10. Kelas

S. Indriani L, M.T

Struktur dan Kelas

- Berdasarkan definisinya struktur (`struct`) adalah suatu kelas-kelas biasa (`class`) dimana anggota-anggotanya secara default adalah bertipe public, sehingga kedua deklarasi dibawah ini adalah sama

- Pada contoh terdapat struktur bernama buku, yang dideklarasikan sbb:

```
struct buku
{
  char judul[35];
  char pengarang[25];
  int jumlah;
};
```

- Sebuah kelas (class) yang menyerupai struktur diatas dapat dibuat dengan menggantikan kata struct dengan class, seperti:

```
class buku
{
  char judul[35];
  char pengarang[25];
  int jumlah;
};
```

- Setelah kelas dideklarasikan (spt pada contoh sebelumnya), kelas tersebut dapat dipakai sebagai tipe data bagi penciptaan variabel, misalnya:

```
buku novel;
```

- Perlu diketahui, pada C++ variabel seperti `novel` (yang berkedudukan sebagai variabel kelas) biasa disebut Objek.
- Kebanyakan pemogram C++ menggunakan nama kelas (misalnya `buku` pada contoh) dengan awalan huruf kapital
- Pada contoh selanjutnya, semua kelas pada buku ini akan menggunakan awalan berupa huruf kapital.

```
#include<iostream.h>
#include<conio.h>
#include<string.h>

struct buku
{
  char judul[35];
  char pengarang[25];
  int jumlah;
};

void main()
{
  clrscr();

  buku novel;
  strcpy(novel.judul, "Pemograman C++");
  strcpy(novel.pengarang, "Abdul Kadir");
  novel.jumlah=12;
  cout<<novel.judul<<endl;
  cout<<novel.pengarang<<endl;
  cout<<novel.jumlah;
}
```

```
#include<iostream.h>
#include<conio.h>
#include<string.h>

class buku
{
  char judul[35];
  char pengarang[25];
  int jumlah;
};

void main()
{
  clrscr();

  Buku novel;    //pendefinisian variabel

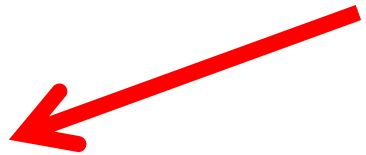
  strcpy(novel.judul, "Pemograman C++");
  strcpy(novel.pengarang, "Abdul Kadir");
  novel.jumlah=12;
  cout<<novel.judul<<endl;
  cout<<novel.pengarang<<endl;
  cout<<novel.jumlah;
}
```

- Namun jika program tersebut dikompilasi, akan muncul pesan error.
- Mengapa?
  - Kesalahan tersebut menunjukkan bahwa anggota kelas Buku tidak dikenal pada fungsi main().

- Kesalahan ini sekaligus menunjukkan bahwa terdapat perbedaan antara kelas dan struktur
- Meskipun pada C++, struktur juga dapat digunakan seperti halnya kelas, kelas lebih umum dipakai khususnya dalam pemograman berorientasi objek.

Penggunaan Public

- Agar program tersebut dapat dikompilasi, kata kunci `public` diikuti dengan tanda titik dua (:) perlu ditambahkan diatas anggota data bernama judul



```
class buku
{
  public:
  char judul[35];
  char pengarang[25];
  int jumlah;
};
```

- Baris yang berisi public diatas menyatakan bahwa yang ada dibawahnya bersifat publik (dapat diakses diluar kelas)

- Jika program dijalankan hasilnya persis sama dengan hasil program struktur
- Kata kunci `public` pada kelas Buku didepan biasa disebut sebagai penentu akses (*access specifier*).
- Selain `public`, juga terdapat penentu akses berupa:
  - ✓ `private`
  - ✓ `Protected` (akan dibahas pada bab pewarisan / inheritance)

Penggunaan Private

- Penentu akses private biasa digunakan pada kelas untuk memproteksi anggota-anggota tertentu pada kelas, agar tidak dapat diakses dari luar kelas secara langsung

```
#include<iostream.h>
#include<conio.h>
#include<string.h>

class Buku
{
   private:
   char judul[35];
   char pengarang[25];
   int jumlah;
   public:
   void inisialisasi (char *Judul, char *Pengarang, int Jumlah)
   {
        strcpy(judul, Judul);
        strcpy(pengarang, Pengarang);
        jumlah = Jumlah;
   }
   void info()
   {    cout<<"Judul          :"<<judul<<endl;
        cout<<"Pengarang      :"<<pengarang<<endl;
        cout<<"Jumlah         :"<<jumlah<<endl;
   }

};

void main()
{
   clrscr();

   Buku novel;
   novel.inisialisasi("Pemograman C++", "Abdul Kadir",12);
   novel.info();
}
```

- Jika program tersebut dikompilasi maka hasilnya akan sama dengan program sebelumnya.
- Dengan kata lain, pada kondisi seperti ini baris yang berisi private mempunyai efek yang sama dengan kalau baris tersebut dihilangkan.
- Mengingat *default* dari kelas adalah private