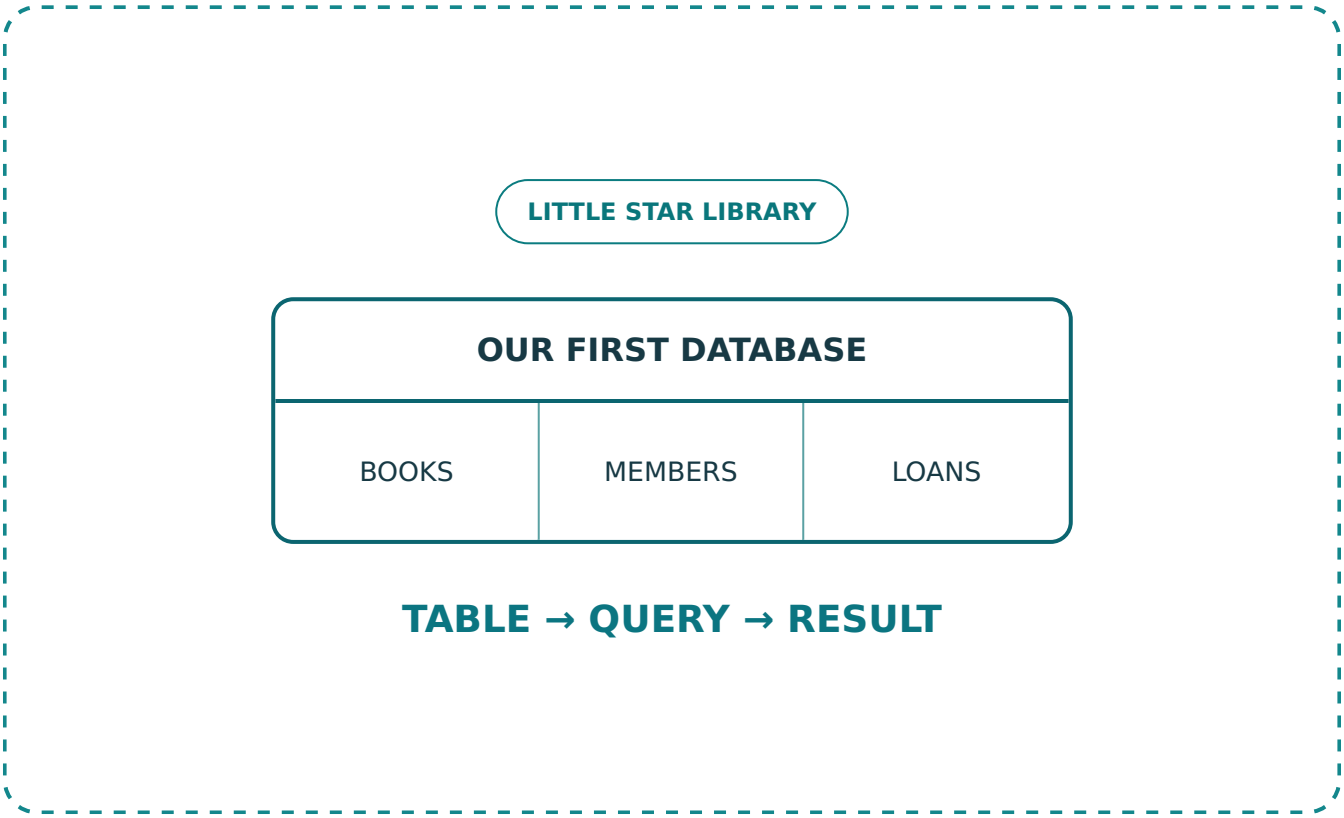
BEGINNER FRIENDLY • 2026

Database කියන්නේ මොකක්ද?

Introduction to Databases

දරුවන්ටත් නවකයන්ටත් සරල සිංහල පාඩමක් • Python + SQLite + SQL



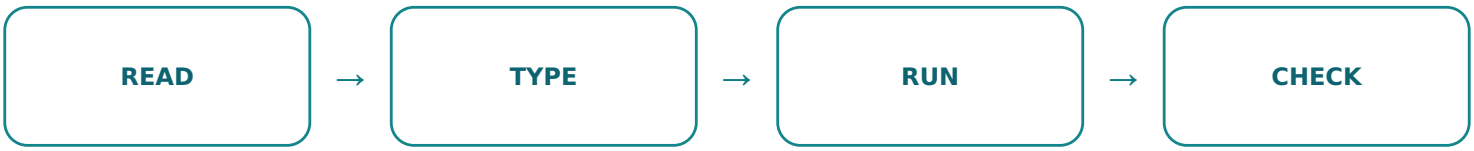
SQLite database එකක් හදා books search කිරීම, borrow/return කිරීම, report එකක් සෑදීම සහ safe backup එකක් ගැන පියවරෙන් පියවර ඉගෙනගනිමු.

HOW TO LEARN

මේ පොත භාවිතා කරන්නේ කොහොමද?

READ → TRY → CHECK → EXPLAIN

මුලින් පිටුවේ කුඩා කතාව කියවන්න. ඉන්පසු table/diagram/code බලන්න. ඒකෙන් පස්සේ "ඔබම කර බලන්න" අභ්‍යාසය කරන්න. සෑම SQL query එකකටම එය data වෙනස් කරනවාද, කියවනවාද කියලා සිතා බලන්න.



මෙම පොතේ අත්හදා බැලීම්

සියලුම names, books, dates සහ test inputs කල්පිත learning data. VS Code සහ Windows CMD panels ඉගැන්වීම සඳහා සකස් කළ previews. Code එක ඉංග්‍රීසියෙන්, විස්තර සිංහලෙන්.

> ✎ **ඉගෙනගැනීමේ අරමුණ**
> අවසානයේ table එකක් සෑදීම, SQL query එකක් ලිවීම, Python database එකකට connect වීම සහ CRUD/Test/Backup කිරීම ඔබටම පැහැදිලි කළ හැකි විය යුතුයි.

LEARNING ROADMAP

අපේ ඉගෙනුම් ගමන

Database එකක් තේරුම්ගෙන කුඩා app එකක් දක්වා

DATABASE → SQL → PYTHON → QA

කොටස	ඉගෙනගන්නා දෙය
01	Table, Row, Column, PK/FK, DBMS
02	Python / VS Code / CMD / SQLite setup
03	CREATE, INSERT, SELECT, UPDATE, DELETE, JOIN
04	Little Star Library: Books / Members / Loans
05	Testing, QA, backup, privacy, 2026 skills
06 - 08	MCQ, small labs, code references, final challenge

තනි පිටු උඩින් පනින්න එපා. අමුතු වචනයක් ආවොත් glossary එක හෝ diagram එක නැවත බලන්න.

PROJECT PREVIEW

Little Star Library: අපි හදන app එක

Windows CMD + Python + SQLite

```
Windows Command Prompt

Little Star Library - egotechworld.com
1 Books  2 Search  3 Add book  4 New member
5 Borrow 6 Return  7 Active loans  8 Categories
9 Export CSV  10 Backup  11 Chrome report  0 Exit
Choose: 1
1 | Space Adventures | available: 3
2 | Little Python      | available: 2
3 | Garden Secrets     | available: 4
```

මේ පිටුවේ preview එක source project එකේ operations අනුසාරයෙන් නිර්මාණය කළ demo එකක්. ඔබගේ output format එකේ සුළු වෙනස්කම් තිබිය හැක.

PROJECT GOAL

Book එකක් සුරකින්න. Search කරන්න. Member එකකට book එකක් lend කරන්න. Return කර stock restore කරන්න. අවසානයේ backup එකක් ගන්න.

PART 01

දත්ත ගබඩාව තේරුම් ගනිමු

අපේ ඉගෙනුම් පොතේ මීළඟ පියවර

CHAPTER 01

දත්ත ගබඩාව තේරුම් ගනිමු

CONCEPT → EXAMPLE → PRACTICE → CHECK

මෙම කොටස අවසන් කරද්දී වැදගත් වචන ඔබේම වචන වලින් පැහැදිලි කරන්න. Code එක copy කරන්නට පමණක් නොව එය කරන වැඩේ තේරුම් ගැනීම අරමුණයි.

✎ **පාඩමට කලින්**

මේ කොටස ගැන ඔබට දැන්ම දන්නා කරුණු දෙකක් කඩදාසියක ලියන්න. අවසානයේ ඒවා නැවත බලන්න.

LESSON

Database එකක් කියන්නේ මොකක්ද?

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

පුංචි පුස්තකාලයක පොත් විස්තර කඩදාසි පොතක ලිව්වොත්, පොත් වැඩි වන විට සෙවීම අපහසු වෙනවා. Database එකක් කියන්නේ ඒ විස්තර ක්‍රමවත් ලෙස සුරැකීමට උපකාරී වන පද්ධතියක්.

දැනගත යුතු වැදගත් අදහස

අපේ උදාහරණය: පොතේ නම, කතෘ, පොත් ගණන සහ ඒ පොත ලබාගත් සාමාජිකයා යන දත්ත එකට සම්බන්ධ කර තබමු.



✎ **ඔබම කර බලන්න**

ඔබේ ගෙදර පොත් 5ක නම් ලියා තබන්න. ඒවා සෙවීමට කොහොමද?

★ Database එකක් යනු data එක හා එය පාලනය කරන software එක අතර වෙනසක් ඇති දෙයකි.

LESSON

Data සහ Information අතර වෙනස

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

“3”, “Math Puzzles”, “2” කියන්නේ වෙන වෙනම raw data. “Math Puzzles පොතේ පිටපත් 2ක් තිබෙනවා” කියන එක අපට තේරුමක් ඇති information එකක්.

දැනගත යුතු වැදගත් අදහස

Database එකේ data සුරකිමු. SQL query එකක් දමා ප්‍රශ්නයකට පිළිතුරක් සොයන විට information ලැබේ.

id	title	copies
1	Math Puzzles	2
2	Ocean World	1

✎ **ඔබම කර බලන්න**

දත්ත 3කින් නිවැරදි තොරතුරු වාක්‍යයක් ලියන්න.

★ නිවැරදි data නොතිබුණොත් report එකත් වැරදියි.

LESSON

Database නැතිව කුමක් වෙයිද?

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොත් ලබා දුන්නාට පස්සේ බාරදීම සටහන් නොකළොත් ඊළඟට පොත තිබෙනවා කියලා වැරදිව පෙන්වන්න පුළුවන්.

දැනගත යුතු වැදගත් අදහස

Database එකක් data පිළිවෙළකට තබාගෙන update සහ history සුරකින්න උදව් කරනවා. නමුත් application logic ද නිවැරදිව ලිවිය යුතුයි.



✎ **ඔබම කර බලන්න**

එක පොතක් දෙවරක් ලබාදී ඇත කියන ගැටලුව සිතා බලන්න.

★ එකම update එක transaction එකක් තුළ සිදු කිරීම ඉතා වැදගත්.

LESSON

Table කියන්නේ වගුවක්

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

Table එකක් කියන්නේ එකම වර්ගයට අදාළ records තබන තැනක්. `books` table එකේ පොත්; `members` table එකේ සාමාජිකයන්; `loans` table එකේ ලබාදීම් තබනවා.

දැනගත යුතු වැදගත් අදහස

මෙවැනි වෙන් කළ table කිහිපයක් සම්බන්ධ කරගත් එක relational database එකක්.

id	title	copies_available
1	Space Adventures	3
2	Little Python	2
3	Garden Secrets	4

✎ **ඔබම කර බලන්න**

වෙනම `teachers` table එකක් තිබුණොත් columns තුනක් යෝජනා කරන්න.

★ Table name ඉංග්‍රීසි කුඩා අකුරු වලින් තැබීම පහසුයි.

LESSON

Row සහ Column

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

වගුවේ ඉහළ සිට පහළ යන fields columns. එකම පොතක විස්තර පේළියක් row එකක් හෙවත් record එකක්.

දැනගත යුතු වැදගත් අදහස

`title` column එකේ පොත් නම්; `copies_available` column එකේ පිටපත් ගණන. එක් row එකක් එක පොතක්.

id	title	category
1	Space Adventures	Science
2	Little Python	Coding
3	Math Puzzles	Math

✎ **ඔබම කර බලන්න**

ඉහත table එකේ rows ගණන සහ columns ගණන කියන්න.

★ Rows = 3, Columns = 3; header row එක data record එකක් නොවේ.

LESSON

Field එකට නිවැරදි Data Type

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොතේ නම වචන නිසා TEXT. පිටපත් ගණන සම්පූර්ණ අංකයක් නිසා INTEGER. දිනයක් සරල ඉගෙනුම් project එකේ ISO text ලෙස තැබිය හැක.

දැනගත යුතු වැදගත් අදහස

Database එක type සහ rules මඟින් වැරදි දත්ත ඇතුළු වීම අඩු කරයි. SQLite flexible type system එකක් භාවිතා කරනවා.

column	type	example
id	INTEGER	1
title	TEXT	Ocean World
copies_total	INTEGER	3

✎ **ඔබම කර බලන්න**

මිල සඳහා float එකක් සෘජුවම භාවිතා කරනවාට වඩා cents සම්පූර්ණ අංකයකින් ගණනය කළ හැකි ඇයි?

★ මුදල් සඳහා floating point rounding ගැන ප්‍රවේශම් වන්න.

LESSON

Primary Key: එක් row එකකට එක් ID

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

දරුවන් දෙදෙනෙකුට එකම නම තිබිය හැකියි. ඒ නිසා නම වෙනුවට අද්විතීය ID එකක් භාවිතා කරනවා. `books.id` ඒ පොතේ Primary Key එකයි.

දැනගත යුතු වැදගත් අදහස

එම ID එකෙන් නිශ්චිත record එක තෝරන්න, වෙනත් table එකකින් සම්බන්ධ කරන්න පුළුවන්.

id (PK)	title
1	Math Puzzles
2	Math Puzzles
3	Ocean World

✎ **ඔබම කර බලන්න**

එකම title දෙකක් තිබෙනවා නම් ඒවා වෙන් කරන්නේ කොහොමද?

★ PK එක එක් record එකට වෙනම අගයක් දෙනවා.

