

# Roadmap CRUD E-Commerce Python untuk Kelas XI

## Bab 10: Keamanan Dasar Aplikasi (Password Hashing & Validasi Input)

### Tujuan Pembelajaran

- Siswa memahami pentingnya keamanan dalam aplikasi e-commerce.
- Siswa dapat mengimplementasikan hashing pada password user.
- Siswa mampu melakukan validasi input untuk mencegah error dan celah keamanan.

### Konsep Dasar

1. **Password Hashing:** Password tidak boleh disimpan dalam bentuk teks biasa. Gunakan algoritma hashing seperti `bcrypt` agar lebih aman.
2. **Validasi Input:** Semua data dari user harus divalidasi (misalnya tidak boleh kosong, tipe data sesuai, format email benar).

### Langkah Praktik

#### 1. Install Library Bcrypt

```
pip install bcrypt
```

#### 2. Menyimpan Password dengan Hashing

```
import bcrypt

def hash_password(password):
    salt = bcrypt.gensalt()
    hashed = bcrypt.hashpw(password.encode('utf-8'), salt)
    return hashed

def verify_password(password, hashed):
    return bcrypt.checkpw(password.encode('utf-8'), hashed)

# Contoh penggunaan
password = "12345"
hashed_pw = hash_password(password)
print(hashed_pw)
print(verify_password("12345", hashed_pw)) # True
```

Simpan `hashed_pw` ke database, bukan password asli.

### 1. Validasi Input Registrasi

```
def registrasi_user(username, password):
    if not username or not password:
        print("Username dan password tidak boleh kosong!")
        return
    if len(password) < 5:
        print("Password minimal 5 karakter!")
        return

    hashed_pw = hash_password(password)

    conn = sqlite3.connect('ecommerce.db')
    cursor = conn.cursor()
    cursor.execute("INSERT INTO users (username, password) VALUES
    (?, ?)", (username, hashed_pw))
    conn.commit()
    conn.close()
    print("User berhasil didaftarkan dengan aman!")
```

### 2. Validasi Input Produk

```
def tambah_produk(nama, harga):
    if not nama:
        print("Nama produk tidak boleh kosong!")
        return
    if harga <= 0:
        print("Harga harus lebih dari 0!")
        return

    conn = sqlite3.connect('ecommerce.db')
    cursor = conn.cursor()
    cursor.execute("INSERT INTO produk (nama, harga) VALUES (?, ?)",
    (nama, harga))
    conn.commit()
    conn.close()
    print("Produk berhasil ditambahkan!")
```

## Tugas Siswa

1. Implementasikan **password hashing** saat user registrasi dan login.
2. Tambahkan **validasi input** pada semua form (registrasi, login, tambah produk, checkout).
3. Uji dengan mencoba input kosong atau password yang sangat pendek.

## Refleksi

Dengan menambahkan **keamanan dasar**, aplikasi e-commerce Python menjadi lebih profesional, aman digunakan, dan siap untuk dipamerkan.