

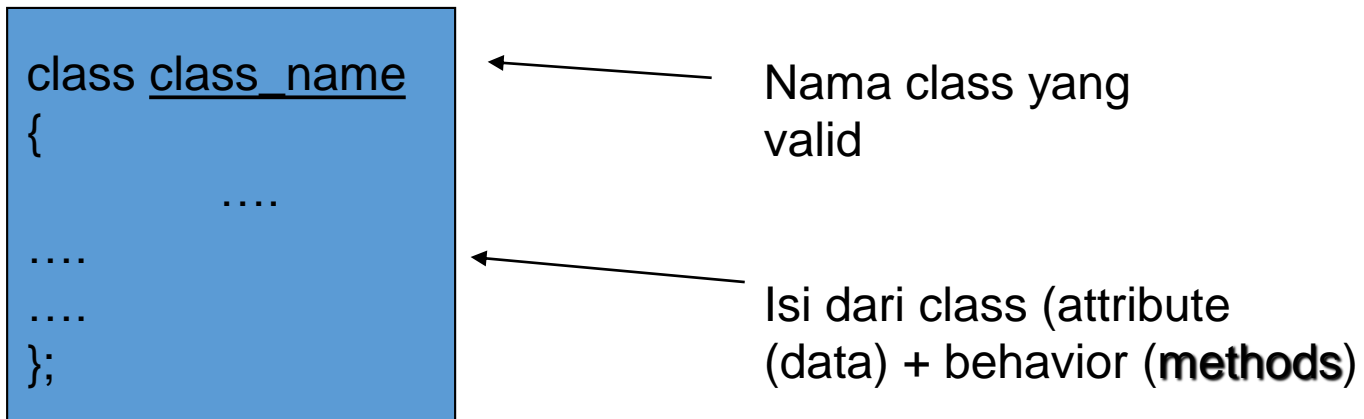


PEMROGRAMAN LANJUT

02 | OOP in C++

Class dalam Bahasa pemrograman C++

- Sebuah class didefinisikan dalam C++ dengan keyword *class*.
- Isi dari sebuah class harus berada diantara tanda kurung kurawal dan titik koma(;)
- Contoh pendefinisian sebuah class



Class dalam Bahasa pemrograman C++

- Di dalam isi dari class, ada keywords *private:* dan *public:* yang menspesifikasikan akses level dari anggota class yang terdiri dari attribute/data dan behavior/method
 - Default dari aksesnya adalah *private*.

```
class class_name
{
    private:
        ...
        ...
        ...
    public:
        ...
        ...
        ...
};
```

← private data atau methods

← Public data atau methods

Member access specifiers

- Spesifikasi level akses dari anggota class ada 2 macam yaitu public atau private
- public:
 - Bisa diakses secara langsung dari luar class.
- private:
 - Hanya bisa diakses dari method yang ada di class yang sama
 - Anggota class yang mempunyai level akses private hanya digunakan untuk kebutuhan di dalam class saja

Contoh class

- Contoh class dibawah ini mengenkapsulasi data dan method menjadi satu.

```
class Circle
{
    private:
        double radius;
    public:
        void setRadius(double r);
        double getArea();
};
```

Class yang lain tidak dapat mengakses variable radius karena private. Nilai radius hanya bisa diberikan di class

Method-method ini dapat diakses dari luar kelas karena public

Membuat object dari suatu class

- Mendeklarasikan sebuah variable yang tipe datanya class dapat dilakukan dengan cara membuat object dari class tersebut. Dari sebuah class dapat dibuat banyak object
 - *Instantiation*
- Sekali sebuah object dari class tertentu diinstansiasi maka memory location yang baru akan dibuat untuk menyimpan anggota class beserta nilainya
- Contoh object hasil instansiasi dari sebuah class
 - `Circle c; Circle *c;`

Class methods

- Class implementation: menuliskan kode isi dari method-method yang ada didalam class
- Ada dua cara:
 1. Method didefinisikan diluar
 - Menggunakan Binary scope resolution operator (::)
 - Nama method harus unik dalam satu class
 - Class yang berbeda boleh memiliki nama method yang sama
 - Format untuk mendefinisikan method dari sebuah class

```
ReturnType ClassName::MethodName( ) {  
    ...  
}
```

Class method

```
class Circle
{
    private:
        double radius;
    public:
        void setRadius(double r);
        double getArea();
};
void Circle::setRadius(double r)
{
    radius = r;
}
double Circle::getArea()
{
    return radius * radius * (22.0/7);
}
```

Didefinisikan diluar class

Class methods

2. Method didefinisikan didalam class
 - Tidak membutuhkan scope resolution operator dan nama class

```
class Circle
{
    private:
        double radius;
    public:
        void setRadius(double r){
            radius = r;
        }
        double getArea(){
            return radius*radius*(22.0/7);
        }
};
```

→ Didefinisikan
didalam
class

Mengakses anggota class

- Operators untuk mengakses anggota class
 - Dot member selection operator (.)
 - Object
 - Reference to object
 - Arrow member selection operator (->)
 - Pointers

```
void main() {  
    Circle c1;  
    c1.setRadius(7);  
    cout<<c1.getArea();  
}
```

```
void main() {  
    Circle *c1=new Circle();  
    c1->setRadius(7);  
    cout<<c1->getArea();  
}
```

Latihan

- Buatlah sebuah class SegiEmpat dan buatlah method untuk menghitung keliling, luas, dan panjang diagonal dari sebuah segiempat