

Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

MODUL 4

ABSTRACT CLASS DAN INTERFACE

PEMROGRAMAN BERORIENTASI OBJECT

Seluruh file pada praktikum ini tersedia pada: <https://github.com/afrzl/Pemrograman-Berorientasi-Objek/tree/main/Week%204/Praktikum>

A. ABSTRACT CLASS

1. Class Orang

```
/**
 * Orang
 */
import java.time.LocalDate;

abstract class Orang {
    String nama;
    LocalDate tanggalLahir;

    public Orang(String nama) {
        this.nama = nama;
    }

    public Orang(String nama, LocalDate tanggalLahir) {
        this.nama = nama;
        this.tanggalLahir = tanggalLahir;
    }

    public String getNama() {
        return nama;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }
}
```

Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

```
public LocalDate getTanggalLahir() {
    return tanggalLahir;
}

public void setTanggalLahir(LocalDate tanggalLahir) {
    this.tanggalLahir = tanggalLahir;
}

public abstract String getAlamat();

public abstract void setAlamat(String alamat);
}
```

2. Class Pegawai

```
/**
 * Pegawai
 */
import java.time.LocalDate;

public class Pegawai extends {
    private String NIP;
    private String namaKantor;
    private String unitKerja;
    private String alamat;

    Pegawai(String nama, String NIP, String namaKantor, String
unitKerja) {
        super(nama);
        this.NIP = nama;
        this.namaKantor = namaKantor;
        this.unitKerja = unitKerja;
    }

    Pegawai(String nama, LocalDate tanggalLahir, String NIP,
String namaKantor, String unitKerja) {
        super(nama, tanggalLahir);
    }
}
```

Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

```
this.NIP = nama;
this.namaKantor = namaKantor;
this.unitKerja = unitKerja;
}

public String getNIP() {
    return NIP;
}

public void setNIP(String nIP) {
    NIP = nIP;
}

public String getNamaKantor() {
    return namaKantor;
}

public void setNamaKantor(String namaKantor) {
    this.namaKantor = namaKantor;
}

public String getUnitKerja() {
    return unitKerja;
}

public void setUnitKerja(String unitKerja) {
    this.unitKerja = unitKerja;
}

public void getGaji() {
    System.out.println("7 juta");
}
```

Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

```
@Override
public String getAlamat() {
    return alamat;
}

@Override
public void setAlamat(String alamat) {
    this.alamat = alamat;
}

@Override
public String toString() {
    return ("Pegawai[" + super.toString() + ", NIP=" + NIP +
", NamaKantor=" + namaKantor + ", unitKerja=" + unitKerja +
"]" );
}
}
```

3. Class Test

```
/**
 * OrangTest
 */
public class OrangTest {

    public static void main(String[] args) {
        //Orang o = new Orang(); //error karena orang hanya
        abstract class, tidak bisa dibuat object
        Pegawai lutfi = new Pegawai("Lutfi", "1111", "STIS",
        "IT");
        lutfi.setAlamat("Otista 64 C");

        System.out.println(lutfi.getAlamat());
    }
}
```

Nama : Muh. Nur Afrizal
NIM : 222212738
Kelas / No : 2KS1 / 25

Output:

```
muham@Afrizal MINGW64 /g/My Drive/00. Tingkat 2/Pemrograman Berorientasi Objek/Week 4/Praktikum (main)
$ java OrangTest
Otista 64 C
```

B. INTERFACE

1. Interface Dosen

```
/**
 * Dosen
 */
interface Dosen {
    public String getNIDN();
    public void setNIDN(String NIDN);
    public String getKelompokKeahlian();
    public void setKelompokKeahlian(String kelompokKeahlian);
}
```

2. Abstract Class Orang

```
/**
 * Orang
 */
import java.time.LocalDate;

abstract class Orang {
    String nama;
    LocalDate tanggalLahir;

    public Orang(String nama) {
        this.nama = nama;
    }

    public Orang(String nama, LocalDate tanggalLahir) {
        this.nama = nama;
        this.tanggalLahir = tanggalLahir;
    }

    public String getNama() {
        return nama;
    }
}
```

