

chainfunction

```
1  #include <iostream>
2  using namespace std;
3  class buku
4  {
5      string judul;
6
7  public:
8      buku setJudul(string judul)
9      {
10         this->judul = judul;
11         return *this; // chain function
12     }
13     string getJudul()
14     {
15         return this->judul;
16     }
17
18 } bukunya;
19
20 int main()
21 {
22     // bukunya.setJudul("Matematika");
23     // cout << bukunya.getJudul();
24     cout << bukunya.setJudul("Matematika").getJudul(); // chain function calls
25     return 0;
26 }
```

destruktor

```
1  #include <iostream>
2  using namespace std;
3
4  class angka{
5  private:
6      int *arr;
7      int panjang;
8  public:
9      angka(int); //Constructor
10     ~angka(); //Destructor
11     void cetakData();
12     void isiData();
13 };
14 //Definisi member Functon
15 angka::angka(int i){ //Constructor
16     panjang = i;
17     arr = new int[i];
18     isiData();
19 }
20
21 angka::~angka(){ //Destructor
22     cout<<endl;
23     cetakData();
24     delete[] arr;
25     cout<<"Alamat Array Sudah Dilepaskan"<<endl;
26 }
27
28 void angka::cetakData(){
29     for(int i=1;i<=panjang;i++){
30         cout<<i<<" = "<<arr[i]<<endl;
31     }
32 }
33
34 void angka::isiData(){
35     for(int i=1;i<=panjang;i++){
36         cout<<i<<" = ";cin>>arr[i];
37     }
38     cout<<endl;
39 }
40
41 int main(){
42     angka belajarcpp(3); //Constructor Dipanggil
43     angka *ptrBelajarcpp = new angka(5); //Constructor Dipanggil
44     delete ptrBelajarcpp; //Destructor Dipanggil
45
46     return 0;
47 }//Destructor Dipanggil
```

konstruktor

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  class mahasiswa{
6  private:
7  int nim;
8  string nama;
9  public:
10  mahasiswa();
11  mahasiswa(int);
12  mahasiswa(string);
13  mahasiswa(int iNim, string iNama);
14  void cetak();
15  };
16
17  mahasiswa :: mahasiswa(){
18      nim = 0;
19      nama = "";
20  }
21
22  mahasiswa :: mahasiswa(int iNim){
23      nim=iNim;
24  }
25
26  mahasiswa :: mahasiswa(string iNama){
27      nama=iNama;
28  }
29
30  mahasiswa :: mahasiswa(int iNim, string iNama){
31      nim=iNim;
32      nama=iNama;
33  }
34  void mahasiswa::cetak(){
35      cout<<endl<< "Nim =" <<nim<<endl;
36      cout<< " Nama = " <<nama<<endl;
37  }
38
39  int main(){
40      mahasiswa mhs1;
41      mahasiswa mhs2(20);
42      mahasiswa mhs3("Indra");
43      mahasiswa mhs4(30, "Fauzan");
44
45      mhs1.cetak();
46      mhs2.cetak();
47      mhs3.cetak();
48      mhs4.cetak();
49
50      return 0;
51  }
52
```

mbrStatikfunc

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  class mahasiswa{
6  private:
7      static int nim;
8  public:
9      int id;
10     string nama;
11
12     void setID();
13     void printAll();
14
15     static void setNim(int pNim){nim=pNim; /*Definisi Function*/}
16     static int getNim(){return nim; /*Definisi Function*/}
17
18     mahasiswa(string pnama):nama(pnama){setID();}
19 };
20
21 int mahasiswa::nim=0;
22
23 void mahasiswa::setID(){
24     id = ++nim;
25 }
26
27 void mahasiswa::printAll(){
28     cout<<"ID    = "<<id<<endl;
29     cout<<"Nama  = "<<nama<<endl;
30     cout<<endl;
31 }
32
33 int main(){
34     mahasiswa mhs1("Sri Dadi");
35     mahasiswa mhs2("Budi Jatmiko");
36     mahasiswa::setNim(9); //mengakses nim melalui static member function "setNim"
37     mahasiswa mhs3("Andi Janu");
38     mahasiswa mhs4("Joko Wahono");
39
40     mhs1.printAll();
41     mhs2.printAll();
42     mhs3.printAll();
43     mhs4.printAll();
44
45     cout<<"akses dari luar object = "<<mahasiswa::getNim()<<endl; //mengakses nim melalui
static member function "getNim"
46
47     return 0;
48 }
```