

# **SISTEM PENGELOLAAN BUKU DI PERPUSTAKAAN KAMPUS - RABU (C)**

Di tengah kesibukan mahasiswa Universitas Kristen Duta Wacana, perpustakaan kampus menjadi tempat favorit untuk mencari bahan referensi. Terutama bagi mahasiswa tingkat akhir yang sedang dikejar deadline skripsi, ketersediaan buku sangat menentukan.

Untuk menghindari antrean panjang dan kebingungan dalam pencatatan, pihak perpustakaan ingin membuat sistem peminjaman buku otomatis yang simpel namun efektif.

Sebagai mahasiswa informatika, kamu diminta untuk membantu merancang program manajemen **pengembalian buku** dengan ketentuan berikut

## **Aturan Program Pengembalian:**

### **1. Validasi Pengembalian**

- Sistem hanya boleh memproses pengembalian jika buku yang dikembalikan memang tercatat sedang dipinjam oleh mahasiswa tersebut.
- Jika buku tidak tercatat sebagai pinjaman mahasiswa, maka otomatis gagal diproses.
- 

### **2. Update Data Buku**

- Jika **pengembalian berhasil**, stok buku **bertambah 1**.
- Jika **stok** buku sebelumnya **0**, maka status buku **berubah dari "Kosong"** menjadi **"Tersedia"**.
- 

### **3. Update Mahasiswa**

- Jika **pengembalian berhasil**, buku tersebut **dihapus dari daftar `pinjaman` mahasiswa**.

### **4. Laporan**

- Setiap transaksi pengembalian menghasilkan laporan yang memuat:
  - Daftar buku yang **berhasil dikembalikan**.
  - Daftar buku yang **gagal diproses** beserta alasannya (contoh: tidak tercatat sebagai pinjaman).
  - Data buku terbaru (**stok & status**).
  - Data mahasiswa terbaru (**daftar pinjaman yang sudah diperbarui**).

## **Parameter Program:**

Dictionary "buku" : menyimpan data (nama, ketersediaan, stok) buku

Dictionary "mahasiswa" menyimpan data dan keperluan mahasiswa

Anda diminta untuk melengkapi skeleton program **Pengembalian Buku** sesuai dengan aturan yang tersedia agar memunculkan output yang diminta. Gunakan dictionary buku dan mahasiswa untuk membantu anda menyelesaikan program. Semangat :>

**Jika berhasil** menyelesaikan program untuk menjalankan streamlit, untuk menjalankannya jangan lupa pindah ke **direktori tempat file app.py berada**, kemudian **streamlit run app.py**, jika tidak terselesaikan, cukup menjalankan program **pengembalian.py** seperti menjalankan program python pada umumnya.

### Test Case:

1. Sukses - (25 point)

Input:

```
{
  "name": "1. Sukses sederhana (kembali 1 buku)",
  "buku": {"algoritma dasar": {"stok": 0, "status": "Kosong"}},
  "mahasiswa": {"Rina": {"pinjaman": ["algoritma dasar"], "ingin_kembali": ["algoritma dasar"]}}
},
```

Output:

```
TEST: 1. Sukses sederhana (kembali 1 buku)
=== Laporan Pengembalian ===
Mahasiswa: Rina
  Berhasil dikembalikan : algoritma dasar

=== Data Buku Terbaru ===
- algoritma dasar | stok: 1 | status: Tersedia

=== Data Mahasiswa Terbaru ===
- Rina | pinjaman: -
```

2. Gagal (tidak pernah meminjam) - (30 point)

Input:

```
{
  "name": "2. Gagal - tidak pernah pinjam",
  "buku": {"basis data": {"stok": 1, "status": "Tersedia"}},
  "mahasiswa": {"Budi": {"pinjaman": [], "ingin_kembali": ["basis data"]}}
},
```

Output:

```

TEST: 2. Gagal - tidak pernah pinjam
=== Laporan Pengembalian ===
Mahasiswa: Budi
Gagal diproses      : basis data (Tidak tercatat sebagai pinjaman)

=== Data Buku Terbaru ===
- basis data | stok: 1 | status: Tersedia

- basis data | stok: 1 | status: Tersedia

=== Data Mahasiswa Terbaru ===
- Budi | pinjaman: -

- Budi | pinjaman: -

```

### 3. Kombinasi pengembalian (ada yang sukses ada yang gagal) - (35 poin)

Input:

```

{
  "name": "3. Beberapa pengembalian campuran (ada yang sukses & gagal)",
  "buku": {
    "struktur data": {"stok": 0, "status": "Kosong"},
    "logika matematika": {"stok": 1, "status": "Tersedia"}
  },
  "mahasiswa": {
    "Citra": {"pinjaman": ["struktur data"], "ingin_kembali": ["struktur data", "logika matematika"]}
  }
},

```

Output:

```

=====
TEST: 3. Beberapa pengembalian campuran (ada yang sukses & gagal)
=== Laporan Pengembalian ===
Mahasiswa: Citra
Berhasil dikembalikan : struktur data
Gagal diproses      : logika matematika (Tidak tercatat sebagai pinjaman)

=== Data Buku Terbaru ===
- struktur data | stok: 1 | status: Tersedia
- logika matematika | stok: 1 | status: Tersedia

=== Data Mahasiswa Terbaru ===
- Citra | pinjaman: -

```

### 4. Tidak mengembalikan apapun (10 poin)

Input:

```
{
  "name": "4. Tidak ada yang dikembalikan",
  "buku": {"pemrograman python": {"stok": 2, "status": "Tersedia"}},
  "mahasiswa": {"Deni": {"pinjaman": ["pemrograman python"], "ingin_kembali": []}}
},
```

Output:

```
TEST: 4. Tidak ada yang dikembalikan
=== Laporan Pengembalian ===
Mahasiswa: Deni
  Tidak ada buku yang dikembalikan.

=== Data Buku Terbaru ===
- pemrograman python | stok: 2 | status: Tersedia

=== Data Mahasiswa Terbaru ===
- Deni | pinjaman: pemrograman python
```

### Bonus (10 poin):

Isi variabel (pemberitahuan, sukses, gagal) dengan fungsi dari streamlit yang benar sehingga bisa menghasilkan output dengan benar...

```
# =====
# Bonus: uncomment variabel (pemberitahuan, sukses, gagal)
# kemudian definisikan value yang seharusnya
# =====

# pemberitahuan =
# sukses =
# gagal =
```

Output **jika berhasil** menjalankan Streamlitnya

## Sistem Pengembalian Buku

Pilih skenario uji:

1. Sukses sederhana (kembali 1 buku) ▾

Proses Pengembalian