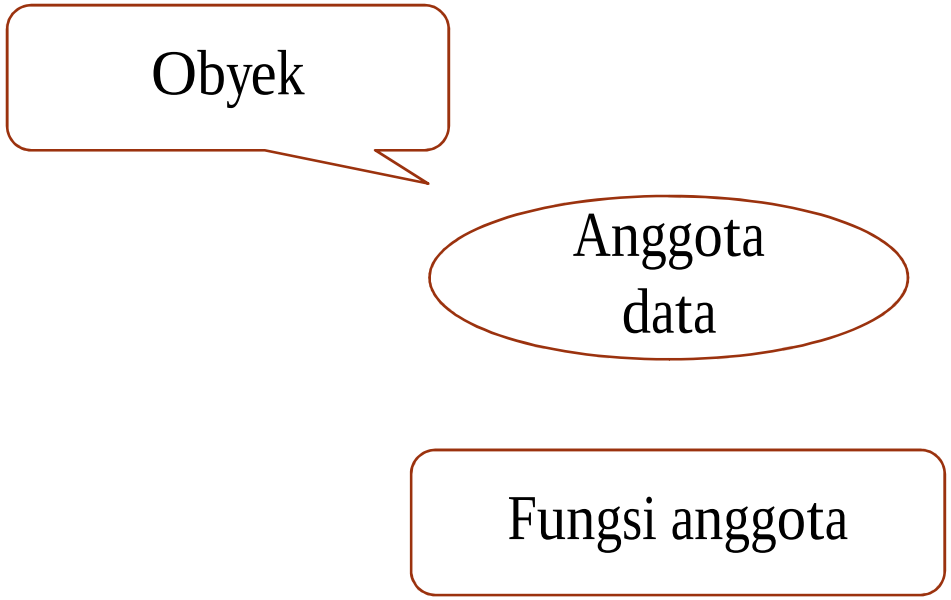
12.Objek

S. Indriani L, M.T

Menambahkan fungsi anggota

- Konsep obyek dari C++ sebenarnya digunakan untuk menyatukan data dan fungsi yang mengakses data dalam satu wadah



Data yang terletak di dalam kelas biasa disebut anggota data.
Fungsi yang terletak didalamnya disebut fungsi anggota

- Misalnya, bila terdapat data:
  - *judul buku*
  - *nama pengarang*
  - *jumlah buku*

  di awal perancangan harus dipikirkan, fungsi-fungsi dasar yang digunakan untuk mengakses tiga data tersebut.

  Fungsi-fungsi yang dimaksud dapat berupa:
  - ✓ mengisikan data
  - ✓ menampilkan data

- Selain itu perlu juga direncanakan, data atau fungsi mana saja yang boleh diakses diluar obyek dan yang hanya dipergunakan secara internal oleh obyek itu sendiri.

  → dalam hal ini penentu akses `public` atau `private` yang menentukannya

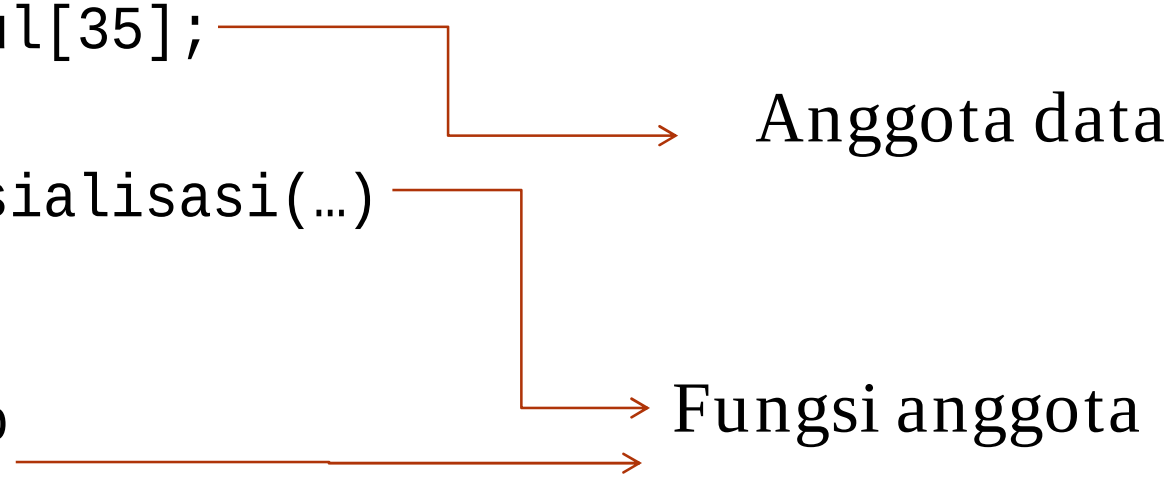
```
#include<iostream.h>
#include<conio.h>
#include<string.h>

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 Buku :"<<jumlah<<endl;
      }
};
void main()
{ clrscr();
  Buku novel;
  novel.inisialisasi("Meriam Benteng Navarone","Alistair", 12);
  novel.info();
}
```

```
class Buku
{
        private:
        char judul[35];
        …
        public:
        void inisialisasi(…)
        {
             ….
        }
        void info
        {…
        }

};
```



