diawali dengan angka. Apa yang akan terjadi jika kita tetap memaksa untuk membuat Class yang Namanya diawali dengan angka ?

**Jawaban**

Jika kita mengawali membuat Class dengan angka maka akan terjadi error

**Soal Ketiga**

1. Didalam satu file Java, setidaknya harus memiliki 1 class dengan nama yang sama seperti nama file .java nya. Cobalah untuk mengganti Nama Class menjadi berbeda dengan nama file javanya, apakah yang akan terjadi ? jika ada error, apa pesan error nya?

**Jawaban**

Yang terjadi akan error dengan pesan error “Class JavaApplicatio(file java yang kita buat) is public, should be declared in a file named JavaApplicatio.java”

**Tugas Keempat**

1. Sebenarnya dalam 1 file java, kita dapat menampung banyak Class didalamnya. Buatlah satu buah file java dengan nama BangunRuang.java, kemudian coba buat Class Kubus,Balok, dan Lingkaran tepat dibawah Class BangunRuang tersebut!

**Jawaban**

Run

Class kubus

Int sisi = 5

Return sisi \* sisi \* sisi

Output : Volume kubus = 125

Class Balok

Int Panjang =5 , lebar = 6, tinggi = 9;

Return Panjang \* lebar \* tinggi

Output : Volume Balok = 270

Class Lingkaran

Int Radius = 7

Return radius \* radius \* 3.14

Output : 153.86

**Source code**

|  |
| --- |
| 1. **package** javaapplication4; 2. **public** **class** BangunRuang { 3. **public** **static** **void** main(String[] args) { 4. kubus kubus1 = **new** kubus(); 5. **System.out.println("volume kubus : " + kubus1.sisi());** 6. Balok balok1 =**new** Balok(); 7. System.out.println("volume Balok : " + balok1.volume()); 8. Lingkaran lingkaran1 = **new** Lingkaran(); 9. System.out.println("luas lingkaran : " + lingkaran1.luas()); 10. **}** 11. } 12. **class** kubus{ 13. **int** sisi; 14. **int** sisi(){ 15. **sisi = 5;** 16. **return** sisi \* sisi \*sisi; 17. } 18. } 19. **class** Balok { 20. **int panjang =5 , lebar = 6 , tinggi = 9;** 21. **int** volume(){ 22. **return** panjang \* lebar \* tinggi; 23. } 24. } 25. **class Lingkaran{** 26. **int** radius = 7; 27. **double** luas(){ 28. **return** radius \*radius \* 3.14; 29. } 30. **}** |

**OUTPUT**

|  |
| --- |
|  |

**Soal Kelima**

1. STUDI KASUS PROJECT AKHIR : Berdasarkan studi kasus yang telah diberikan, cobalah buat/terapkan beberapa Class, Atribut, dan Method yang kalian buat!

**Jawaban**

* Class register, yaitu berfungsi untuk tempat menyimpan atribut dan method yang akan dibuat dan didalam nya ada constructor untuk di panggil sebagai objek di Class main atribut nya yaitu name, kursus, alamat, tanggal lahir dan umur dan ada method untuk menginput data ke dalam atribut.

**Source Code**

|  |
| --- |
| 1. **import** java.util.Scanner; 3. **class** register { 4. **private** String name, kursus , alamat , tanggal\_lahir; 5. **private int umur;** 6. Scanner inp = **new** Scanner(System.in); 8. **public** **void** Register(){ 9. **System.out.println("===================================");** 10. System.out.println("Registrasi Akun kursus"); 11. System.out.print("Masukkan nama : "); 12. name = inp.nextLine(); 13. **System.out.print("Masukkan umur : ");** 14. umur = inp.nextInt(); 15. System.out.print("Masukkan Alamat : "); 16. alamat = inp.next(); 17. System.out.print("Masukkan tanggal lahir : "); 18. tanggal\_lahir = inp.next(); 19. tanggal\_lahir = getTanggal\_lahir(); 20. } 21. **public class App {** 22. **public** **static** **void** main(String[] args) **throws** Exception { 24. **}** 25. } |