

LAMPIRAN

CODE WEB SERVER

```
from flask import Flask, render_template, request, redirect, url_for, session, flash
from datetime import datetime
import pandas as pd
import os

app = Flask(__name__)
app.secret_key = 'perpustakaan123'

# Inisialisasi file Excel
def init_excel():
    if not os.path.exists('library_data.xlsx'):
        books_df = pd.DataFrame({
            'id': [],
            'judul': [],
            'pengarang': [],
            'tahun_terbit': [],
            'status': [],
            'rak': []
        })
        loans_df = pd.DataFrame({
            'id_buku': [],
            'judul_buku': [],
            'nama_siswa': [],
            'kelas': [],
            'tanggal_pinjam': [],
            'tanggal_kembali': [],
            'status': []
        })
    with pd.ExcelWriter('library_data.xlsx') as writer:
        books_df.to_excel(writer, sheet_name='books', index=False)
        loans_df.to_excel(writer, sheet_name='loans', index=False)
```

```

@app.route('/')
def home():
    return render_template('index.html')

@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        if request.form['username'] == 'admin' and request.form['password'] == 'admin123':
            session['logged_in'] = True
            return redirect('/')
        else:
            flash('Username atau password salah!', 'danger')
    return render_template('login.html')

@app.route('/logout')
def logout():
    session.clear()
    return redirect('/')

@app.route('/books', methods=['GET', 'POST'])
def books():
    df = pd.read_excel('library_data.xlsx', sheet_name='books')
    if request.method == 'POST' and 'judul' in request.form:
        new_book = {
            'id': len(df) + 1,
            'judul': request.form['judul'],
            'pengarang': request.form['pengarang'],
            'tahun_terbit': request.form['tahun_terbit'],
            'status': request.form['status'],
            'rak': request.form['rak']
        }
        df = pd.concat([df, pd.DataFrame([new_book])], ignore_index=True)
        with pd.ExcelWriter('library_data.xlsx', engine='openpyxl', mode='a', if_sheet_exists='replace') as writer:
            df.to_excel(writer, sheet_name='books', index=False)
    return redirect('/books')

```

```

keyword = request.args.get('search', '').lower()

if keyword:
    df = df[df['judul'].str.lower().str.contains(keyword)]

return render_template('books.html', books=df.to_dict('records'))

@app.route('/loans', methods=['GET', 'POST'])
def loans():
    if not session.get('logged_in'):
        return redirect(url_for('login'))

    loans_df = pd.read_excel('library_data.xlsx', sheet_name='loans')
    books_df = pd.read_excel('library_data.xlsx', sheet_name='books')

    if request.method == 'POST':
        judul_buku = request.form['judul_buku']
        nama_siswa = request.form['nama_siswa']
        kelas = request.form['kelas']
        tanggal_pinjam = request.form['tanggal_pinjam']
        tanggal_kembali = request.form['tanggal_kembali']

        new_loan = {
            'id_buku': None,
            'judul_buku': judul_buku,
            'nama_siswa': nama_siswa,
            'kelas': kelas,
            'tanggal_pinjam': tanggal_pinjam,
            'tanggal_kembali': tanggal_kembali,
            'status': 'Dipinjam'
        }

        loans_df = pd.concat([loans_df, pd.DataFrame([new_loan])], ignore_index=True)
        books_df.loc[books_df['judul'] == judul_buku, 'status'] = 'Dipinjam'

        with pd.ExcelWriter('library_data.xlsx', engine='openpyxl', mode='a', if_sheet_exists='replace') as writer:
            books_df.to_excel(writer, sheet_name='books', index=False)
            loans_df.to_excel(writer, sheet_name='loans', index=False)

        return redirect('/loans')

    available_books = books_df[books_df['status'].str.lower() == 'tersedia']

```

```

    return render_template('loans.html', loans=loans_df.to_dict('records'),
books=available_books.to_dict('records'))

@app.route('/returns', methods=['GET', 'POST'])
def returns():
    if not session.get('logged_in'):
        return redirect(url_for('login'))

    df_loans = pd.read_excel('library_data.xlsx', sheet_name='loans')
    df_books = pd.read_excel('library_data.xlsx', sheet_name='books')

    if request.method == 'POST':
        index = int(request.form['index'])
        judul = df_loans.iloc[index]['judul_buku']

        df_loans.at[index, 'status'] = 'Dikembalikan'

        df_books.loc[df_books['judul'] == judul, 'status'] = 'Tersedia'

        with pd.ExcelWriter('library_data.xlsx', engine='openpyxl', mode='a', if_sheet_exists='replace') as
writer:

            df_loans.to_excel(writer, sheet_name='loans', index=False)

            df_books.to_excel(writer, sheet_name='books', index=False)

        return redirect('/returns')

    borrowed = df_loans[df_loans['status'] == 'Dipinjam'].reset_index()

    return render_template('returns.html', loans=borrowed.to_dict('records'))

@app.route('/overdue')
def overdue():
    df = pd.read_excel('library_data.xlsx', sheet_name='loans')

    df['tanggal_kembali'] = pd.to_datetime(df['tanggal_kembali'], errors='coerce')

    today = datetime.now()

    overdue_loans = df[(df['status'] == 'Dipinjam') & (df['tanggal_kembali'] < today)].to_dict('records')

    return render_template('overdue.html', overdue_loans=overdue_loans)

if __name__ == '__main__':
    init_excel()
    app.run(debug=True)