- Pada program, kelas Buku memiliki tiga buah anggota data (`judul`, `pengarang` dan `jumlah`) dan dua buah fungsi anggota (`inisialisasi ()` dan `info()`).
- Ketiga data anggota dinyatakan sebagai `private`. Oleh karena itu, ketiga data ini tidak dapat diakses diluar kelas. Namun data ini dapat diakses oleh kedua fungsi anggota dan semua fungsi anggota dapat diakses diluar kelas, mengingat dinyatakan sebagai `public`. Melalui kedua fungsi inilah data kelas dapat diakses, misalnya pada fungsi `main()`.

Mendefinisikan beberapa objek

- Beberapa objek dengan kelas sama dapat didefinisikan sebagaimana pendefinisian variabel biasa (bukan objek). Misalnya:

  ```
  Buku novel, fiksi;
  ```

- Merupakan pernyataan untuk mendefinisikan obyek bernama `novel` dan `fiksi`. Kelasnya sama yaitu `Buku`.
- Nama kelas mendahului nama-nama objek.
- Perlu diketahui, jika terdapat lebih dari sebuah objek dan obyek-obyek tersebut mempunyai kelas yang sama, data anggota pada masing-masing obyek bersifat terpisah.
- Lihat program berikut ini:

```
#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 Buku              :"<<jumlah<<endl;
        }
};

void main()
{
   clrscr();

   Buku novel, fiksi;

   novel.inisialisasi("Meriam Benteng Navarone","Alistair", 12);

   fiksi.inisialisasi("Harry Potter and the socerrer stone","J.K Rowling", 20);

   novel.info();

   fiksi.info();
}
```

Penugasan Nilai Antar obyek

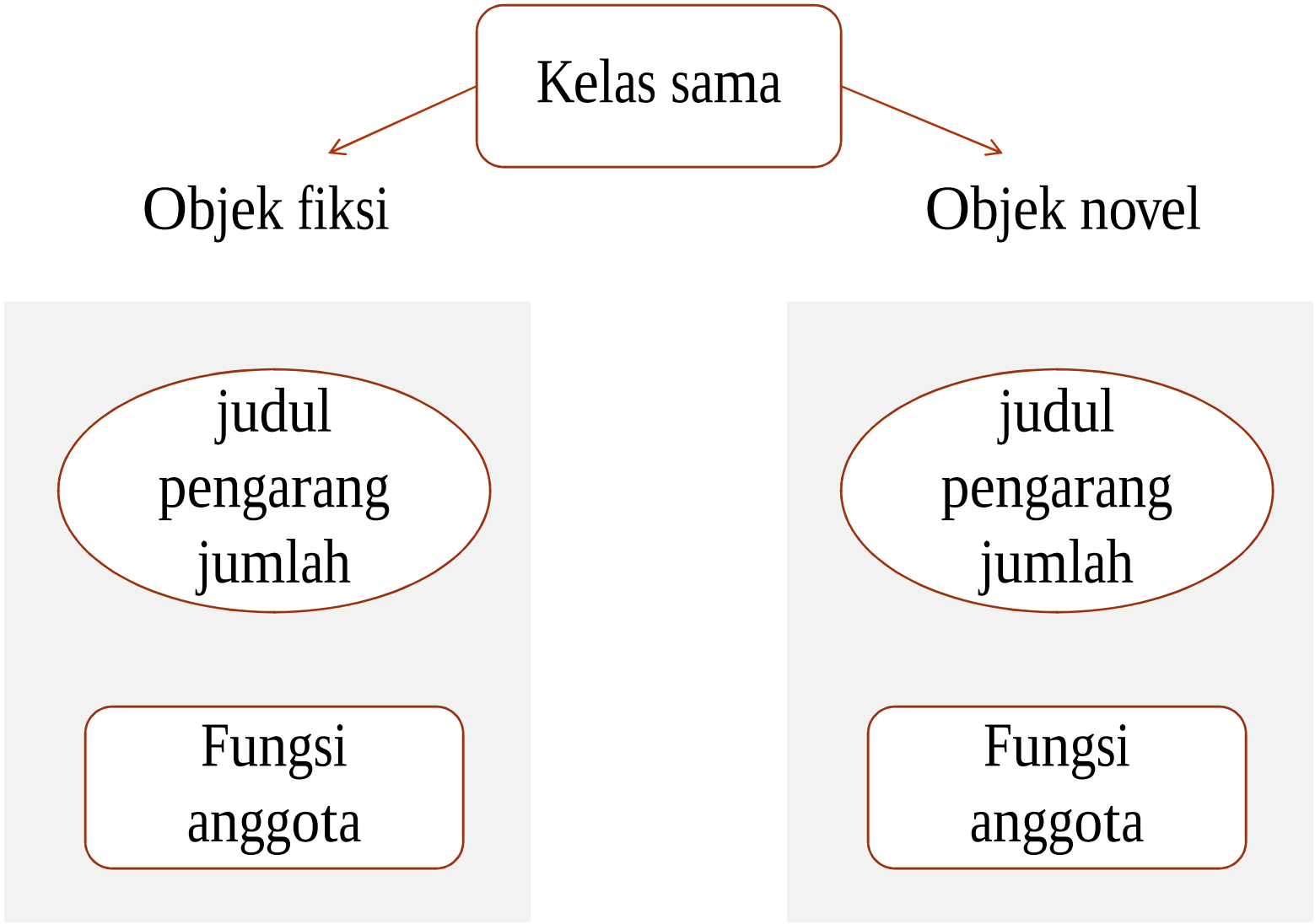
- C++ memungkinkan penyalinan nilai antar obyek dapat dilakukan dengan mudah, yaitu cukup menggunakan operator sama dengan (=).
- Sebagai contoh, dua buah obyek berkelas Buku diciptakan dengan melalui pernyataan:

```
Buku novel, fiksi;
```