LESSON

Foreign Key: tables අතර සම්බන්ධය

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

`loans` table එකේ `book_id` column එක book row එකකට යොමු කරයි. `member_id` column එක member row එකකට යොමු කරයි. ඒවා Foreign Keys.

දැනගත යුතු වැදගත් අදහස

එය “කවුද, මොන පොතද, කවදාද?” කියන ප්‍රශ්නය එකට විසඳන්න උපකාරී වේ.



✎ **ඔබම කර බලන්න**

නැති `book_id=999` එකක් loan එකකට දුන්නොත් මොකද වෙන්න ඕනේ?

★ SQLite සම්බන්ධතා enforce කිරීම සඳහා `PRAGMA foreign_keys = ON` යොදන්න.

LESSON

Database Management System (DBMS)

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

Database කියන්නේ data. DBMS කියන්නේ එය සුරකින්න, සොයන්න, සංස්කරණය කරන්න සහ ආරක්ෂා කරන්න භාවිතා කරන software එක.

දැනගත යුතු වැදගත් අදහස

SQLite, MySQL, PostgreSQL, SQL Server වැනි පද්ධති විවිධ අවශ්‍යතා සඳහා තිබෙනවා. අපි මුලින් SQLite භාවිතා කරමු.



✎ **ඔබම කර බලන්න**

SQLite database file එකක් සුරකින්නේ ඔබේ computer එකේ කොහේද?

★ SQLite සඳහා වෙනම server process එකක් අවශ්‍ය නැහැ.

LESSON

Spreadsheet එකකට වඩා database වෙනස

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

Excel හෝ CSV එකක rows/columns තිබුණත් database එකකට primary keys, constraints, relationships සහ transactions වැනි ශක්තිමත් features ඇත.

දැනගත යුතු වැදගත් අදහස

කුඩා ලැයිස්තුවකට spreadsheet හොඳයි. එකිනෙකට සම්බන්ධ data වෙනස් කරමින් වැඩ කරන app එකකට database වඩා ගැළපෙන්න පුළුවන්.

need	spreadsheet	database
Simple list	Yes	Yes
Relationships	Limited	Strong
Transactions	Limited	Yes

✎ **ඔබම කර බලන්න**

පොත් 5ක් පමණක් තිබුණොත් කුමක් පහසුද? පොත් 500ක් තිබුණොත්?

★ Tools දෙකම වටිනායි; නිවැරදි තේරීම use case එක අනුවයි.

LESSON

Relational සහ NoSQL

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

Relational databases table සහ rows අතර සම්බන්ධතා භාවිතා කරයි. සමහර NoSQL databases documents හෝ වෙනත් ආකාරවලින් data තබයි.

දැනගත යුතු වැදගත් අදහස

2026දී මේ දෙවර්ගයම වැදගත්. මුල් SQL පාඩමට සහ දරුවන්ගේ library app එකට SQLite සරල තේරීමක්.



✎ **ඔබම කර බලන්න**

සෑම අවස්ථාවකම එකම database වර්ගය හොඳම දේ නොවන හේතුවක් ලියන්න.

★ NoSQL කියන්නේ “SQL කිසිවක් නැහැ” කියන එකම අර්ථය නොවේ.

LESSON

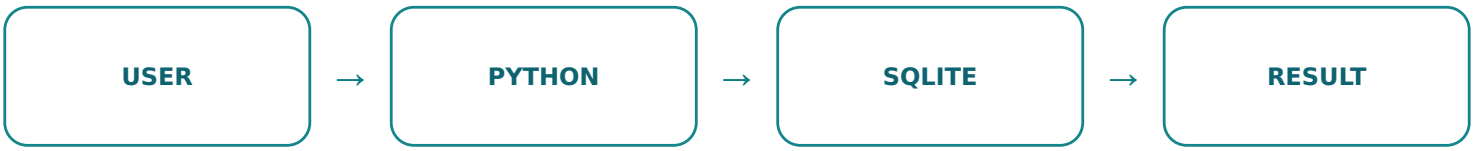
Database එකේ කුඩා ගමන

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

Application එකක ඔබ “Search” කියලා type කරනවා. Python code එක query එකක් database engine එකට යවයි. Engine එක matching rows ආපසු එවයි.

දැනගත යුතු වැදගත් අදහස

ඒ data CMD එකේ හෝ browser report එකක පෙන්වන්න පුළුවන්.



✎ **ඔබම කර බලන්න**

`Little Python` සොයන ගමනේ පියවර හතර ලියන්න.

★ Data query එකේ user input සැමවිටම safe parameters මඟින් යවන්න.

LESSON

අපේ Mini Project: Little Star Library

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොත්, සාමාජිකයන් සහ loans තුනක් භාවිතා කර පුංචි library එකක් පාලනය කරමු. CMD menu එකෙන් books add/search/borrow/return කරමු.

දැනගත යුතු වැදගත් අදහස

අවසානයේ CSV file එකක්, Chrome හි HTML report එකක් සහ backup database එකක් ලබාගනිමු.



✎ **ඔබම කර බලන්න**

ඔබ හදන library එකට අලුත් feature එකක් යෝජනා කරන්න.

★ මෙහි සියලු member names ඉගෙනීමට භාවිතා කරන කල්පිත නාමයි.

LESSON

පොතක් ලබාදෙන හැටි මුලින් කඩදාසියේ

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

අපේ library එකේ Space Adventures පිටපත් 3ක්. Alex ඒකෙන් එකක් borrow කළා. Available පිටපත් 2ක් වනවා.

දැනගත යුතු වැදගත් අදහස

Loans table එකේ කාටද, කවදාද, කුමන book_id එකද සුරැකිය යුතුයි. මෙය ඉදිරියේදී transaction එකක් වෙයි.



✎ **ඔබම කර බලන්න**

පොත ආපසු දුන්නාම available ගණන කීයද?

★ Total copies සහ Available copies එකම දේ නොවේ.

LESSON

2026 දී database ගැන මතක තබාගමු

01 · දත්ත ගබඩාව තේරුම් ගනිමු | පියවරෙන් පියවර ඉගෙනගනිමු

AI tools SQL ලියන්න උදව් කළත්, ඒ query එක නිවැරදිද, ආරක්ෂිතද සහ data privacy රකිනවාද කියලා මිනිසා පරීක්ෂා කළ යුතුයි.

දැනගත යුතු වැදගත් අදහස

Cloud databases, backups, encryption සහ access control වැනි දේ modern projects වල භාවිතා වේ. මෙහි අපි offline practice කරමු.



✎ **ඔබම කර බලන්න**

AI ලියා දුන් DELETE query එකක් එලෙසම run කරන්න හොඳ නැත්තේ ඇයි?

★ අධ්‍යාපනික database එකට සැබෑ දරුවන්ගේ පුද්ගලික දත්ත නොදමන්න.

PART 02

Python හා tools සූදානම් කරමු

අපේ ඉගෙනුම් පොතේ මීළඟ පියවර

CHAPTER 02

Python හා tools සූදානම් කරමු

CONCEPT → EXAMPLE → PRACTICE → CHECK

මෙම කොටස අවසන් කරද්දී වැදගත් වචන ඔබේම වචන වලින් පැහැදිලි කරන්න. Code එක copy කරන්නට පමණක් නොව එය කරන වැඩේ තේරුම් ගැනීම අරමුණයි.

✎ **පාඩමට කලින්**

මේ කොටස ගැන ඔබට දැන්ම දන්නා කරුණු දෙකක් කඩදාසියක ලියන්න. අවසානයේ ඒවා නැවත බලන්න.

LESSON

අවශ්‍ය software

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

Windows computer එකක්, Python 3.10 හෝ නවතම supported version එකක්, VS Code සහ CMD/Terminal අවශ්‍යයි. SQLite module එක සාමාන්‍ය Python installation එකේ ඇත.

දැනගත යුතු වැදගත් අදහස

වෙනම MySQL server හෝ pip package එකක් අවශ්‍ය නැති නිසා ඉගෙනීම සරලයි.



✎ **ඔබම කර බලන්න**

Install කරනවිට වැඩිහිටියෙකුගේ උපකාරය ලබාගන්න.

★ Official sites: python.org, code.visualstudio.com, sqlite.org.

LESSON

Python install කළාද බලමු

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

Windows හි Start menu එකෙන් CMD විවෘත කර `py --version` යොදන්න. Python version එක පෙන්වනවා නම් සූදානම්.

දැනගත යුතු වැදගත් අදහස

`py` command එක නැති පද්ධතියක `python --version` යොදා බලන්න.

■ Windows Command Prompt

```
C:\Users\Student> py --version
Python 3.x.x
C:\Users\Student> py -c "import sqlite3; print(sqlite3.sqlite_version)"
3.x.x
```

✎ **ඔබම කර බලන්න**

ඔබේ computer එකේ versions දෙක ලියාගන්න.

★ Versions ඔබේ installation එක අනුව වෙනස් විය හැක.

LESSON

VS Code එකෙන් folder එකක් open කරමු

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

ZIP source files extract කරලා `EGOTECHWORLD_Databases_Sinhala_2026_Project` folder එක VS Code තුළ Open Folder මඟින් විවෘත කරන්න.

දැනගත යුතු වැදගත් අදහස

Terminal → New Terminal තෝරන්න. Commands ධාවනය කරන්නේ `app.py` තිබෙන folder එකේද කියලා බලන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
project/
  app.py
  db_setup.py
  library_db.py
  sql/schema.sql
  examples/
  tests/
  outputs/
```

✎ **ඔබම කර බලන්න**

`README.md` කියවා project files හඳුනාගන්න.

★ Folder path එකේ user account name වෙනස් වෙයි.

LESSON

Folder structure එක කියවමු

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

`app.py` කියන්නේ CMD menu එක. `library_db.py` තුළ SQL functions. `sql/schema.sql` තුළ tables. `tests` තුළ automated QA.

දැනගත යුතු වැදගත් අදහස

`outputs` තුළ CSV, HTML සහ backup files. `little_star_library.db` project එක ධාවනය කරන විට සාදේ.

VS Code | SQL / Python

EXPLORER

```
little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
little-star-library/
  app.py
  library_db.py
  db_setup.py
  sql/
    schema.sql
  docs/
  examples/
  tests/
  outputs/
```

✎ **ඔබම කර බලන්න**

File එකේ වැඩ පැහැදිලි කිරීමක් එක් වාක්‍යයකින් කියන්න.

★ Files වෙන වෙනම තබන එක project එක තේරුම් ගැනීමට උදව් වෙයි.

LESSON

Python print() සමඟ පටන් ගනිමු

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

Python code එක screen එකට පෙන්වීමට `print()` භාවිතා කරනවා. අපේ database results ද අවසානයේ මේ ආකාරයට පෙන්වයි.

දැනගත යුතු වැදගත් අදහස

String එකක් quotes තුළ. Integer එකක් quotes නැති අංකයක්.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
print("Hello, Database!")
print("Available books:", 3)

# Expected output:
# Hello, Database!
# Available books: 3
```

✎ **ඔබම කර බලන්න**

`print("My Library")` යන code එක ලියා run කරන්න.

★ Code comments ඉංග්‍රීසියෙන් දමා තිබෙනවා.

LESSON

Variables කියන්නේ නම් තැබූ boxes

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොතක නම සහ ගණන වගේ අගයන් තබාගන්න Python variable භාවිතා කරනවා. පසුව query එකකට ඒ අගයන් parameters ලෙස යවමු.

දැනගත යුතු වැදගත් අදහස

`title` කියන්නේ variable name එක. `"Little Python"` කියන්නේ එහි value එක.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
title = "Little Python"
copies = 2

print(title)
print(copies + 1)
```

✎ **ඔබම කර බලන්න**

copies = 4 කළාම output එක කීයද?

★ Variable name එකට පැහැදිලි අදහසක් දෙන්න.

LESSON

List සහ Dictionary හඳුනාගනිමු

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

List එකක items කිහිපයක්. Dictionary එකක key-value යුගල. Database එකෙන් ලැබෙන book row එකක් Python dictionary එකක් ලෙස පෙන්වන්න පුළුවන්.

දැනගත යුතු වැදගත් අදහස

මේ දෙක වෙන්කර හඳුනාගෙන ඉදිරි පාඩම්වල rows තේරුම් ගන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
titles = ["Little Python", "Ocean World"]
book = {"id": 2, "title": "Little Python"}

print(titles[0])
print(book["title"])
```

✎ **ඔබම කර බලන්න**

අලුත් පොතක් list එකකට එකතු කර බලන්න.

★ List index ගණන් කිරීම 0 සිට ආරම්භ වේ.

LESSON

Function එකක් හදා බලමු

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

එකම වැඩ නැවත නැවත කිරීමට function එකක් හොඳයි. අපේ project එකේ `list_books()` සහ `borrow_book()` වගේ functions තිබෙනවා.

දැනගත යුතු වැදගත් අදහස

Python තුළ `def` භාවිතා කර function එකක් සාදනවා.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
def greet_reader(name):
    return "Hello, " + name

print(greet_reader("Alex"))
```

✎ **ඔබම කර බලන්න**

`Alex` වෙනුවට `Sam` යොදා බලන්න.

★ Indentation (code lines තුළ ඉඩ) Python වල අතිශයින් වැදගත්.

LESSON

sqlite3 module එක import කරමු

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

Python තුළ SQLite සමඟ කතා කරන්න `sqlite3` module එක import කරනවා. `connect()` සමඟ `.db` file එකක් විවෘත කරනවා.

දැනගත යුතු වැදගත් අදහස

File එක නොමැති නම් SQLite ඒ නමෙන් අලුත් database file එකක් සාදයි.



✎ **ඔබම කර බලන්න**

Run කර `hello.db` file එක ඇත්ද කියලා බලන්න.

★ Database file එක binary format එකක්; Notepad එකෙන් edit නොකරන්න.

LESSON

Temporary database එකක්

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

`:memory:` කියන special name එකෙන් memory තුළ පමණක් database එකක් සෑදේ. Program එක අවසන් වූ විට එය නැතිවෙයි.

දැනගත යුතු වැදගත් අදහස

එය පුහුණු අභ්‍යාස සහ tests සඳහා ප්‍රයෝජනවත්. පස්සේ සැබෑ `.db` file එක භාවිතා කරමු.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
import sqlite3

with sqlite3.connect(":memory:") as conn:
    conn.execute("CREATE TABLE colors(name TEXT)")
    conn.execute("INSERT INTO colors VALUES (?)", ("Blue",))
    print(conn.execute("SELECT * FROM colors").fetchall())
```

✎ **ඔබම කර බලන්න**

Memory database එක සඳහා වෙනම file එකක් ලැබෙනවාද?