```

LAMPIRAN

INDEX.HTML

```
<!DOCTYPE html>
<html lang="id">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Perpustakaan SDN 112152 Bandar Tinggi</title>
  <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
</head>
<body>
  <header>
    <h1>Perpustakaan 112152 Bandar Tinggi</h1>
    <nav>
      <a href="/">Beranda</a>
      <a href="/books">Daftar Buku</a>
      <a href="/loans">Peminjaman</a>
      <a href="/overdue">Keterlambatan</a>
      <a href="/returns">Pengembalian</a>
    </nav>
  </header>
  <main>
    <div class="menu-cards">
      <div class="card">
        <h3>Daftar Buku</h3>
        <p>Lihat koleksi buku perpustakaan</p>
        <a href="/books" class="button">Lihat Buku</a>
      </div>
      <div class="card">
        <h3>Peminjaman</h3>
        <p>Kelola peminjaman buku</p>
```

```
<a href="/loans" class="button">Kelola Peminjaman</a>
</div>
<div class="card">
  <h3>Keterlambatan</h3>
  <p>Daftar buku yang belum dikembalikan tepat waktu</p>
  <a href="/overdue" class="button">Lihat Keterlambatan</a>
</div>
<div class="card">
  <h3>Pengembalian</h3>
  <p>Konfirmasi pengembalian buku oleh siswa</p>
  <a href="/returns" class="button">Kelola Pengembalian</a>
</div>
</div>
<div class="login-link">
  <a href="/login" class="admin-login-btn">🔒 Masuk sebagai Admin</a>
</div>
</main>
</body>
</html>
```