Dengan menggunakan pernyataan:

```
fiksi = novel;
```

maka semua anggota data pada obyek fiksi akan diisi sesuai dengan anggota data pada obyek novel.


Kelas sama
Objek fiksi
Objek novel
judul
pengarang
jumlah
Fungsi
anggota
judul
pengarang
jumlah
Fungsi
anggota

Fiksi = novel;

```
#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 Buku              :"<<jumlah<<endl;
        }
};

void main()
{
   clrscr();

   Buku novel, fiksi;

   novel.inisialisasi("Meriam Benteng Navarone","Alistair", 12);

   fiksi=novel;

   fiksi.info();
}
```

Alternatif penulisan fungsi anggota

Cara mendefinisikan fungsi anggota sebenarnya ada dua macam:

- Cara yang pertama adalah fungsi anggota didefinisikan didalam deklarasi kelas. (sesuai dengan contoh didepan)
- Bentuk kedua, pada kelas hanya terdapat deklarasi fungsi anggota (prototipe), lihat contoh berikut.

- Definisi fungsi anggota ditempatkan didalam kelas.

```
class Buku
{
    private:
    char judul[35];
    char pengarang[25];
    int jumlah;

    public:
    void inisialisasi(char *Judul, char *Pengarang, int Jumlah);
    void info();
};
```

- Definisi kedua fungsi anggota pada kelas Buku ditulis menjadi:

```
void Buku::inisialisasi(char *Judul, char *Pengarang, int Jumlah)
{   strcpy(judul, Judul);
    strcpy(pengarang, Pengarang);
    jumlah = Jumlah;
}
void Buku::info()
{
    cout<<"Judul        :"<<judul<<endl;
    cout<<"Pengarang    :"<<pengarang<<endl;
    cout<<"Jumlah Buku  :"<<jumlah<<endl;
}
```

```
#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);
   void info();

};

void main()
   {
        clrscr();

        Buku novel;

        novel.inisialisasi("Meriam Benteng Navarone","Alistair", 12);

        novel.info();
   }

void Buku::inisialisasi(char *Judul, char *Pengarang, int Jumlah)
   {    strcpy(judul, Judul);
        strcpy(pengarang, Pengarang);
        jumlah = Jumlah;
   }
void Buku::info()
   {
        cout<<"Judul                  :"<<judul<<endl;
        cout<<"Pengarang              :"<<pengarang<<endl;
        cout<<"Jumlah Buku            :"<<jumlah<<endl;
   }
```

Nilai balik fungsi anggota

- Apakah fungsi anggota selalu tidak mempunyai nilai balik?
- Fungsi anggota juga bisa mempunyai nilai balik

```
class Tanggal
{   private:
            int tanggal;
            int bulan;
            int tahun;
    public:
            void beri_tanggal(int tanggal, int bulan, int tahun);
            int info_tanggal()
            { return tanggal; };
            int info_bulan()
            { return bulan; };
            int info tahun()
            { return tahun; };
};
```