★ `with conn` transaction එක commit/rollback කළත් connection close කිරීම වෙනම කළ යුතුයි.

LESSON

First example එක run කරමු

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

Source ZIP එකේ `examples/01_first_table.py` file එක open කරලා code එක කියවන්න. Table එකක් සෑදී row එකක් insert කරලා select කරයි.

දැනගත යුතු වැදගත් අදහස

මෙය වෙනස් කළත් ප්‍රධාන library database එකට බලපෑමක් නැහැ; in-memory example එකක්.

■ Windows Command Prompt

```
C:\little-star-library> py examples\01_first_table.py
[(1, 'Biscuit', 80)]
```

✎ **ඔබම කර බලන්න**

Biscuit වෙනුවට Apple කියලා වෙනස් කර result බලන්න.

★ Output එක tuple එකක් ඇති list එකක් ලෙස පෙන්වයි.

LESSON

Run කිරීමට ප්‍රධාන commands තුන

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

මුලින් `db_setup.py` database file සහ sample books සෑදේ. පසුව `app.py` menu එක විවෘත කරයි. අවසානයේ `unittest` වලින් test කරයි.

දැනගත යුතු වැදගත් අදහස

CLI කියන්නේ Command Line Interface; mouse click නැතිව text commands භාවිතා කිරීම.

■ Windows Command Prompt

```
C:\little-star-library> py db_setup.py
Created: ...little_star_library.db
C:\little-star-library> py app.py
Little Star Library - egotechworld.com
1 Books  2 Search  3 Add book ...
```

✎ **ඔබම කර බලන්න**

1 තෝරා books පෙන්වන්න. ඉන්පසු 0 තෝරා ඉවත් වන්න.

★ `python` command එකත් භාවිතා කළ හැකියි.

LESSON

DB Browser for SQLite (optional)

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

SQL tables කෙළින්ම screen එකකින් බැලීමට DB Browser for SQLite වැනි GUI tool එකක් උපකාරී වේ. එය අනිවාර්ය නැහැ.

දැනගත යුතු වැදගත් අදහස

ඉගෙනීමට Python code සහ CMD ප්‍රමාණවත්. GUI tool භාවිතා කළොත් database copy එකක පමණක් experiment කරන්න.



✎ **ඔබම කර බලන්න**

GUI tool එක විවෘත කර `books` table එකේ row තුනක් හඳුනාගන්න.

★ GUI app menus version අනුව වෙනස් විය හැක.

LESSON

Python error එකක් කියවමු

02 · Python හා tools සූදානම් කරමු | පියවරෙන් පියවර ඉගෙනගනිමු

`ModuleNotFoundError` කියන්නේ module එක හමු නොවීම. `SyntaxError` කියන්නේ code ලිවීමේ වරද. `sqlite3.OperationalError: no such table` කියන්නේ table එක නැති වීම.

දැනගත යුතු වැදගත් අදහස

Error message එකේ අන්තිම පේළිය බලා, file path එකත් පරීක්ෂා කරන්න.

■ Windows Command Prompt

```
C:\project> py app.py
# If a module is missing, check Python setup.
# If a table is missing, run: py db_setup.py
```

✎ **ඔබම කර බලන්න**

Intentional typo එකක් ලියා error එකක් බලා පසුව නිවැරදි කරන්න.

★ Error එකක් කියන්නේ ඉගෙනගැනීමේ clue එකක්.

PART 03

SQL මුල සිට කර බලමු

අපේ ඉගෙනුම් පොතේ මීළඟ පියවර

CHAPTER 03

SQL මුල සිට කර බලමු

CONCEPT → EXAMPLE → PRACTICE → CHECK

මෙම කොටස අවසන් කරද්දී වැදගත් වචන ඔබේම වචන වලින් පැහැදිලි කරන්න. Code එක copy කරන්නට පමණක් නොව එය කරන වැඩේ තේරුම් ගැනීම අරමුණයි.

✎ **පාඩමට කලින්**

මේ කොටස ගැන ඔබට දැන්ම දන්නා කරුණු දෙකක් කඩදාසියක ලියන්න. අවසානයේ ඒවා නැවත බලන්න.

LESSON

SQL කියන්නේ කුමක්ද?

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

SQL යනු Structured Query Language. Database එකට “අලුත් row එකක් එකතු කරන්න”, “මෙම පොත් පමණක් පෙන්වන්න” වැනි උපදෙස් දීමට භාවිතා කරන භාෂාවක්.

දැනගත යුතු වැදගත් අදහස

`SELECT` කියලා ලිව්වම database එක data කියවයි. SQL සාමාන්‍ය වචන වගේ පෙනෙනවා, නමුත් syntax රීති තිබෙනවා.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT title FROM books;

-- Ask the database for book titles.
```

✎ **ඔබම කර බලන්න**

SQL keywords capital ලියන එක අනිවාර්යද?

★ Uppercase කියන්නේ පාඩම කියවීමට පහසු convention එකක්.

LESSON

CREATE TABLE: අලුත් table එක

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

පළමු SQL statement එකෙන් `snacks` table එකක් හදමු. එහි id, name, price columns තිබෙනවා.

දැනගත යුතු වැදගත් අදහස

`PRIMARY KEY` නිසා id එක unique. `NOT NULL` නිසා name එක හිස්ව නොතිබිය යුතුයි.

VS Code | SQL / Python

EXPLORER

```
little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
CREATE TABLE snacks (
    id INTEGER PRIMARY KEY,
    name TEXT NOT NULL,
    price INTEGER NOT NULL
);
```

✎ **ඔබම කර බලන්න**

`price` වෙනුවට `stock` column එකක් නිර්මාණය කළොත් type එක කුමක්ද?

★ මේ table එක source හි `sql/first_steps.sql` තුළත් තිබෙනවා.

LESSON

SQLite file එකේ schema

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Schema කියන්නේ tables හා constraints වල සැලැස්ම. ඒකෙන් data සංවිධානය කොහොමද කියලා පෙන්වයි.

දැනගත යුතු වැදගත් අදහස

Project එකේ `sql/schema.sql` file එකේ books, members, loans tables නිර්වචනය කර තිබෙනවා.



✎ **ඔබම කර බලන්න**

Source ZIP එකේ schema file එක open කර table names තුන ලියන්න.

★ Tables තැනීම හා rows ඇතුළත් කිරීම වෙන වෙනම වැඩ.

LESSON

INSERT: අලුත් row එකක්

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොත් ලැයිස්තුවට row එකක් එක් කරන SQL keyword එක `INSERT`. Columns හා values ගැළපිය යුතුයි.

දැනගත යුතු වැදගත් අදහස

Python code තුළ user values යවද්දී `?` placeholders භාවිතා කළ යුතුයි. Static practice SQL එකෙහි literal values පෙන්වමු.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
INSERT INTO snacks (name, price)
VALUES ('Biscuit', 80);

SELECT * FROM snacks;
```

✎ **ඔබම කර බලන්න**

Biscuit row එකට `price` අගය 80 ලබාදෙන්න.

★ Data එක insert කළ පසුව `SELECT` එකෙන් බලා තහවුරු කරන්න.

LESSON

Python placeholder `?` එක

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Python value එක SQL string එකට අමු අමුවේ එකතු කිරීම අහිතකරයි. `?` placeholder හරහා values වෙනම bind කිරීම වඩා ආරක්ෂිතයි.

දැනගත යුතු වැදගත් අදහස

SQL injection වලින් වැළකීමේ ප්‍රධාන පියවරක් මෙයයි; නමුත් සම්පූර්ණ security system එකක් නොවේ.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
name = "Biscuit"
price = 80
conn.execute(
    "INSERT INTO snacks(name,price) VALUES(?,?)",
    (name, price)
)
```

✎ **ඔබම කර බලන්න**

`name` වෙනස් කර නැවත parameter values බැඳ දමන්න.

★ `?` placeholders values සඳහා; table names සඳහා නොවේ.

LESSON

SELECT: දත්ත කියවමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

`SELECT` මඟින් table එකකින් rows ලබාගන්න පුළුවන්. `*` කියන්නේ සියලු columns. විශාල table එකකදී අවශ්‍ය columns පමණක් ලබාගැනීම හොඳයි.

දැනගත යුතු වැදගත් අදහස

කොතැනකවත් change කරන්නේ නැතිව data කියවන්න පුළුවන් නිසා first practice එකට SELECT යෝග්‍යයි.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT * FROM books;
SELECT title, copies_available FROM books;
```

✎ **ඔබම කර බලන්න**

ලැබෙන columns දෙක වෙනස් කර author එකත් පෙන්වන්න.

★ SELECT කියන්නේ CRUD වල Read operation එකයි.

LESSON

WHERE: අවශ්‍ය rows පමණක්

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොත් සියල්ලක් නොව Science වර්ගයේ පොත් පමණක් අවශ්‍ය නම් `WHERE` භාවිතා කරන්න.

දැනගත යුතු වැදගත් අදහස

`=` කියන්නේ SQL equality comparison. Text value quotes තුළ.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT title, category FROM books
WHERE category = 'Science';
```

✎ **ඔබම කර බලන්න**

Category එක `Coding` කළොත් කුමන පොතක් ලැබේද?

★ SQL conditions ගැළපෙන rows පමණක් පෙන්වයි.

LESSON

Numbers compare කරමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

අවම වශයෙන් copies 2ක් තිබෙන පොත් සොයමු. SQL තුළ `>`, `<`, `>=`, `<=`, `!=` comparisons භාවිතා කළ හැක.

දැනගත යුතු වැදගත් අදහස

අපේ sample එකේ available count වෙනස් වුණොත් result එකත් වෙනස් වේ.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```sql
SELECT title, copies_available
FROM books
WHERE copies_available >= 2;
```

✎ **ඔබම කර බලන්න**

`>= 3` කරලා බලන්න. පොත් කීයක්ද?

★ Condition එක සපුරාලන්නේ නැති rows පෙන්වන්නේ නැහැ.

LESSON

ORDER BY: පිළිවෙළට සකසමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොත් නාම අකුරු අනුව පිළිවෙළට සකස් කිරීම `ORDER BY title ASC`. පිටපත් ගණන වැඩි සිට අඩුවට `DESC`.

දැනගත යුතු වැදගත් අදහස

`ORDER BY` නැති විට result එකේ order එකක් උපකල්පනය කරන්න එපා.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT title, copies_available FROM books
ORDER BY title ASC;

SELECT title, copies_available FROM books
ORDER BY copies_available DESC;
```

✎ **ඔබම කර බලන්න**

ABC පිළිවෙළ සහ reversed පිළිවෙළ අතර වෙනස කියන්න.

★ `ASC` = ascending, `DESC` = descending.

LESSON

LIMIT: මුල් කිහිපයක් පෙන්වමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොත් ලැයිස්තුව විශාල නම් මුල් rows 3ක් පමණක් පෙන්විය හැක.

දැනගත යුතු වැදගත් අදහස

Paging නිර්මාණය කිරීමට `LIMIT` සහ order එක වැදගත්.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT id, title FROM books
ORDER BY id
LIMIT 3;
```

✎ **ඔබම කර බලන්න**

LIMIT 2 කළාම rows කීයක් ලැබේද?

★ ප්‍රතිඵල consistent වීමට ORDER BY යොදන්න.

LESSON

LIKE: නමෙන් සොයමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

සම්පූර්ණ title එක මතක නැති නම් `LIKE` සහ `%` භාවිතා කර ගැළපෙන නමක් සොයන්න පුළුවන්. `%` කියන්නේ වෙනත් අකුරු 0ක් හෝ වැඩි ගණනක්.

දැනගත යුතු වැදගත් අදහස

`%Python%` කියන්නේ Python අඩංගු title සොයන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT id, title FROM books
WHERE title LIKE '%Python%';
```

✎ **ඔබම කර බලන්න**

`%World%` දාලා ලැබෙන පොත කුමක්ද?

★ User search සඳහා Python placeholders භාවිතා කරන්න.

LESSON

COUNT: ගණන් කරමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Database එකේ පොත් titles කීයක්ද කියලා `COUNT(*)` භාවිතා කරන්න. COUNT rows ගණන් කරනවා; physical copies ගණන් කරන එක නොවේ.

දැනගත යුතු වැදගත් අදහස

Titles 5ක් තිබුණත් books copies ගණන වෙනස් විය හැක.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT COUNT(*) AS title_count
FROM books;
```

✎ **ඔබම කර බලන්න**

Physical copies ගණන් කරන්නේ COUNT ද SUM ද?

★ `AS` භාවිතා කර report column එකට නමක් දිය හැක.

LESSON

SUM සහ AVG

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

පිටපත් එකතුව සොයාගැනීමට `SUM(copies_total)`. සාමාන්‍ය copies ගණන දැනගැනීමට `AVG(copies_total)`.

දැනගත යුතු වැදගත් අදහස

SUM සහ AVG NULL values සමඟ වෙනස් හැසිරීමක් පෙන්වන්න පුළුවන්; constraints නිසා අපේ count columns NULL නොවේ.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT SUM(copies_total) AS all_copies,
       AVG(copies_total) AS average_copies
FROM books;
```

✎ **ඔබම කර බලන්න**

අපේ මුල් sample එකේ total copies 12ද? එකතු කර තහවුරු කරන්න.

★ අපේ seed data එකේ 3+2+4+2+1 = 12.

LESSON

GROUP BY: කණ්ඩායම් කරමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Science, Coding, Math, Nature වැනි category අනුව පොත් ගණන් කිරීමට `GROUP BY` භාවිතා කරනවා.

දැනගත යුතු වැදගත් අදහස

විශාල database එකක reports සඳහා මෙය ඉතාම වැදගත්.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```sql
SELECT category, COUNT(*) AS titles
FROM books
GROUP BY category
ORDER BY category;
```

✎ **ඔබම කර බලන්න**

Science category එකට titles දෙකක් තිබෙනවාද?

