

PRAKTIKUM 2: CLASSES AND OBJECTS

```
// di bawah ini adalah nama class, penamaan class sebaiknya diawali dengan
// huruf kapital dan menggunakan camel case, contoh nama class Satu, SatuDua,
// dan SatuDuaTiga
// class Mahasiswa merupakan Object Class karena nanti object dibuat dari class
// ini
class Mahasiswa {
    // nim dan nama adalah state, disebut juga sebagai attribute atau field
    // nim menggunakan tipe data String karena ada nim yang diawali
    // dengan angka 0

    // pada field di bawah ini perlu diberikan nilai awal/initiation
    // karena belum menggunakan constructor atau method untuk
    // memberikan nilai, pada class ini kita menggunakan default constructor
    // yaitu constructor yang disediakan secara otomatis oleh Java compiler
    // default constructor tidak perlu kita tulis dalam class

    String nim = "1844190001";
    String nama = "Andi";

    // di bawah ini adalah behavior atau disebut juga method, nama lain dari
    // fungsi
    // tipe method di bawah ini adalah void, yaitu tidak memberikan return
    // value, sehingga pada saat dipanggil dapat langsung diberikan fungsi
    // print
    void cetakNim() {
        System.out.println(nim);
    }

    // tipe method di bawah ini adalah non void, method ini harus
    // memberikan return value sesuai dengan tipe data dari method
    // tersebut
    String valNama() {
        return nama;
    }
}

// class TestMahasiswa adalah Driver Class, karena memiliki main method yang
// akan dipanggil pertama kali pada saat program dijalankan
class TestMahasiswa {
    public static void main(String[] args) {
        // mhs01 adalah nama object yang telah dibuat dengan
        // instantiation
        Mahasiswa mhs01 = new Mahasiswa();

        // di bawah ini memanggil field nim dan nama secara langsung
```

```

        // untuk kemudian dicetak
        System.out.println(mhs01.nim);
        System.out.println(mhs01.nama);

        // di bawah ini memanggil field nim dan nama secara tidak
        // langsung, yaitu melalui method yang telah disediakan

        // method cetakNim() dapat dipanggil langsung karena sudah ada
        // method print di dalam method tersebut
        mhs01.cetakNim();

        // untuk mencetak method valNama() harus dengan
        // menambahkan method print seperti dibawah ini, karena pada
        // saat method valNama() dipanggil hanya memberikan nilai saja
        System.out.println(mhs01.valNama());
    }
}

```

Catatan:

Sebaiknya class dibuat dengan file terpisah dan gunakan nama yang sama dengan nama class nya.

Contoh class Mahasiswa disimpan dengan nama file Mahasiswa.java, begitu juga dengan class TestMahasiswa.

Coba tambahkan sebuah object dari class Mahasiswa tersebut dengan nama mhs02 pada class TestMahasiswa dan panggil dengan cara yang sama!