

Pertemuan 2: Properti dan Method dalam OOP

Tujuan Pembelajaran

Setelah mengikuti pertemuan ini, mahasiswa diharapkan dapat:

1. Memahami konsep properti dan method secara mendalam
2. Membedakan jenis-jenis properti (static vs instance)
3. Membedakan jenis-jenis method (static vs instance)
4. Menggunakan method getter dan setter
5. Memahami konsep method chaining
6. Implementasi method dengan parameter dan return value

Properti (Properties)

Definisi Properti

Properti adalah variabel yang didefinisikan dalam class yang digunakan untuk menyimpan data atau state dari sebuah object. Properti menentukan karakteristik yang dimiliki oleh object.

Jenis-jenis Properti

1. Instance Properties

- Properti yang dimiliki oleh setiap instance/object
- Setiap object memiliki nilai properti yang independen
- Diakses menggunakan `$this->namaProperti`

2. Static Properties

- Properti yang dimiliki oleh class, bukan object
- Semua instance berbagi nilai yang sama
- Diakses menggunakan `self::$namaProperti` atau `NamaClass::$namaProperti`
- Dideklarasikan dengan keyword `static`

Inisialisasi Properti

```
class ContohClass {  
    // Properti dengan nilai default  
    public $nama = "Default Name";  
    public $umur = 0;  
  
    // Static property  
    public static $jumlahInstance = 0;  
}
```

Method

Definisi Method

Method adalah fungsi yang didefinisikan dalam class yang menentukan perilaku atau aksi yang dapat dilakukan oleh object.

Jenis-jenis Method

1. Instance Method

- Method yang dipanggil dari instance/object
- Dapat mengakses instance properties menggunakan `$this`
- Dipanggil dengan `$object->namaMethod()`

2. Static Method

- Method yang dipanggil dari class, bukan object
- Tidak dapat mengakses instance properties
- Dapat mengakses static properties
- Dipanggil dengan `NamaClass::namaMethod()`
- Dideklarasikan dengan keyword `static`

Method dengan Parameter dan Return Value

```
public function namaMethod($parameter1, $parameter2 = "default") {  
    // Kode method  
    return $hasilReturn;  
}
```

Getter dan Setter Method

Getter Method

- Method yang digunakan untuk mengambil nilai properti
- Biasanya dimulai dengan kata "get"
- Memberikan kontrol akses terhadap properti

Setter Method

- Method yang digunakan untuk mengatur nilai properti
- Biasanya dimulai dengan kata "set"
- Dapat melakukan validasi sebelum mengatur nilai

```
class Contoh {  
    private $nilai;
```

```
// Getter
public function getNilai() {
    return $this->nilai;
}

// Setter
public function setNilai($nilai) {
    if ($nilai >= 0) {
        $this->nilai = $nilai;
    }
}
```

Method Chaining

Method chaining adalah teknik di mana beberapa method dapat dipanggil secara berurutan dalam satu baris kode. Ini dicapai dengan mengembalikan `$this` dari method.

```
class ChainExample {
    public function method1() {
        // Kode method
        return $this;
    }

    public function method2() {
        // Kode method
        return $this;
    }
}

// Penggunaan
$obj = new ChainExample();
$obj->method1()->method2();
```

Method Overloading vs Method Overriding

Method Overloading

- PHP tidak mendukung method overloading secara native
- Dapat disimulasikan menggunakan `__call()` magic method

Method Overriding

- Akan dipelajari lebih detail di pertemuan inheritance
- Mendefinisikan ulang method parent class di child class

Best Practices

1. Naming Convention

- Gunakan camelCase untuk nama method
- Method names harus deskriptif
- Getter: `getNamaProperti()`
- Setter: `setNamaProperti($nilai)`

2. Single Responsibility

- Setiap method harus memiliki tanggung jawab tunggal
- Method tidak boleh terlalu panjang

3. Parameter Validation

- Validasi parameter sebelum memproses
- Berikan error message yang jelas

4. Return Values

- Konsisten dalam tipe return value
- Dokumentasikan return value dengan comment

Contoh Implementasi

Lihat file `example.php` untuk berbagai contoh implementasi properti dan method di PHP.

Latihan

1. Buat class `Rekening` dengan properti saldo dan method debit/kredit
2. Implementasikan getter dan setter untuk semua properti
3. Buat static property untuk menghitung total rekening yang dibuat
4. Implementasikan method chaining untuk operasi berurutan

Tugas Rumah

Buat class `Produk` untuk sistem inventory dengan:

- Properti: nama, harga, stok, kategori
- Static property: totalProduk
- Method: tambahStok(), kurangiStok(), hitungNilaiInventory()
- Getter dan setter untuk semua properti
- Validasi untuk memastikan harga dan stok tidak negatif
- Method chaining untuk operasi berurutan