★ GROUP BY එකේ nonaggregated columns ගැන SQL dialect rules වෙනස් විය හැක.

LESSON

DISTINCT: නැවතීම් නොමැති values

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොත් සියල්ලේ category එකෙන් වෙන වෙනම category names පමණක් ලබාගන්න `DISTINCT` භාවිතා කළ හැක.

දැනගත යුතු වැදගත් අදහස

එක category එකට titles කිහිපයක් තිබුණත් ලැයිස්තුවේ එම නම එක්වරක් පමණක් පෙන්වයි.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT DISTINCT category
FROM books
ORDER BY category;
```

✎ **ඔබම කර බලන්න**

`books` table එකේ unique category names කීයක්ද?

★ DISTINCT row එකේ තෝරාගත් සියලු columns මත ක්‍රියා කරයි.

LESSON

UPDATE: පවතින data වෙනස් කරමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

පොතක author නම වැරදි නම් `UPDATE` භාවිතා කළ හැක. `WHERE` නොමැති UPDATE එකකින් rows ගණනාවක් වෙනස් විය හැක.

දැනගත යුතු වැදගත් අදහස

මුලින් SELECT එකෙන් තෝරාගත් row හඳුනාගන්න; පසුව test database එකේ update කරමු.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
UPDATE books
SET author = 'A. New Author'
WHERE id = 2;

SELECT id, title, author FROM books WHERE id=2;
```

✎ **ඔබම කර බලන්න**

WHERE id=2 ඉවත් කළොත් ඇති අවදානම පැහැදිලි කරන්න.

★ ස්ථිර වෙනස්කම් කිරීමට පෙර backup එකක් තබන්න.

LESSON

DELETE: row එක මකමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

DELETE විශේෂයෙන් ප්‍රවේශමෙන් භාවිතා කරන්න. `WHERE` නොමැති DELETE එකකින් table එකේ සියලු rows මැකිය හැක.

දැනගත යුතු වැදගත් අදහස

අපේ lending system එකේ borrow history තිබෙන books මකා නොදැමීමට foreign key constraints උපකාරී වේ.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
-- Practice on a throwaway table only.
DELETE FROM snacks WHERE name = 'Juice';
SELECT * FROM snacks;
```

✎ **ඔබම කර බලන්න**

`DELETE FROM books;` copy-paste කර run නොකරන්න. ඒ වෙනුවට අවදානම කියන්න.

★ Database එකේ වැදගත් data යළි ලබාගන්න අමාරු විය හැක.

LESSON

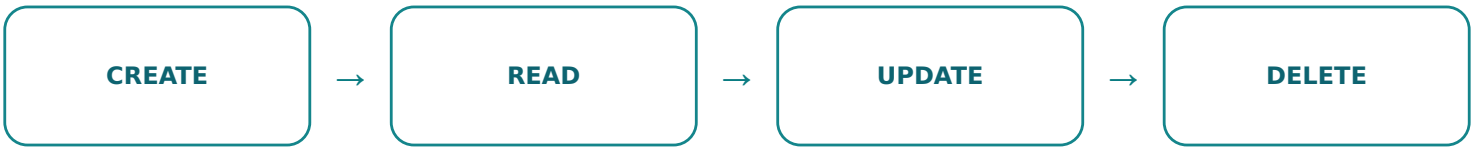
CRUD හතර මතක තබාගනිමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Create, Read, Update, Delete කියන මූලික database actions හතරට CRUD කියනවා. බොහෝ apps මේ හතර වෙනස් විදිහට භාවිතා කරනවා.

දැනගත යුතු වැදගත් අදහස

Library app එකේ Add book = Create; List/Search = Read; Return = Update loan; Delete operation හිතාමතා menu එකට එක්කරලා නැහැ.



✎ **ඔබම කර බලන්න**

Book name එකක් වෙනස් කරන්නේ CRUD වල කුමන action එකද?

★ DELETE තිබුණත් සෑම application එකටම user-facing delete feature අවශ්‍ය නැහැ.

LESSON

NULL කියන්නේ මොකක්ද?

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

`NULL` කියන්නේ value එකක් තවම නැති බවයි. Loan එකක් return කරලා නැත්නම් `returned_on` NULL.

දැනගත යුතු වැදගත් අදහස

SQL වල `= NULL` වෙනුවට `IS NULL` භාවිතා කරන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```sql
SELECT id, book_id, returned_on
FROM loans
WHERE returned_on IS NULL;
```

✎ **ඔබම කර බලන්න**

පොත ආපසු දුන් පසු `returned_on` තවදුරටත් NULL ද?

★ NULL = 0 හෝ හිස් text එක යන අර්ථය නොවේ.

LESSON

AND සහ OR

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Condition දෙකම සත්‍ය විය යුතු නම් AND. එක් condition එකක් සත්‍ය වුණත් හරි නම් OR.

දැනගත යුතු වැදගත් අදහස

`AND` සහ `OR` එකට ලියන විට brackets දා පැහැදිලි කරන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT title FROM books
WHERE category = 'Science'
  AND copies_available > 0;

SELECT title FROM books
WHERE category='Science' OR category='Coding';
```

✎ **ඔබම කර බලන්න**

Science හෝ Math ගෙන එන query එකක් ලියන්න.

★ Conditions තේරුම්ගැනීමට කඩදාසියේ True/False table එකක් අඳින්න.

LESSON

CHECK සහ NOT NULL

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

`CHECK` එකෙන් copies ඍණ අගයකට නොයන බවට rule එකක් දිය හැක. `NOT NULL` එකෙන් අත්‍යවශ්‍ය field හිස් value එකක් නොමැති බව පාලනය කරයි.

දැනගත යුතු වැදගත් අදහස

App එකේ validation සහ database rules දෙකම භාවිතා කළොත් අඩු mistakes.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
copies_available INTEGER NOT NULL
    CHECK(copies_available >= 0)

title TEXT NOT NULL
```

✎ **ඔබම කර බලන්න**

-1 copies එකක් save කිරීමට අවසර දිය යුතුද?

★ CHECK only validates a declared expression; real-world rules තවම app logic එකෙන් බලන්න.

LESSON

UNIQUE කියන්නේ නැවතීම් වැළැක්වීම

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Member email වගේ field එක unique විය යුතු අවස්ථා තිබෙනවා. නමුත් එකම title එක ඇති books දෙකක් තිබිය හැකි නිසා title එක unique කරන්න අවශ්‍ය නැහැ.

දැනගත යුතු වැදගත් අදහස

UNIQUE භාවිතා කිරීමට පෙර business rule එක හොඳින් අහන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
CREATE TABLE categories (
    id INTEGER PRIMARY KEY,
    name TEXT NOT NULL UNIQUE
);
```

✎ **ඔබම කර බලන්න**

Book title එකට UNIQUE දැමූ විට ඇති ගැටලුව කුමක්ද?

★ Unique rules භාවිතා කරන්නේ use case එක අනුවයි.

LESSON

JOIN: Tables දෙක තුනක් එකට

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Loan එකේ තිබෙන්නේ member_id=1 සහ book_id=2. මිනිසුන්ට අවශ්‍ය “Alex borrowed Little Python” වගේ result එකක්. JOIN මඟින් tables සම්බන්ධ කරමු.

දැනගත යුතු වැදගත් අදහස

Primary Key සහ Foreign Key values ගළපා තොරතුරු සොයාගත හැක.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT m.name, b.title
FROM loans l
JOIN members m ON m.id = l.member_id
JOIN books b ON b.id = l.book_id;
```

✎ **ඔබම කර බලන්න**

JOIN එකෙන් member name සහ book title දෙකම ලැබෙන්නේ ඇයි?

★ `JOIN` සාමාන්‍යයෙන් matching rows පමණක් ගනී.

LESSON

One-to-Many Relationship

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

එක member කෙනෙකුට කාලය තුළ loans ගණනාවක් තිබිය හැක. එක book එකකටත් loans කිහිපයක් තිබිය හැක.

දැනගත යුතු වැදගත් අදහස

එය “one-to-many” සම්බන්ධතාවක්. එක් loans row එකකට එක් member සහ එක් book එකක් සම්බන්ධයි.



✎ **ඔබම කර බලන්න**

ආපසු බාරදුන් පොත් history එක මකා නොදමන්නේ ඇයි?

★ History තබාගැනීම reports සඳහා ප්‍රයෝජනවත්.

LESSON

ERD කියන්නේ මොකක්ද?

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Entity Relationship Diagram එකක entities (tables), keys සහ relationships අඳිනවා. Code ලිවීමට පෙර team එකට data design ගැන එකඟ වෙන්න එය උදව් කරනවා.

දැනගත යුතු වැදගත් අදහස

Our mini ERD: BOOKS(id PK) — LOANS(book_id FK, member_id FK) — MEMBERS(id PK).



✎ **ඔබම කර බලන්න**

`members.name` කුමන table එකේ තැබිය යුතුද?

★ එකම member නම loans rows සියල්ලේ නැවත සුරකින්න අවශ්‍ය නැහැ.

LESSON

Indexes: search වේගවත් කරමු

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Book ID වගේ primary key columns index කරගැනීම සාමාන්‍යයි. විශාල tables වල සෙවීම් වේගවත් කිරීමට වෙනත් indexes එකතු කළ හැක.

දැනගත යුතු වැදගත් අදහස

Index එකක් search වේගවත් කළ හැකි නමුත් write/space cost එකක් තිබෙනවා. පොඩි project එකට වැඩි indexes අවශ්‍ය නැහැ.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
CREATE INDEX IF NOT EXISTS idx_loans_book
ON loans(book_id);
```

✎ **ඔබම කර බලන්න**

Index එකක් සෑම column එකටම දැමීම හොඳ තීරණයක් නොවන්නේ ඇයි?

★ Schema එකේ loans foreign-key columns සඳහා indexes තිබෙනවා.

LESSON

Transaction: වැඩ දෙක එකට

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Borrow කරන විට loans row එකක් insert කිරීමත් copies_available එක 1කින් අඩු කිරීමත් එකට සාර්ථක විය යුතුයි. එකක් පමණක් සිදුවෙලා නතර වුණොත් data අසමතුලිත වේ.

දැනගත යුතු වැදගත් අදහස

Transaction එකක් දමා වැඩ දෙකම සාර්ථක නම් commit, error නම් rollback කරමු.



✎ **ඔබම කර බලන්න**

Loan එක save වෙලා copies ගණන අඩු නොවුණොත් පෙනෙන ගැටලුව විස්තර කරන්න.

★ Transactions data consistency සඳහා අත්‍යවශ්‍යයි.

LESSON

COMMIT සහ ROLLBACK

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

COMMIT කියන්නේ transaction එකේ වෙනස්කම් සුරකින්න. ROLLBACK කියන්නේ commit නොකළ වෙනස්කම් අහෝසි කරන්න.

දැනගත යුතු වැදගත් අදහස

ඉගෙනීමට database copy එකක transaction අත්හදා බලන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
BEGIN;
UPDATE books SET copies_available = 0 WHERE id=1;
ROLLBACK;
-- The update was undone.
```

✎ **ඔබම කර බලන්න**

ROLLBACK වෙනුවට COMMIT යොදා තිබුණොත් වෙනස කුමක්ද?

★ Production data වල test නොකරන්න.

LESSON

SQL injection: අමු text වල අවදානම

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

User input එක SQL code එකට join කළොත් එම text query එකේ අර්ථය වෙනස් කරවීමට පුළුවන්.

දැනගත යුතු වැදගත් අදහස

Python sqlite3 `execute(sql, params)` මඟින් user-supplied values placeholders වෙත bind කරන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
keyword = input("Book name: ")
rows = conn.execute(
    "SELECT title FROM books WHERE title LIKE ?",
    ("%" + keyword + "%",)
).fetchall()
```

✎ **ඔබම කර බලන්න**

User type කරන SQL-like text එක code ලෙස ධාවනය නොවිය යුත්තේ ඇයි?

★ Always separate SQL syntax from untrusted values.

LESSON

Backup කියන්නේ දෙවැනි පිටපතක්

03 · SQL මුල සිට කර බලමු | පියවරෙන් පියවර ඉගෙනගනිමු

Computer crash වුණොත් database file නැති විය හැක. Backup copy එකක් වෙනම තැනක ආරක්ෂිතව තබාගන්න.

දැනගත යුතු වැදගත් අදහස

Live SQLite file එක copy-paste කිරීම වෙනුවට SQLite backup API එක භාවිතා කළාම consistent copy එකක් ලබාගත හැක.



✎ **ඔබම කර බලන්න**

ඔබේ test backup එක වෙනම folder එකක තබන්න.

★ Backup එක සාර්ථකද කියලා restore test එකකින් බලන්න.

PART 04

Little Star Library Project

අපේ ඉගෙනුම් පොතේ මීළඟ පියවර

CHAPTER 04

Little Star Library Project

CONCEPT → EXAMPLE → PRACTICE → CHECK

මෙම කොටස අවසන් කරද්දී වැදගත් වචන ඔබේම වචන වලින් පැහැදිලි කරන්න. Code එක copy කරන්නට පමණක් නොව එය කරන වැඩේ තේරුම් ගැනීම අරමුණයි.

✎ **පාඩමට කලින්**

මේ කොටස ගැන ඔබට දැන්ම දන්නා කරුණු දෙකක් කඩදාසියක ලියන්න. අවසානයේ ඒවා නැවත බලන්න.

LESSON

Requirements: app එකෙන් මොනවාද කළ යුත්තේ?

04 · Little Star Library Project | **පියවරෙන් පියවර ඉගෙනගනිමු**

Software ලියන්න කලින් project එකෙන් කළ යුතු වැඩ ලැයිස්තුවක් හදමු. පොත් list/search/add, member add, borrow, return, category report, backup.

දැනගත යුතු වැදගත් අදහස

මේවා SRS එකේ Functional Requirements. Source ZIP හි `docs/SRS.md` බලන්න.



✎ **ඔබම කර බලන්න**

FR01: List books යන්න කියවා ඔබම FR02 එකක් ලියන්න.

★ Requirements පැහැදිලි නම් testing පහසු වෙනවා.

LESSON

Mini SRS එකක වැදගත් කොටස්

04 · Little Star Library Project | **පියවරෙන් පියවර ඉගෙනගනිමු**

SRS කියන්නේ Software Requirements Specification. App එකෙන් කළ යුතු කාර්ය, පරිශීලකයා, input/output, non-functional qualities මෙහි ලියනවා.

දැනගත යුතු වැදගත් අදහස

අපේ project එක offline, Python standard library only, fictional data යන constraints තිබෙනවා.

ID	Requirement	Check
FR01	List books	Rows show
FR05	Borrow	Stock -1
NFR02	Safe values	Use ?

✎ **ඔබම කර බලන්න**

SRS එකේ requirements දෙකකට simple test එකක් ගළපන්න.

★ Documentation කියන්නේ අන් අයටත් project එක තේරුම්ගැනීමට දෙන උපකාරයයි.

LESSON

CMD menu එකේ Wireframe

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Design කියන්නේ screen layout එක සැලසුම් කිරීම. අපි button වෙනුවට CMD හි number menu එකක් භාවිතා කරනවා.

දැනගත යුතු වැදගත් අදහස

නවකයන්ට GUI coding වලට නොගිහින් database basics තේරුම් ගැනීමට මෙය හොඳයි.

