**#7 OOP DASAR pada PHP Inheritance (Bagian 2)**

Kita mau menerapkan inheritance kedalam kasus kita ya, nah nanti class Mobil ini kita bisa instance kedalam macam-macam mobil, ini misal semua mobil bisa belokKanan atau belokKiri tapi ada juga hanya mobil khusus yg bisa membuka atap mobil itu nah kalo hal yg tidak ada di class mobil kita bisa buat method baru di class MobilBukaAtap itu bisa kita sebut dengan inheritance.

Misal kita inheritance nya mobil sports saja ya gini :

class Mobil {

    public $nama, $merk, $warna, $kecepatanMaksimal, $jumlahPenumpang;

    public function tambahKecepatan() {

        return "Kecepatan bertambah!";

    }

}

Nah sekarang cara bikin child class nya yaitu MobilSport ya :

class Mobil {

    public $nama, $merk, $warna, $kecepatanMaksimal, $jumlahPenumpang;

    public function tambahKecepatan() {

        return "Kecepatan bertambah!";

    }

}

class MobilSport extends Mobil {

    public $turbo = false;

    public function jalankanTurbo() {

**$this**->turbo = true;

        return "Turbo dijalankan!";

    }

}

Buat nya kita seperti buat class pada umum nya Cuma kita tambahkan keyword extends lalu nama class parent nya. Yg artinya saya membuat class baru sebagai anak nya dari class mobil.

Dan cara kita instance ya begini :

$mobil = new MobilSport();

echo $mobil->tambahKecepatan();

echo "<br>";

echo $mobil->jalankanTurbo();

kita menginstance nya class MobilSport nah tapi inget class ini anak dari class Mobil jadi dia masih bisa memakai method atau property dari Class Mobil terkecuali sifat property atau method nya private tergantung akses di class Mobil nya ya.

Nah sekarang kita kembali ke studi kasus sebelum nya, tapi inheritance kita ini belum sempurna dan belum lengkap kita akan lengkapi bertahap di video seanjut nya.

Sebelum nya kita menambahkan hal baru yaitu jmlHalaman dan waktuMain fitur baru yg ingin kita tambahkan yaitu getInfoLengkap() dimana kita ingin menampilkan dari semua produk yang kita punya.

Yang akan kita selesaikan nya itu getInfoLengkap() bagian pengkondisian nya ya. Kita membuat class baru Komik dan Game, nah dengan melakukan itu saja berarti kita sudah bikin child class.

class Komik extends Produk {

}

class CetakInfoProduk{

    public function cetak(Produk $produk) {

        return $produk->getInfoLengkap();

    }

}

$produk2 = new Komik("Twenty Five Twenty One", "Jeong Jiman", "TVN", 1500000, 100, 0, "Drama Korea");

$produk3 = new Produk("Naruto", "Masashi Kishimoto", "Shonen Jump", 30000, 0, 50, "Game");

$newProduk1 = new CetakInfoProduk();

echo $newProduk1->cetak($produk2);

echo "<br>";

echo $newProduk1->cetak($produk3);

dengan membuat child class yaitu class komik dan isi nya kosong lalu saat instace new object yg dipanggil class komik hasil nya tidak error sama sekali wahhh keren. Itu membuktikan bahwa saya bisa menggunakan apapun yg dimiliki di class parent nya.

Pemanggilan new object komik dengan mengirimkan parameter itu di tangkep oleh constructor di class mobil ya, nah class komik tidak punya constructor. Nah salah satu perilaku dari inheritance kalo kita memmanggil sebuah method untuk sebuah child (anak) maka yg dilakukan adalah dia carikan dulu didalam class Komik ada method tersebut ga, kalo gaada otomatis dia akan carikan di parent nya (emak nya).

Kita akan membuat function di class komik nama nya sama getInfoLengkap() seperti di parent nya nah itu untuk memanggil info produk lengkap nya khsus yg komik aja.

Nah ini yg kita lakukan sama saja Cuma mehilangkan property tipe jmlHalaman dan waktuMain beserta if else di getInfoLengkap() si parent nya, gini :

class Komik extends Produk {

    public function getInfoLengkap() {

        $str = "Komik : {**$this**->judul} | {**$this**->getLabel()}, (Rp. {**$this**->harga}) - {**$this**->jmlHalaman} Halaman";

        return $str;

    }

}

Nah buat juga untuk class Game nya, sekarang kita sudah membuat inherintace, memang agak aneh kaya jadi banyakin class aja tapi ini belum selesai jika sudah sampai tahapan kompleks akan terasa enak dilihat, dan ini akan memudahkan ketika fungsionalitas dari child ini sudah tambah banyak.

Nah kalian liat di class produk function getInfoProduk() itu bisa kita jalankan di class komik atau class anak nya, nah gimana tuh cara memanggil nya, nah sebetul nya ada fungsi yg namanya overriding, dimana kita bisa menggunakan fungsi dengan nama yg sama tapi punya nya class Parent nah pembahasan nya ada di video setelah ini.