

CHAPTER 09

OBJECT ORIENTED PROGRAMMING

ACADEMY.CYBERKARTA.COM

OOP (Object Oriented Programming)

- Sebuah pergantian paradigma program
 - saat ini (**prosedural**) → program = instruksi yang dijalankan secara urut
 - **OOP** → program = interaksi dari elemen-elemen
- Kelebihan OOP
 - mempermudah abstraksi entitas
 - membuat kode lebih terorganisir
 - intinya: mempermudah pengelolaan kode (penting saat kode sudah 5000+ baris)
- Kekurangan OOP
 - overkill untuk project kecil
 - butuh definisi kode lebih panjang

Konsep OOP (I)

Konsep	Deskripsi	Implementasi
Class	Rancangan untuk membuat object, terdiri atas attribute & method	<pre>class User: def __init__(self, username, password): self.username = username self.password = password</pre>
Object	Perwujudan dari class yang akan berinteraksi dalam program.	<pre>u = User("root", "password")</pre>
Method	Function yang melekat dalam class.	<pre>def check_username_availability(username):</pre>

Konsep OOP (II)

Konsep	Deskripsi	Implementasi
Attribute	Variable yang melekat pada class. Dapat di-assign dan manipulasi ketika sudah dibuat object-nya.	<pre>u = user("root", "password") u.username = "nonroot"</pre>
Constructor	Method khusus untuk inisialisasi object dari class. Syntax Python: <code>__init__</code>	<pre>def __init__(self, username, password):</pre>
Encapsulation	Sembunyikan detail attribute class, hanya bisa akses lewat method khusus getter/setter	<pre>self.__password (private)</pre>

Konsep OOP (III)

Konsep	Deskripsi	Implementasi
Inheritance	Class bisa mengikuti definisi class lain, dengan ditambah pengembangan logika	<pre>class Admin(User):</pre>
Polymorphism	Kemampuan override method yang ada menjadi method baru	<pre>def login(self): # (overridden in subclass)</pre>
Abstraction	Deklarasi interface dahulu, baru didefinisikan isinya kemudian	<pre>def connect(self): # (defined in base class, implemented in subclass)</pre>
Decorator	tambahan keterangan yang mengubah behavior class, syntax @	<pre>@dataclass class User:</pre>

- **SSH Brute Force, with Class** 🍷
 - Buat SSH brute force dengan paramiko
 - menyerang 1 IP sesuai input
 - bisa membaca credential dari file external “credentials.txt”
 - bisa menulis hasil attack ke file external “attack_log_192.168.1.21.txt”
 - Definisikan dataclass untuk data credential dan log attack
 - Definisikan class untuk SSH brute force dengan method berikut
 - baca input credential
 - eksekusi serangan
 - logging hasil serangan
 - Jangan lupa DRIVE framework
 - ⚠ **Kode hanya untuk keperluan edukasi. Jangan dijalankan di mesin yang tidak kamu miliki *authorization*-nya.**