```
■  Windows Command Prompt

Little Star Library - egotechworld.com
1 Books  2 Search  3 Add book  4 New member
5 Borrow 6 Return  7 Active loans
8 Categories  9 Export CSV
10 Backup  11 Chrome report  0 Exit
Choose: _
```

✎ **ඔබම කර බලන්න**

ඔබට වැදගත්ම menu options තුන තෝරන්න.

★ Screen previews ඉගෙනීම සඳහා නිර්මාණය කළ ආදර්ශ දර්ශන වේ.

LESSON

ERD: Books, Members, Loans

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Project එකේ database design එකේ table තුනක් තියෙනවා. Book හෝ member විස්තර loans table එකේ නැවත නැවත නොලියා IDs මඟින් යොමු කරනවා.

දැනගත යුතු වැදගත් අදහස

මෙය duplicate data අඩු කර, consistent information තබාගැනීමට උදව් කරයි.



✎ **ඔබම කර බලන්න**

Loan එකක borrower name නැති වුවත් name එක සොයාගත හැක්කේ කොහොමද?

★ JOIN query එකෙන් IDs අදාළ names වෙත map කරයි.

LESSON

Books table එක

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

`books` table එකේ title, author, category, copies_total, copies_available තිබෙනවා. copies_available එක මුල් total එකට වඩා වැඩි වෙන්න බැහැ.

දැනගත යුතු වැදගත් අදහස

Schema constraints මඟින් count එක negative හෝ total ඉක්මවා යාම වැළැක්වීමට උදව් කරයි.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```sql
CREATE TABLE IF NOT EXISTS books (
  id INTEGER PRIMARY KEY,
  title TEXT NOT NULL,
  author TEXT NOT NULL,
  category TEXT NOT NULL,
  copies_total INTEGER NOT NULL,
  copies_available INTEGER NOT NULL
);
```

✎ **ඔබම කර බලන්න**

සම්පූර්ණ CHECK rules සඳහා `sql/schema.sql` කියවන්න.

★ මේ code එක concept summary එකකි; ZIP හි complete schema එක භාවිතා කරන්න.

LESSON

Members table එක

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Members table එකේ member ID සහ name තියෙනවා. පුහුණුව සඳහා Alex, Sam, Robin වැනි fictional names තබමු.

දැනගත යුතු වැදගත් අදහස

Real users සඳහා privacy, consent, retention policy සහ permissions අවශ්‍යයි; මේ project එක ඒවා implement කරන්නේ නැහැ.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
CREATE TABLE IF NOT EXISTS members (
  id INTEGER PRIMARY KEY,
  name TEXT NOT NULL,
  joined_at TEXT NOT NULL
    DEFAULT CURRENT_TIMESTAMP
);
```

✎ **ඔබම කර බලන්න**

ඔබේ පුහුණු project එකේ සත්‍ය phone numbers දමන්න එපා.

★ Data minimization: අවශ්‍ය fields පමණක් සුරකින්න.

LESSON

Loans table එක

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Loan row එකක book ID, member ID, borrowed date සහ optional returned date තිබෙනවා. Loan එකට අද්විතීය id එකක් ඇත.

දැනගත යුතු වැදගත් අදහස

Book/member IDs foreign keys. Database connection එකේ foreign keys ON කරන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```sql
CREATE TABLE IF NOT EXISTS loans (
  id INTEGER PRIMARY KEY,
  book_id INTEGER NOT NULL,
  member_id INTEGER NOT NULL,
  borrowed_on TEXT NOT NULL,
  returned_on TEXT,
  FOREIGN KEY(book_id) REFERENCES books(id),
  FOREIGN KEY(member_id) REFERENCES members(id)
);
```

✎ **ඔබම කර බලන්න**

Return නොකළ loan එකක returned_on value කුමක්ද?

★ පිළිතුර NULL. Complete rules සඳහා source schema බලන්න.

LESSON

init_db(): database සකස් කිරීම

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

`db_setup.py` එකෙන් `init_db()` call කරනවා. මෙය schema file එක run කර tables සාදනවා; sample books තිබුණේ නැත්නම් seed rows එක්කරනවා.

දැනගත යුතු වැදගත් අදහස

`CREATE TABLE IF NOT EXISTS` නිසා app එක නැවත run කළාට tables නැවත සාදන්නේ නැහැ.



✎ **ඔබම කර බලන්න**

`py db_setup.py` දෙවරක් run කර duplicate sample rows ඇතිද බලන්න.

★ Project tests තුළ seed idempotency එක පරීක්ෂා කර තිබෙනවා.

LESSON

Sample data එක කියවමු

04 · Little Star Library Project | **පියවරෙන් පියවර ඉගෙනගනිමු**

Source එකේ පොත් 5ක් සහ fictional members 3ක් තිබෙනවා. පොත් categories: Science, Coding, Nature, Math.

දැනගත යුතු වැදගත් අදහස

මූලික sample එකේ total copies 12ක්. පසුව borrow කිරීමෙන් available count අඩු වේ.

id	title	available
1	Space Adventures	3
2	Little Python	2
3	Garden Secrets	4
4	Math Puzzles	2
5	Ocean World	1

✎ **ඔබම කර බලන්න**

Science category එකේ titles ගණන කීයද?

★ මේවා සත්‍ය පුස්තකාලයක නොව, ඉගෙනීම සඳහා නිර්මාණය කළ data.

LESSON

connect() function එක

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Source හි `connect()` function එක SQLite connection එක විවෘත කරන අතර foreign keys ON කරයි. වැඩ අවසන් වූ පසු connection එක close කිරීමට context manager එකක් යොදා තිබෙනවා.

දැනගත යුතු වැදගත් අදහස

Python + SQLite code එකක safe resource handling ඉගෙනගැනීමට හොඳ උදාහරණයක්.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```python
@contextmanager
def connect(db_path=DEFAULT_DB):
    conn = sqlite3.connect(str(db_path))
    try:
        conn.row_factory = sqlite3.Row
        conn.execute("PRAGMA foreign_keys=ON")
        with conn:
            yield conn
    finally:
        conn.close()
```

✎ **ඔබම කර බලන්න**

`finally` block එකේ කරන වැඩේ කුමක්ද?

★ මෙම snippet සඳහා sqlite3 සහ contextmanager imports අවශ්‍යයි.

LESSON

list_books(): rows පෙන්වීම

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

`list_books()` function එක SELECT query එකක් run කර Python dictionaries ලෙස rows ලබා දෙනවා. App menu එකේ option 1 ඒ result පෙන්වනවා.

දැනගත යුතු වැදගත් අදහස

Result එකේ id, title, author, category සහ copies counts තිබෙනවා.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
with connect(db_path) as conn:
    rows = conn.execute(
      "SELECT id,title,category,copies_available "
      "FROM books ORDER BY id"
    ).fetchall()
    books = [dict(row) for row in rows]
```

✎ **ඔබම කර බලන්න**

Option 1 run කර books 5ම පෙනෙනවාද බලන්න.

★ Complete code එක `library_db.py` තුළ ඇත.

LESSON

Search: title එකේ කොටසක්

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Option 2 යටතේ user type කරන keyword එකට `%` යොදලා LIKE search කරනවා. User value එක placeholder එකකින් bind කරනවා.

දැනගත යුතු වැදගත් අදහස

Search term එකෙන් matching titles පමණක් පෙන්වයි. SQL injection text එක code ලෙස interpret නොවේ.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
keyword = "Python"
rows = conn.execute(
  "SELECT title FROM books WHERE title LIKE ?",
  (f"%{keyword}%",)
).fetchall()
```

✎ **ඔබම කර බලන්න**

`World` search කළොත් කුමන පොතක්ද?

★ Source හි `list_books(keyword=...)` function එක වෙනස් නොකර run කළ හැක.

LESSON

Add Book: validate කර insert කරමු

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Option 3 යටතේ title, author, category, copies ගන්නවා. Title හෝ author හිස් නම් error. Copies සෘණ අගයක් නම් error.

දැනගත යුතු වැදගත් අදහස

Valid data පමණක් INSERT කිරීමෙන් database එක පිරිසිදු තබාගන්න පුළුවන්.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
book_id = add_book(
    "Star Stories", "T. Writer",
    "Fiction", 3
)
print("Saved ID:", book_id)
```

✎ **ඔබම කර බලන්න**

අලුත් පොතක් insert කර option 1 යටතේ එය පෙන්වෙනවාද බලන්න.

★ අලුත් row එකේ available = total = 3.

LESSON

Add Member: පළමු සාමාජිකයා

04 · Little Star Library Project | **පියවරෙන් පියවර ඉගෙනගනිමු**

Option 4 යටතේ member name එක ලබාදී `add_member()` function එක run කරනවා. App එක නැවත list members පෙන්වයි.

දැනගත යුතු වැදගත් අදහස

නව member එකට SQLite unique ID එකක් දෙනවා.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
member_id = add_member("Taylor")
print("New member ID:", member_id)
```

✎ **ඔබම කර බලන්න**

අලුත් fictional member කෙනෙක් එක්කර ලැබුණු ID එක ලියාගන්න.

★ සැබෑළමුන්ගේ names/addresses data එකට ඇතුළත් නොකරන්න.

LESSON

Borrow flow: මුලින් පරීක්ෂා කරමු

04 · Little Star Library Project | **පියවරෙන් පියවර ඉගෙනගනිමු**

User book ID සහ member ID දෙයි. Member ඉන්නවාද, book ඉන්නවාද, available > 0 ද කියලා බලන්න ඕන.

දැනගත යුතු වැදගත් අදහස

වැරදි ID එකක් දුන්නොත් loan එකක් save නොවිය යුතුයි.



✎ **ඔබම කර බලන්න**

Book ID=999 දුන්නොත් friendly error ලැබෙනවාද බලන්න.

★ Validation database constraints මඟින්ද සහාය ලබයි.

LESSON

Borrow flow: transaction එක

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Borrow කරන විට copies_available එක 1කින් අඩු කර loans row එකක් එක්කරනවා. `BEGIN IMMEDIATE` මඟින් අවශ්‍ය write transaction එක ආරම්භ කරනවා.

දැනගත යුතු වැදගත් අදහස

Error එකක් ඇත්නම් දෙකම rollback. සාර්ථක නම් commit.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
with connect(db_path) as conn:
    conn.execute("BEGIN IMMEDIATE")
    # Validate member and book here.
    conn.execute(
      "UPDATE books SET copies_available="
      "copies_available-1 WHERE id=?", (book_id,)
    )
    # Then INSERT the loan in same transaction.
```

✎ **ඔබම කර බලන්න**

මේ snippet එකේ validation lines නොපෙන්වන නිසා සම්පූර්ණ `borrow_book()` code එක කියවන්න.

★ Full function එක තුළ available check සහ affected-row check ඇත.

LESSON

Borrow demo: CMD එකෙන් අත්හදා බලමු

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

ප්‍රධාන menu එකේ option 5 තෝරන්න. Book ID=1 සහ Member ID=1. Output එකේ Loan ID එක ලැබේ.

දැනගත යුතු වැදගත් අදහස

පසුව option 1 run කර Space Adventures available copies 3 සිට 2 දක්වා අඩු වී තිබෙනවාද බලන්න.

```
■  Windows Command Prompt

Choose: 5
Book ID: 1
Member ID: 1
Borrowed successfully; Loan ID: 1
Choose: 1
1 | Space Adventures | available: 2
```

✎ **ඔබම කර බලන්න**

මෙය එක වතාවක් පමණක් run කර expected stock update එක නිරීක්ෂණය කරන්න.

★ Screenshot එක නිරූපණයක්; ඔබගේ data අනුව loan ID වෙනස් විය හැක.

LESSON

Return demo: පොත ආපසු

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Option 7 එකෙන් active loans පෙන්වන්න. Option 6 යටතේ අදාල Loan ID එක දෙන්න. App එක returned_on date එක save කර available copies 1කින් වැඩිකරයි.

දැනගත යුතු වැදගත් අදහස

එක loan එක twice return කිරීමට අවසර නැහැ.

■ Windows Command Prompt

```
Choose: 7
id: 1 | title: Space Adventures | name: Alex
Choose: 6
Loan ID to return: 1
Book returned successfully.
```

✎ **ඔබම කර බලන්න**

Option 1 යටතේ available 3 වෙලා තිබෙනවාද පරීක්ෂා කරන්න.

★ Return එකත් transaction එකක් තුළ සිදුවේ.

LESSON

Active loans: කවුද පොත තියාගෙන?

04 · Little Star Library Project | **පියවරෙන් පියවර ඉගෙනගනිමු**

Loan එකේ returned_on NULL නම් එය තවම active. Books සහ Members JOIN කරලා names එකට පෙන්වමු.