LAMPIRAN

STYLE.CSS

```
body {
    font-family: Arial, sans-serif;
    background-color: #f4f4f4;
    margin: 0;
    padding: 0;
}

header {
    background-color: #20ba15;
    color: white;
    text-align: center;
    padding: 1rem;
}

nav {
    margin-top: 1rem;
}

nav a {
    color: black;
    background-color: white;
    text-decoration: none;
    padding: 0.5rem 1rem;
    margin: 0 0.5rem;
    border-radius: 3px;
    border: 1px solid #ccc;
    font-weight: bold;
    transition: background-color 0.3s, color 0.3s;
}

nav a:hover {
    background-color: #ddd;
    color: black;
}

main {
```

```
    max-width: 1200px;
    margin: 2rem auto;
    padding: 0 1rem;
  }
  .menu-cards {
    display: flex;
    flex-wrap: wrap;
    justify-content: center;
    gap: 1.5rem;
  }
  .card {
    background-color: white;
    padding: 1rem;
    border-radius: 8px;
    box-shadow: 0 4px 10px rgba(0, 0, 0, 0.1);
    width: 250px;
    text-align: center;
  }
  .card h3 {
    margin-top: 0;
  }
  .button {
    display: inline-block;
    margin-top: 10px;
    padding: 0.5rem 1rem;
    background-color: #2193b0;
    color: white;
    text-decoration: none;
    border-radius: 5px;
    transition: 0.3s;
  }
  .button:hover {
    background-color: #17657e;
  }
```

```
}  
.login-link {  
    margin-top: 40px;  
    text-align: center;  
}  
.admin-login-btn {  
    display: inline-block;  
    padding: 14px 30px;  
    font-size: 16px;  
    background: linear-gradient(to right, #00b4db, #0083b0);  
    color: #fff;  
    font-weight: bold;  
    text-transform: uppercase;  
    border: none;  
    border-radius: 50px;  
    text-decoration: none;  
    transition: 0.4s ease;  
    box-shadow: 0 4px 15px rgba(0, 179, 219, 0.4);  
    position: relative;  
    overflow: hidden;  
}  
.admin-login-btn:hover {  
    background: linear-gradient(to right, #0083b0, #00b4db);  
    transform: scale(1.05);  
    box-shadow: 0 6px 20px rgba(0, 179, 219, 0.5);  
}  
.login-page {  
    font-family: 'Segoe UI', sans-serif;  
    background: linear-gradient(to right, #444d4f, #9da0a1);  
    display: flex;  
    justify-content: center;  
    align-items: center;  
    height: 100vh;
```

```
    margin: 0;
}
.login-box {
    background: white;
    padding: 30px 40px;
    border-radius: 12px;
    box-shadow: 0 8px 25px rgba(0, 0, 0, 0.2);
    width: 320px;
}
.login-box h2 {
    text-align: center;
    margin-bottom: 20px;
    color: #333;
}
.login-box label {
    font-weight: bold;
    display: block;
    margin-top: 10px;
}
.login-box input[type="text"],
.login-box input[type="password"] {
    width: 100%;
    padding: 10px;
    border: 1px solid #ccc;
    border-radius: 8px;
    margin-top: 5px;
}
.login-box input[type="submit"] {
    margin-top: 20px;
    width: 100%;
    padding: 10px;
    background-color: #2193b0;
    border: none;
```

```
    color: white;
    font-weight: bold;
    border-radius: 8px;
    cursor: pointer;
    transition: background-color 0.3s;
}
.login-box input[type="submit"]:hover {
    background-color: #17657e;
}
.error-message {
    color: red;
    text-align: center;
    margin-top: 10px;
}
body {
    font-family: Arial, sans-serif;
    background-color: #f4f4f4;
}
h1, h2 {
    text-align: center;
    margin-top: 2rem;
}
.book-table {
    width: 90%;
    margin: 1rem auto;
    border-collapse: collapse;
}
.book-table th, .book-table td {
    border: 1px solid #aaa;
    padding: 10px;
    text-align: left;
}
.book-table th {
```

```
    background-color: #333;
    color: white;
}

.book-table tr:nth-child(even) {
    background-color: #f9f9f9;
}

.book-form {
    width: 60%;
    margin: 2rem auto;
    background: #fff;
    padding: 1rem;
    border-radius: 5px;
    box-shadow: 0 0 5px rgba(0,0,0,0.1);
}

.book-form input, .book-form select {
    width: 100%;
    padding: 10px;
    margin: 8px 0;
    box-sizing: border-box;
}

.book-form button {
    padding: 10px 20px;
    background: #333;
    color: white;
    border: none;
    cursor: pointer;
}

.book-form button:hover {
    background: #444;
}

.search-form {
    width: 60%;
    margin: 2rem auto;
```

```
    display: flex;
    gap: 10px;
}

.search-form input {
    flex: 1;
    padding: 10px;
}

.search-form button {
    padding: 10px 20px;
    background: #2193b0;
    color: white;
    border: none;
    cursor: pointer;
    border-radius: 4px;
}

.search-form button:hover {
    background-color: #17657e;
}

.loan-table {
    width: 90%;
    margin: 2rem auto;
    border-collapse: collapse;
}

.loan-table th, .loan-table td {
    border: 1px solid #aaa;
    padding: 10px;
    text-align: left;
}

.loan-table th {
    background-color: #333;
    color: white;
}

.loan-table tr:nth-child(even) {
```

```
        background-color: #f4f4f4;
    }
    .loan-form {
        width: 60%;
        margin: 2rem auto;
        background: #f9f9f9;
        padding: 1rem;
        border: 1px solid #ccc;
        border-radius: 5px;
    }
    .loan-form input,
    .loan-form select {
        width: 100%;
        padding: 10px;
        margin: 8px 0;
        box-sizing: border-box;
    }
    .loan-form button {
        padding: 10px 20px;
        background: #333;
        color: white;
        border: none;
        cursor: pointer;
    }
    .loan-form button:hover {
        background: #444;
    }
    .overdue-table {
        width: 90%;
        margin: 2rem auto;
        border-collapse: collapse;
    }
    .overdue-table th, .overdue-table td {
```

```
border: 1px solid #aaa;
padding: 10px;
text-align: left;
}
.overdue-table th {
background-color: #c0392b;
color: white;
}
.overdue-table tr:nth-child(even) {
background-color: #f9f9f9;
}
h1 {
text-align: center;
margin-top: 2rem;
font-size: 24px;
color: #2c3e50;
}
.returns-table {
width: 90%;
margin: 2rem auto;
border-collapse: collapse;
}
.returns-table th, .returns-table td {
border: 1px solid #aaa;
padding: 10px;
text-align: left;
}
.returns-table th {
background-color: #333;
color: white;
}
.returns-table tr:nth-child(even) {
background-color: #f9f9f9;
```

```
    }  
h1 {  
text-align: center;  
    margin-top: 2rem;  
    font-size: 24px;  
    color: #2c3e50;  
}  
.btn-return {  
    padding: 6px 12px;  
    background-color: green;  
    color: white;  
    border: none;  
    cursor: pointer;  
}  
.btn-return:hover {  
    background-color: darkgreen;  
}
```