Nama : Muh. Nur Afrizal
NIM : 222212738
Kelas / No : 2KS1 / 25

```
}

public void setNama(String nama) {
    this.nama = nama;
}

public LocalDate getTanggalLahir() {
    return tanggalLahir;
}

public void setTanggalLahir(LocalDate tanggalLahir) {
    this.tanggalLahir = tanggalLahir;
}

public abstract String getAlamat();

public abstract void setAlamat(String alamat);
}
```

3. Class Pegawai

```
/**
 * Pegawai
 */
import java.time.LocalDate;

public class Pegawai extends Orang implements Dosen {
    private String NIP;
    private String namaKantor;
    private String unitKerja;
    private String alamat;
    private String NIDN;
    private String keahlian;
}
```

Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

```
Pegawai(String nama, String NIP, String namaKantor, String
unitKerja) {
    super(nama);
    this.NIP = nama;
    this.namaKantor = namaKantor;
    this.unitKerja = unitKerja;
}

Pegawai(String nama, LocalDate tanggalLahir, String NIP,
String namaKantor, String unitKerja) {
    super(nama, tanggalLahir);
    this.NIP = nama;
    this.namaKantor = namaKantor;
    this.unitKerja = unitKerja;
}

public String getNIP() {
    return NIP;
}

public void setNIP(String nIP) {
    NIP = nIP;
}

public String getNamaKantor() {
    return namaKantor;
}

public void setNamaKantor(String namaKantor) {
    this.namaKantor = namaKantor;
}

public String getUnitKerja() {
    return unitKerja;
}
```

Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

```
public void setUnitKerja(String unitKerja) {
    this.unitKerja = unitKerja;
}

public void getGaji() {
    System.out.println("7 juta");
}

@Override
public String getAlamat() {
    return alamat;
}

@Override
public void setAlamat(String alamat) {
    this.alamat = alamat;
}

@Override
public String getNIDN() {
    return NIDN;
}

@Override
public void setNIDN(String NIDN) {
    this.NIDN = NIDN;
}

@Override
public String getKelompokKeahlian() {
    return keahlian;
}
```


Nama : Muh. Nur Afrizal

NIM : 222212738

Kelas / No : 2KS1 / 25

```
@Override
public void setKelompokKeahlian(String kelompokKeahlian) {
    this.keahlian = kelompokKeahlian;
}

@Override
public String toString() {
    return ("Pegawai[" + super.toString() + ", NIP=" + NIP +
", NamaKantor=" + namaKantor + ", unitKerja=" + unitKerja +
"]" );
}
}
```

4. Class Test

```
/**
 * OrangTest
 */
public class OrangTest {

    public static void main(String[] args) {
        //Orang o = new Orang(); //error karena orang hanya
        abstract class, tidak bisa dibuat object
        Pegawai lutfi = new Pegawai("Lutfi", "1111", "STIS",
"IT");
        lutfi.setAlamat("Otista 64 C");
        System.out.println(lutfi.getAlamat());

        lutfi.setNIDN("12345678");
        lutfi.setKelompokKeahlian("Computer Science");
        System.out.println("ada dosen " + lutfi.getNama() + "
dengan NIDN " + lutfi.getNIDN() + " kelompok " +
lutfi.getKelompokKeahlian());
    }
}
```

Nama : Muh. Nur Afrizal
NIM : 222212738
Kelas / No : 2KS1 / 25

Output:

```
muham@Afrizal MINGW64 /g/My Drive/00. Tingkat 2/Pemrograman Berorientasi Objek/Week 4/Praktikum (main)
$ java OrangTest
Otista 64 C
ada dosen Lutfi dengan NIDN 12345678 kelompok Computer Science
```