දැනගත යුතු වැදගත් අදහස

මෙම query එක report එකකටත් උපකාරී වේ.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT l.id, b.title, m.name
FROM loans AS l
JOIN books AS b ON b.id=l.book_id
JOIN members AS m ON m.id=l.member_id
WHERE l.returned_on IS NULL;
```

✎ **ඔබම කර බලන්න**

Loan එක return කළ පසු active loans ලැයිස්තුවෙන් ඉවත් වෙනවාද?

★ History table එකෙන් row එක delete නොකරනවා.

LESSON

Category report: කුමන පොත් වැඩිද?

04 · Little Star Library Project | **පියවරෙන් පියවර ඉගෙනගනිමු**

Books table එක category අනුව group කරලා titles, total copies, available copies පෙන්වන්න.

දැනගත යුතු වැදගත් අදහස

මේක library manager කෙනෙකුට නව පොත් මිලදී ගැනීම සැලසුම් කරන්න උදව් කරන්න පුළුවන්.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT category, COUNT(*) AS titles,
       SUM(copies_total) AS total,
       SUM(copies_available) AS available
FROM books GROUP BY category;
```

✎ ඔබම කර බලන්න

Science category එකේ copies total 4 ක් ඇති බව තහවුරු කරන්න.

★ Report එකේ title count සහ copies count වෙනස් විය හැක.

LESSON

CSV export: Spreadsheet එකට ගෙන යමු

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Option 9 මඟින් books.csv එකක් සෑදෙනවා. CSV කියන්නේ comma-separated values; Excel වැනි spreadsheet app එකකින් විවෘත කළ හැක.

දැනගත යුතු වැදගත් අදහස

Database file එක වෙනුවට report එක share කිරීමට CSV සුදුසු අවස්ථා තියෙනවා.

■ Windows Command Prompt

```
Choose: 9
CSV saved: ...outputs\books.csv

# CSV header:
id,title,author,category,copies_total,copies_available
```

✎ **ඔබම කර බලන්න**

CSV file එක spreadsheet program එකක විවෘත කර columns බලන්න.

★ CSV එකේ private data තිබුණොත් share කරන්න පෙර අවසර ලබාගන්න.

LESSON

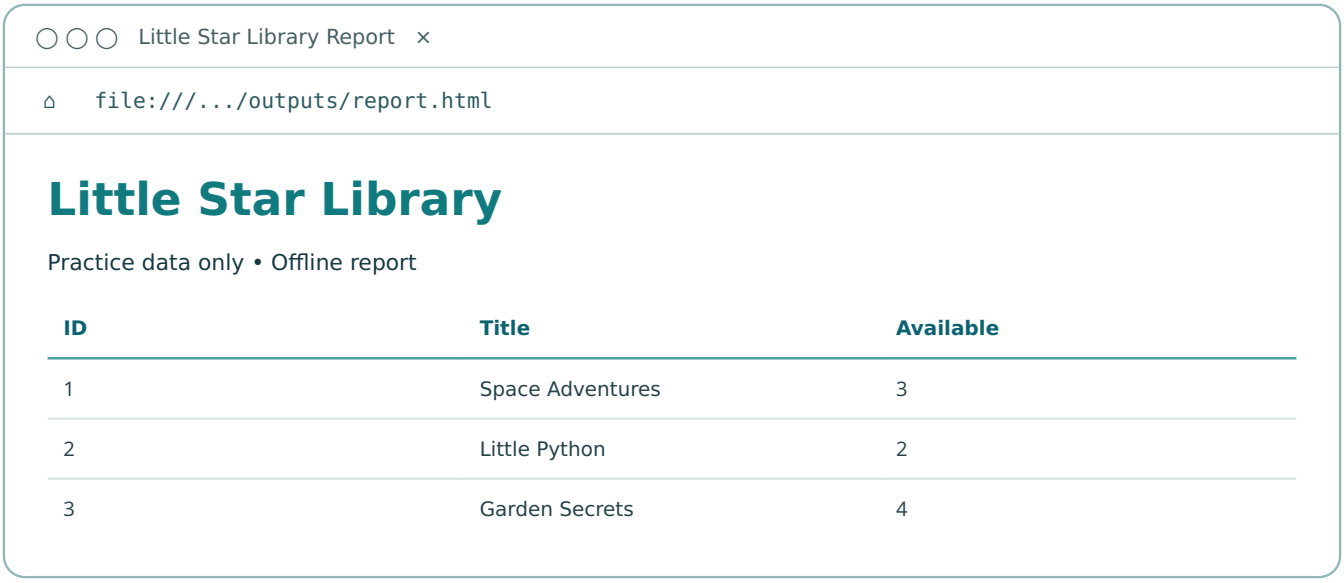
Chrome report: HTML page එකක්

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Option 11 මඟින් `outputs/report.html` generate වෙනවා. එය Chrome තුළ open කළාම Books table එක සහ category summary එක පෙනෙනවා.

දැනගත යුතු වැදගත් අදහස

HTML page එක offline file එකක්; public website එකක් නොවේ. Browser screenshot preview එක ඉගෙනීම සඳහා mockup එකක්.



✎ ඔබම කර බලන්න

Option 11 run කර browser report එක open කරන්න.

★ HTML දත්ත output වෙද්දී escape කිරීම වැදගත්; source code එය කරනවා.

LESSON

SQLite backup file එක

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

Option 10 භාවිතා කළොත් `outputs/library_backup.db` file එක සෑදෙනවා. App එක SQLite backup API භාවිතා කරනවා.

දැනගත යුතු වැදගත් අදහස

Backup එක වෙනම folder එකක තබා, අවශ්‍ය විට restore කර පරීක්ෂා කළ හැක.

■ Windows Command Prompt

```
Choose: 10
Backup saved: ...outputs\library_backup.db

# Check backup using a copy of the project.
```

✎ **ඔබම කර බලන්න**

Main DB එක delete කිරීමෙන් පරීක්ෂා නොකරන්න; backup copy එක වෙනම open කරන්න.

★ Backup කියන්නේ data protection වැඩපිළිවෙළක එක් කොටසක්.

LESSON

අලුත් records සුරැකෙන්නේ කොහේද?

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

`little_star_library.db` කියන file එකට data save වෙනවා. App එක නවතා නැවත run කළත් saved rows ඉතිරි වේ.

දැනගත යුතු වැදගත් අදහස

`:memory:` database එකේ data program close කළ විට නැති වුණත් file database එකේ ඉතිරි වෙන වෙනස දැන් පැහැදිලියි.



✎ **ඔබම කර බලන්න**

App එක නවතා reopen කර නව පොත තවම තියෙනවාද පරීක්ෂා කරන්න.

★ Database file එකට path එක වැරදි නම් වෙනම හිස් DB එකක් සාදන්න පුළුවන්.

LESSON

Program එක නැවත run කරමු

04 · Little Star Library Project | පියවරෙන් පියවර ඉගෙනගනිමු

`py app.py` කියලා නැවත run කිරීමෙන් menu එක පෙනෙයි. `init_db()` duplicate sample rows දමන්නේ හිස් table එකකට පමණයි.

දැනගත යුතු වැදගත් අදහස

Database state එක නිවැරදිව පරීක්ෂා කරන්න tests folder එකේ separate temporary DB files භාවිතා කරනවා.

```
■  Windows Command Prompt

C:\little-star-library> py app.py
Little Star Library - egotechworld.com
Choose: 1
... saved books are still here ...
```

✎ **ඔබම කර බලන්න**

Reload කිරීමෙන් data අහිමි නොවෙන බව තහවුරු කරන්න.

★ Source tests production database එක වෙනස් නොකරයි.

PART 05

Testing, QA සහ 2026 දැනුම

අපේ ඉගෙනුම් පොතේ මීළඟ පියවර

CHAPTER 05

Testing, QA සහ 2026 දැනුම

CONCEPT → EXAMPLE → PRACTICE → CHECK

මෙම කොටස අවසන් කරද්දී වැදගත් වචන ඔබේම වචන වලින් පැහැදිලි කරන්න. Code එක copy කරන්නට පමණක් නොව එය කරන වැඩේ තේරුම් ගැනීම අරමුණයි.

✎ **පාඩමට කලින්**

මේ කොටස ගැන ඔබට දැන්ම දන්නා කරුණු දෙකක් කඩදාසියක ලියන්න. අවසානයේ ඒවා නැවත බලන්න.

LESSON

Testing කියන්නේ මොකක්ද?

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Program එක හරියට වැඩ කරනවාද කියලා උදාහරණ inputs දාලා බලා තහවුරු කිරීම testing. Developer විතරක් නොව QA tester කෙනෙකුත් එය කරයි.

දැනගත යුතු වැදගත් අදහස

අපේ library project එකේ test cases functional requirements වලට සම්බන්ධයි.



✎ **ඔබම කර බලන්න**

“Borrow one book” සඳහා test input සහ expected stock change එක ලියන්න.

★ Expected සහ Actual අතර වෙනසක් නම් bug එකක් විය හැක.

LESSON

Test Case: ආකෘතියක්

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Test case එකක ID, steps, input, expected, actual, status තිබෙනවා. උදාහරණයක්: T04 borrow book.

දැනගත යුතු වැදගත් අදහස

සැබෑ app එකක් test කරද්දී expected outcome එක මුලින් ලියන එක වැදගත්.

ID	Action	Expected
T01	Set up DB	3 tables
T04	Borrow book	Stock -1
T07	Double return	Error

✎ **ඔබම කර බලන්න**

T04 සඳහා actual result එක ඔබම run කර සටහන් කරන්න.

★ Test pass වීමට outcome එක expected එකට ගැළපිය යුතුයි.

LESSON

Negative Testing: වැරදි input

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Blank title, missing ID, zero copies, duplicate return වැනි වැරදි cases පරීක්ෂා කරන්න. Real users හැමවිටම perfect inputs දෙන්නේ නැහැ.

දැනගත යුතු වැදගත් අදහස

App එක crash නොවී පැහැදිලි error message දීම හොඳ quality එකක්.

```
■  Windows Command Prompt

Choose: 5
Book ID: 999
Member ID: 1
Please check your input: Book does not exist.
```

✎ **ඔබම කර බලන්න**

Invalid book ID එකකින් loan record එකක් save නොවන බව confirm කරන්න.

★ Negative test එකක් PASS වීම කියන්නේ error එක නිවැරදිව handle වීමයි.

LESSON

Unit Testing: Python unittest

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Python standard library හි `unittest` module එකෙන් functions එකින් එක පරීක්ෂා කරන්න පුළුවන්. අපේ source folder එකේ tests/ ඇත.

දැනගත යුතු වැදගත් අදහස

Automated tests sample database එකට අත නොතබා temporary DB භාවිතා කරනවා.

■ Windows Command Prompt

```
C:\little-star-library> py -m unittest discover -s tests -v
... test_borrow_decrements_stock ... ok
... test_double_return_rejected ... ok
... test_foreign_keys_enforced ... ok

Ran 25 tests ... OK
```

✎ **ඔබම කර බලන්න**

ඔබේ computer එකේ tests pass වෙනවාද බලන්න.

★ Run count source version අනුව 25කි; future changes වලදී ගණන වෙනස් විය හැක.

LESSON

Bug Report එක ලියමු

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Bug එකක් හමු වුණොත් නැවත සිදු කරන steps ලියන්න. Expected සහ Actual එක වෙන්කර ලිවීම developer ට උදව් කරයි.

දැනගත යුතු වැදගත් අදහස

Example: T07 duplicate return. Expected: show error, stock unchanged. Actual: ඒ වෙනුවට stock +1 නම් bug.

Field	Example
ID	BUG-01
Steps	Return same loan twice
Expected	Error
Actual	Stock increased (bug)
Severity	High

✎ **ඔබම කර බලන්න**

මෙම hypothetical bug එකේ severity High කියන හේතුව කියන්න.

★ අපේ final code එක double return error වළක්වයි; මෙය bug-report උදාහරණයක්.

LESSON

QA සහ QC අතර වෙනස

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Quality Assurance (QA) හරියට වැඩ කිරීම සඳහා ක්‍රියාවලිය සැලසුම් කරයි. Quality Control (QC) නිෂ්පාදනයේ දෝෂ සොයා බැලීමේ පැත්තෙන් වැඩ කරයි.

දැනගත යුතු වැදගත් අදහස

Requirements review, test plan, coding standards, code reviews සහ regression tests quality වැඩි කරයි.



✎ **ඔබම කර බලන්න**

Release කරන්න කලින් QA checklist තුනක් ලියන්න.

★ Quality එක testing අවසානයේ පමණක් හදන දෙයක් නොවේ.

LESSON

Backup එකෙන් restore කරන පුහුණුව

05 · Testing, QA සහ 2026 දැනුම | **පියවරෙන් පියවර ඉගෙනගනිමු**

Backup එක තියෙනවා කියලාම ආරක්ෂිතයි කියන්න බැහැ. වෙනම practice copy එකකට restore කර rows කියවන්න.

දැනගත යුතු වැදගත් අදහස

අලුත් folder එකක් තබා `library_backup.db` copy කර SQLite සමඟ open කිරීමෙන් සත්‍යාපනය කළ හැක.



✎ **ඔබම කර බලන්න**

Backup copy එකේ Space Adventures row එක තියෙනවාද බලන්න.

★ දැනට run වන production DB file එක overwrite නොකරන්න.

LESSON

Data Privacy: දරුවන්ට විශේෂ වැදගත්

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

පොත්/සාමාජික practice data වලට සත්‍ය ළමයින්ගේ address, phone, email, passwords දමන්න එපා.

දැනගත යුතු වැදගත් අදහස

Real software එකක permissions, retention, deletion procedures, backups සහ encryption අවශ්‍ය විය හැක.



✎ **ඔබම කර බලන්න**

Public GitHub repo එකකට `.db` file එකක් upload නොකළ යුතු අවස්ථාවක් කියන්න.

★ පුද්ගලික තොරතුරු share කිරීමට පෙර අවසර ලබාගන්න.

LESSON

SQL Safety Checklist

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

User values safe placeholders; UPDATE/DELETE වල WHERE; input validation; FK constraints; transactions; backups; least-privilege access; tests.

දැනගත යුතු වැදගත් අදහස

මේ සෑම කරුණක්ම risk අඩු කරන ක්‍රමයක්. කිසිවක් තනිවම සම්පූර්ණ ආරක්ෂාවක් නොවේ.

Check	Why
? placeholders	Values safely bound
WHERE clause	Target rows
Foreign keys	Valid references
Transaction	All-or-nothing
Backup	Recover data

✎ **ඔබම කර බලන්න**

Code එකක raw SQL string concatenation එකක් හමු වුණොත් කුමක් කරන්නද?

★ Real apps සඳහා authentication, authorization, secure deployment ද අවශ්‍යයි.

LESSON

SQLite vs MySQL vs PostgreSQL

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

SQLite file database එකක් නිසා single-user practice/embedded apps සඳහා ඉතා සරලයි. MySQL සහ PostgreSQL වැනි server databases team apps සඳහා තෝරාගත හැක.

දැනගත යුතු වැදගත් අදහස

එකම SQL syntax බොහෝ තැන්වල පොදු වුණත් type rules සහ features වෙනස් විය හැක.

Tool	Typical fit
SQLite	Local/embedded
MySQL	Web apps
PostgreSQL	Complex server apps

✎ **ඔබම කර බලන්න**

Large online store එකක් සඳහා server DB එකක් අවශ්‍ය වෙන්නේ ඇයි?

★ Database තෝරන්නේ users, scale, security, operations අනුවයි.

LESSON

2026 Database Skills

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

2026දී SQL basics සමඟ privacy, secure backups, schema migration, cloud services, monitoring සහ AI-generated queries verify කිරීම ද වැදගත්.

දැනගත යුතු වැදගත් අදහස

මේ පොත foundational skills හදනවා. ඊළඟ පියවර MySQL/PHP web app එකක් හෝ PostgreSQL analytics project එකක් විය හැක.



✎ **ඔබම කර බලන්න**

ඔබට ඊළඟට ඉගෙනීමට කැමති database tool එකක් තෝරන්න.

★ Tools නිරන්තරයෙන් update වන නිසා official docs බලන්න.

LESSON

AI ලියා දුන් SQL එක පරීක්ෂා කරමු

05 · Testing, QA සහ 2026 දැනුම | **පියවරෙන් පියවර ඉගෙනගනිමු**

AI coding helper එකක් query එකක් යෝජනා කළාම tables/columns සත්‍යද, WHERE එක තියෙනවාද, එය data delete කරනවාද, sensitive data expose වෙනවාද බලන්න.

දැනගත යුතු වැදගත් අදහස

Test database එකක SELECT query එකකින් ආරම්භ කරන්න. AI output එක fact ලෙස සලකන්න එපා.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
-- Safer first step: preview target rows.
SELECT id, title FROM books WHERE id = 2;

-- Only after review consider an UPDATE.
```

✎ **ඔබම කර බලන්න**

ඔබට නොතේරෙන query එකක් run කිරීමට පෙර කළ යුතු දේවල් දෙකක් කියන්න.

★ AI නොව මිනිසා අවසාන වගකීම භාරගත යුතුයි.

LESSON

Schema migration කියන්නේ වෙනසක්

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

App එක වැඩෙන විට table එකකට අලුත් column එකක් අවශ්‍ය විය හැක. Existing rows නොනසා වෙනස් කිරීම schema migration.

දැනගත යුතු වැදගත් අදහස

Migration කරද්දී backup, migration script, rollback plan සහ tests වැදගත්.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
-- On a practice copy of the database:
ALTER TABLE books ADD COLUMN shelf TEXT;

SELECT id, title, shelf FROM books;
```

✎ **ඔබම කර බලන්න**

shelf column එකක් ඇතුළත් කළාම පරණ rows වල අගය කුමක්ද?

★ ALTER TABLE production data එකේ අහඹුවෙන් run නොකරන්න.

LESSON

Database හා Git

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

SQL schema, Python scripts, documentation සහ tests Git repository එකට ඇතුළත් කළ හැක. නමුත් සැබෑ private data ඇති `.db` files සාමාන්‍යයෙන් public repo වලට නොයැවිය යුතුයි.

දැනගත යුතු වැදගත් අදහස

`.gitignore` තුළ local database file paths දාන්න පුළුවන්.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
# Example .gitignore
*.db
__pycache__/
outputs/*.csv

# Commit source code, not private data.
```

✎ **ඔබම කර බලන්න**

කෙනෙක් පෞද්ගලික DB file එකක් GitHub වෙත push කළොත් ඇති අවදානම කියන්න.

★ Sample data පමණක් භාවිතා කළත් project policy අනුව repo content පරීක්ෂා කරන්න.

LESSON

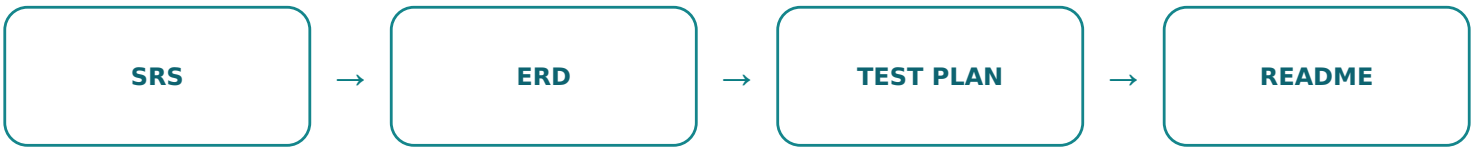
Documentation files කියවමු

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Source ZIP එකේ `docs/SRS.md`, `docs/ERD.md`, `docs/TEST_PLAN.md`, `docs/SQL_CHEATSHEET.md` තිබෙනවා.

දැනගත යුතු වැදගත් අදහස

Team එකේ වෙනත් කෙනෙකුට project එක තේරුම්ගෙන test කිරීමට documentation උදව් වෙයි.



✎ **ඔබම කර බලන්න**

SRS එකේ FR05 එක හා T04 test case එක සම්බන්ධ කරන්න.

★ Docs update නොකළොත් code එක හා documentation වෙනස් විය හැක.

LESSON

Release checklist

05 · Testing, QA සහ 2026 දැනුම | **පියවරෙන් පියවර ඉගෙනගනිමු**

Release කරන්න කලින් data model check, tests pass, sample workflow manual test, backup and restore, no private data, docs updated බව බලන්න.

දැනගත යුතු වැදගත් අදහස

මේ project එක learning/offline demo නිසා production-ready කියලා හඳුන්වන්නේ නැහැ.

Item	Status to check
Schema	Reviewed
Tests	25 pass
Borrow/return	Manual
Backup	Restored
Privacy	No real records

✎ **ඔබම කර බලන්න**

ඔබේ local run එකෙන් පසු සෑම item එකකටම tick දාන්න.

★ QA report එකක known limitations වෙනම ලියන්න.

LESSON

Troubleshooting: no such table

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

`no such table: books` error එකක් නම් setup කර නැති හෝ වෙනත් path එකක database file එකක් open වී ඇති විය හැක.

දැනගත යුතු වැදගත් අදහස

`py db_setup.py` run කර නිවැරදි folder එකෙන් `py app.py` run කරන්න.

■ Windows Command Prompt

```
sqlite3.OperationalError: no such table: books

Try: py db_setup.py
Check: project folder and .db file path
```

✎ **ඔබම කර බලන්න**

Error එකේ database file path එක source code එකෙන් හඳුනාගන්න.

★ `DEFAULT_DB` path එක `library_db.py` යටතේ project folder එකට anchored.

LESSON

Troubleshooting: database is locked

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

එකම SQLite file එකට writes කිහිපයක් එකවර උත්සාහ කරන විට database locked error දකින්න පුළුවන්.

දැනගත යුතු වැදගත් අදහස

Project එකේ busy timeout සහ short transactions තිබුණත් multi-user server app එකක් ලෙස හඳුන්වන්නේ නැහැ.



✎ **ඔබම කර බලන්න**

Lock error එකක් ලැබුණොත් database tool එකක open transaction තිබෙනවාද බලන්න.

★ Long-running transaction නොතබන්න.

LESSON

Troubleshooting: datatype error

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Integer field එකකට අකුරු ඇතුළත් කළාම conversion හෝ constraint error ලැබිය හැක. App එකේ number input validation අවශ්‍යයි.

දැනගත යුතු වැදගත් අදහස

Negative inputs ද test කරන්න. Database constraints පමණක් නොව UI validation ද තිබීම වටිනවා.

```
■  Windows Command Prompt

Choose: 3
Copies: abc
Please check your input: invalid literal ...

Try again with a whole number: 2
```

✎ **ඔබම කර බලන්න**

Copies = -1 දී app එක දෙන error message එක බලන්න.

★ Input errors සාමාන්‍යයි; clear messages ඉගෙනීමට උපකාරී වේ.

LESSON

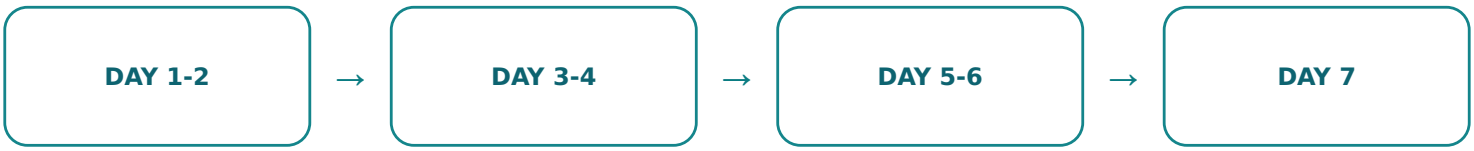
7-Day Learning Plan

05 · Testing, QA සහ 2026 දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

දින 1 data/table. දින 2 Python setup. දින 3 SELECT/INSERT. දින 4 WHERE/UPDATE/DELETE. දින 5 relationships/JOIN. දින 6 project borrow/return. දින 7 tests/reports.

දැනගත යුතු වැදගත් අදහස

දිනකට කුඩා project change එකක් සහ short recap එකක් කරන්න.



✎ **ඔබම කර බලන්න**

අද ඉගෙනගත් වචන 3ක් ඔබේම වචන වලින් පැහැදිලි කරන්න.

★ දින සැලැස්ම flexible; තේරුම් නොගත්තා නම් නැවත අත්හදා බලන්න.

PART 06

පරීක්ෂණය සහ අභ්‍යාස

අපේ ඉගෙනුම් පොතේ මීළඟ පියවර

CHAPTER 06

පරීක්ෂණය සහ අභ්‍යාස

CONCEPT → EXAMPLE → PRACTICE → CHECK

මෙම කොටස අවසන් කරද්දී වැදගත් වචන ඔබේම වචන වලින් පැහැදිලි කරන්න. Code එක copy කරන්නට පමණක් නොව එය කරන වැඩේ තේරුම් ගැනීම අරමුණයි.

✎ **පාඩමට කලින්**

මේ කොටස ගැන ඔබට දැන්ම දන්නා කරුණු දෙකක් කඩදාසියක ලියන්න. අවසානයේ ඒවා නැවත බලන්න.

QUIZ

MCQ 01 · Database Basics

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

Database එකක් කියන්නේ මොකක්ද?

A. Photo editor එකක්

B. Data ක්‍රමවත් සුරකින පද්ධතියක්

C. Game controller එකක්

D. Printer එකක්

✎ **ඔබම කර බලන්න**

Table / Row / Column යන වචනවලට ඔබේම උදාහරණයක් ලියන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

QUIZ

MCQ 02 · Primary Key

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

එකම නම ඇති books දෙකක් වෙන් කර හඳුනාගන්නේ?

A. Color

B. Title

C. Unique ID

D. Font size

✎ **ඔබම කර බලන්න**

PRIMARY KEY භාවිතා කරන අලුත් table එකක් ඇඳ බලන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

QUIZ

MCQ 03 · SELECT

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

Data කියවීමට භාවිතා කරන SQL keyword එක කුමක්ද?

A. SELECT

B. DELETE

C. DROP

D. COMMIT

✎ **ඔබම කර බලන්න**

SELECT title FROM books යන්න පැහැදිලි කරන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

QUIZ

MCQ 04 · WHERE

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

`WHERE id = 2` කියන්නේ?

A. සියලු rows තෝරාගැනීම

B. id 2ට ගැළපෙන rows තෝරාගැනීම

C. table මැකීම

D. database copy කිරීම

✎ **ඔබම කර බලන්න**

WHERE නොමැති UPDATE එකක අවදානම ලියන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

QUIZ

MCQ 05 · COUNT vs SUM

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

Copies එකතුව ගණනය කිරීමට වඩා සුදුසු එක?

A. ORDER BY

B. DISTINCT

C. SUM

D. DELETE

✎ **ඔබම කර බලන්න**

COUNT(*) සහ SUM(copies_total) අතර වෙනස ලියන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

QUIZ

MCQ 06 · Foreign Key

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

loans.book_id යනු කුමක්ද?

A. Title එකක්

B. Books table එකට reference එකක්

C. Password එකක්

D. Output file එකක්

✎ **ඔබම කර බලන්න**

ERD එකෙහි keys තුනම හඳුන්වන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

QUIZ

MCQ 07 · Transaction

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

Borrow කිරීමේ ක්‍රියා දෙක එකට සාර්ථක කළ යුත්තේ ඇයි?

A. Color වෙනස් කිරීමට

B. Keyboard shortcut එකක්

C. Database consistent තැබීමට

D. Print වේගවත් කිරීමට

✎ **ඔබම කර බලන්න**

Rollback වීමේ උදාහරණයක් ගෙනහැර දක්වන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

QUIZ

MCQ 08 · Safe Query

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

User input SQL ට ආරක්ෂිතව bind කරන SQLite feature එක?

A. ? placeholder

B. String concatenation

C. eval()

D. copy/paste

✎ **ඔබම කර බලන්න**

Parameter placeholders සහ data privacy අතර වෙනස කියන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

QUIZ

MCQ 09 · QA

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

Test case එකේ expected result කියන්නේ?

A. අවසානයේ අහඹු ප්‍රතිඵලය

B. කලින් අපේක්ෂා කළ ප්‍රතිඵලය

C. File size

D. Window title

✎ **ඔබම කර බලන්න**

Invalid book ID සඳහා negative test එකක් ලියන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

QUIZ

MCQ 10 · Backup

06 · පරීක්ෂණය සහ අභ්‍යාස | කුඩා ප්‍රශ්නයක් • ඔබේම පිළිතුර තෝරන්න

Backup එකක් ගැන නිවැරදිම අදහස කුමක්ද?

A. කිසිදා test නොකරන්න

B. Public කර share කරන්න

C. Restore කර verify කරන්න

D. Password file එකක්මයි

✎ **ඔබම කර බලන්න**

Backup file එක කියවා verify කරන පියවර දෙකක් ලියන්න.

★ පිළිතුරු අග කොටසේ ඇත. මුලින් ඔබම පිළිතුරු දෙන්න.

PART 07

Practical mini labs

අපේ ඉගෙනුම් පොතේ මීළඟ පියවර

CHAPTER 07

Practical mini labs

CONCEPT → EXAMPLE → PRACTICE → CHECK

මෙම කොටස අවසන් කරද්දී වැදගත් වචන ඔබේම වචන වලින් පැහැදිලි කරන්න. Code එක copy කරන්නට පමණක් නොව එය කරන වැඩේ තේරුම් ගැනීම අරමුණයි.

✎ **පාඩමට කලින්**

මේ කොටස ගැන ඔබට දැන්ම දන්නා කරුණු දෙකක් කඩදාසියක ලියන්න. අවසානයේ ඒවා නැවත බලන්න.

LAB

LAB 01 · Snacks table එකක්

07 · Practical mini labs | පියවරෙන් පියවර ඉගෙනගනිමු

Python එකේ `:memory:` භාවිතා කර snacks table එකක් සාදමු. දෙවැනි row එකට Juice සහ මිල 160 යොදන්න.

දැනගත යුතු වැදගත් අදහස

Table එකේ rows දෙකම පෙන්වන SQL query එක තනිවම ලියන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
CREATE TABLE snacks (
 id INTEGER PRIMARY KEY,
 name TEXT NOT NULL,
 price INTEGER NOT NULL
);
INSERT INTO snacks(name,price) VALUES('Biscuit',80);
INSERT INTO snacks(name,price) VALUES('Juice',160);
```

✎ **ඔබම කර බලන්න**

Juice price අගය 180 කරමින් UPDATE query එක ලියන්න.

★ Source හි `sql/first_steps.sql` උදාහරණය ඇත.

LAB

LAB 02 · CRUD තරඟය

07 · Practical mini labs | පියවරෙන් පියවර ඉගෙනගනිමු

Source project එකේ `examples/02_crud.py` run කරන්න. එය toys table එකක Ball row එක add, read, update, delete කරනවා.

දැනගත යුතු වැදගත් අදහස

එක් පියවරකින් පසු output බලන්න. UPDATE කළ විට Ball quantity 4 සිට 6 වෙයි.

■ Windows Command Prompt

```
C:\project> py examples\02_crud.py
READ [(1, 'Ball', 4)]
UPDATED [(1, 'Ball', 6)]
DELETED []
```

✎ **ඔබම කර බලන්න**

Ball වෙනුවට Toy Car යොදා බලන්න.

★ Temporary database නිසා ප්‍රධාන Library DB එක වෙනස් නොවෙයි.

LAB

LAB 03 · WHERE සහ ORDER BY

07 · Practical mini labs | පියවරෙන් පියවර ඉගෙනගනිමු

`examples/03_where_order.py` run කර `copies_available >= 2` books නම් අකුරු පිළිවෙළට පෙන්වන්න.

දැනගත යුතු වැදගත් අදහස

කොන්දේසිය `>= 3` කර results අතර වෙනස අත්හදා බලන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT title, copies_available
FROM books
WHERE copies_available >= 2
ORDER BY title ASC;
```

✎ **ඔබම කර බලන්න**

මේ SQL code එකේ filtering කොටස සහ sorting කොටස වෙන වෙනම හඳුන්වන්න.

★ Run කිරීමට කලින් py db_setup.py යොදන්න.

LAB

LAB 04 · JOIN පුහුණුව

07 · Practical mini labs | පියවරෙන් පියවර ඉගෙනගනිමු

`examples/04_join.py` borrow එකක් නැති database එකක පළමු loan එක හදලා name හා title JOIN කර පෙන්වයි.

දැනගත යුතු වැදගත් අදහස

JOIN query එකෙන් member name වෙනම table එකකින් ලබාගැනීමේ අගය පැහැදිලි කරන්න.

Windows Command Prompt

```
C:\project> py examples\04_join.py
Alex borrowed Space Adventures
```

✎ **ඔබම කර බලන්න**

app.py option 7 සමඟ output සසඳන්න.

★ Examples ප්‍රධාන practice DB එක වෙනස් කළ හැකි නිසා backup එකක් ගන්න.

LAB

LAB 05 · Reports & Groups

07 · Practical mini labs | පියවරෙන් පියවර ඉගෙනගනිමු

`examples/05_groups.py` file එකෙන් categories අනුව titles ගණන පෙන්වයි.

දැනගත යුතු වැදගත් අදහස

Science: 2; Coding: 1; Nature: 1; Math: 1 යන sample counts පරීක්ෂා කරන්න.

VS Code | SQL / Python

```
EXPLORER

little-star-library/
 app.py
 library_db.py
 db_setup.py
 sql/
 examples/
 tests/
```

```
SELECT category, COUNT(*) AS titles
FROM books
GROUP BY category
ORDER BY category;
```

✎ ඔබම කර බලන්න

මුළු titles ගණන 5 බව report එකෙන් පෙන්වන්න.

★ Add book කළොත් counts වෙනස් වීම සාමාන්‍යයි.

LAB

LAB 06 · CSV + Chrome

07 · Practical mini labs | පියවරෙන් පියවර ඉගෙනගනිමු

`examples/06_export.py` run කර `outputs/books.csv` සහ `outputs/report.html` හදන්න. CSV spreadsheet එකකින්, HTML browser එකකින් විවෘත කරන්න.

දැනගත යුතු වැදගත් අදහස

එකම database එකෙන් output formats දෙකක් ලබාගන්නා හැටි තේරුම් ගන්න.

■ Windows Command Prompt

```
C:\project> py examples\06_export.py
...outputs\books.csv
...outputs\report.html
```

✎ **ඔබම කර බලන්න**

CSV සහ HTML වල display styles වෙනස් වීම ගැන වාක්‍යයක් ලියන්න.

★ CSV සහ HTML එකම data source එකෙන් සෑදෙනවා.

LAB

LAB 07 · Borrow→Return story

07 · Practical mini labs | පියවරෙන් පියවර ඉගෙනගනිමු

Option 1 books බලන්න. Option 5 ගෙන එක පොතක් borrow කරන්න. Option 7 active loans බලන්න. Option 6 return කර option 7 නැවත බලන්න.

දැනගත යුතු වැදගත් අදහස

මෙය end-to-end workflow testing එකක්. `copies_available` මුල් අගයට යළි එන්න ඕනේ.



✎ **ඔබම කර බලන්න**

Borrow කළ loan ID එක සහ stock before/after අගයන් worksheet එකක ලියන්න.

★ Operation මැද error එකක් ආවොත් data inconsistent නොවිය යුතුයි.

LAB

LAB 08 · Wrong ID practice

07 · Practical mini labs | පියවරෙන් පියවර ඉගෙනගනිමු

Book ID=999 හෝ member ID=999 යොදා borrow උත්සාහ කරන්න. App එක friendly error පෙන්විය යුතුයි.

දැනගත යුතු වැදගත් අදහස

ඒකටත් පසු active loans ලැයිස්තුවේ අමතර loan එකක් නැති බව සොයන්න.

```
■  Windows Command Prompt

Choose: 5
Book ID: 999
Member ID: 1
Please check your input: Book does not exist.
```

✎ **ඔබම කර බලන්න**

Error එකෙන් පසුව DB rows ගණන වෙනස් වෙලාද බැලීමට test එකක් ලියන්න.

★ Negative testing ඉතා වැදගත් QA skill එකක්.

LAB

LAB 09 · Backup day

07 · Practical mini labs | පියවරෙන් පියවර ඉගෙනගනිමු

App එකේ 10 තෝරා backup file එකක් හදන්න. පසුව වෙනම temporary copy එකක් විවෘත කර books read කරන්න.

දැනගත යුතු වැදගත් අදහස

Backup එකක් තිබීම පමණක් නොව restore කළ හැකි බව විශ්වාසදායක ලෙස පරීක්ෂා කරන්න.



✎ **ඔබම කර බලන්න**

Backup හා live DB file එකේ sample book titles සසඳන්න.

★ Actual private data ඇති backups share නොකරන්න.

LAB

LAB 10 · Mini QA Report

07 · Practical mini labs | **පියවරෙන් පියවර ඉගෙනගනිමු**

පහත test results වලින් pass/fail ලියන්න: T01 setup, T04 borrow, T07 double return, T11 backup.

දැනගත යුතු වැදගත් අදහස

අපේ source tests 25ම PASS. ඔබේ computer එකේ local manual verification ද වෙනම කරන්න.

Test	Expected	Your result
T01	Tables exist	_____
T04	Stock -1	_____
T07	Friendly error	_____
T11	Backup opens	_____

✎ **ඔබම කර බලන්න**

`docs/TEST_PLAN.md` සමඟ results ගළපන්න.

★ Test evidence ලෙස screenshots/command outputs save කරන්න.

PART 08

ඉදිරියට ගෙන යන දැනුම

අපේ ඉගෙනුම් පොතේ මීළඟ පියවර

CHAPTER 08

ඉදිරියට ගෙන යන දැනුම

CONCEPT → EXAMPLE → PRACTICE → CHECK

මෙම කොටස අවසන් කරද්දී වැදගත් වචන ඔබේම වචන වලින් පැහැදිලි කරන්න. Code එක copy කරන්නට පමණක් නොව එය කරන වැඩේ තේරුම් ගැනීම අරමුණයි.

✎ **පාඩමට කලින්**

මේ කොටස ගැන ඔබට දැන්ම දන්නා කරුණු දෙකක් කඩදාසියක ලියන්න. අවසානයේ ඒවා නැවත බලන්න.

LESSON

SQL Quick Reference

08 · ඉදිරියට ගෙන යන දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

SQL command එකක් දකින විට keyword එකෙන් එහි අරමුණ හඳුනාගන්න. CREATE schema, INSERT add, SELECT read, UPDATE edit, DELETE remove.

දැනගත යුතු වැදගත් අදහස

WHERE filter, ORDER BY sort, GROUP BY summarize, JOIN combine.

SQL	Use
SELECT	Read rows
INSERT	Add row
UPDATE	Edit row
DELETE	Remove row
JOIN	Combine tables
GROUP BY	Report

✎ **ඔබම කර බලන්න**

මේ පිටුව bookmark කර තබන්න.

★ Safe values සඳහා ? placeholders භාවිතා කරන්න.

LESSON

Glossary: වචන 10ක්

08 · ඉදිරියට ගෙන යන දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Database, table, row, column, primary key, foreign key, SQL, query, schema, transaction යන වචන එක් එක් වාක්‍යයකින් පැහැදිලි කර බලන්න.

දැනගත යුතු වැදගත් අදහස

අභ්‍යාස කරද්දී වචනය කියවා ඔබේම කඩදාසියේ කුඩා රූපයක් අඳින්න.



✎ **ඔබම කර බලන්න**

සෑම වචනයකටම පොත් පුස්තකාලයෙන් උදාහරණයක් ලියන්න.

★ තේරුම් ගැනීම rote memorization ට වඩා වටිනායි.

LESSON

උපකාරී නිල මාර්ගෝපදේශ

08 · ඉදිරියට ගෙන යන දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

Python sqlite3 module, SQLite SQL language, SQLite foreign keys, SQLite backup documentation කියවීමෙන් වැඩි විස්තර ගන්න පුළුවන්.

දැනගත යුතු වැදගත් අදහස

මෙම tutorial එක 2026දීත් භාවිතා කළ හැකි මූලික දැනුමට අවධානය දෙයි. Versions, menus වෙනස් විය හැක.

> https://docs.python.org/3/library/sqlite3.html
> https://www.sqlite.org/lang.html
> https://www.sqlite.org/foreignkeys.html
> https://www.sqlite.org/backup.html

✎ **ඔබම කර බලන්න**
SQL query එකක් නොතේරුණාම නිල docs එකක අදාළ keyword එක සොයන්න.

★ 2026 ඔක්තෝබර් දිනවලදී පිළිගත් නිල මූලාශ්‍ර මත ප්‍රධාන තාක්ෂණික සංකල්ප පරීක්ෂා කළා.

LESSON

Quiz Answer Key

08 · ඉදිරියට ගෙන යන දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

MCQ වලට ඔබම පිළිතුරු දුන් පසු මේ key එක සමඟ සසඳන්න. වැරදි ප්‍රශ්න ඇතිනම් ඒ පාඩම් නැවත කියවා run කරන්න.

දැනගත යුතු වැදගත් අදහස

පිළිතුරු: 01-B, 02-C, 03-A, 04-B, 05-C, 06-B, 07-C, 08-A, 09-B, 10-C.

MCQ	Answer	Topic
01	B	Database
02	C	PK
03	A	SELECT
04	B	WHERE
05	C	SUM
06	B	FK
07	C	Transaction
08	A	Placeholder
09	B	QA
10	C	Backup

✎ **ඔබම කර බලන්න**

ඔබට වැරදුණු මාතෘකා තුනක් තෝරා යළි lab කරන්න.

★ Mistakes are part of learning.

LESSON

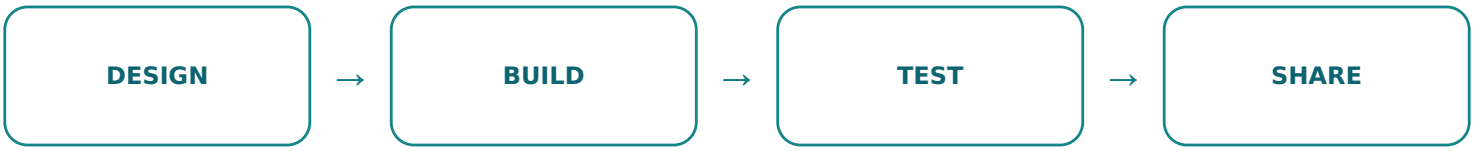
Certificate / Final Challenge

08 · ඉදිරියට ගෙන යන දැනුම | පියවරෙන් පියවර ඉගෙනගනිමු

ඔබේ Mini Library project එක නව කෙනෙකුට පෙන්වන්න. Books add, search, borrow, return, report, backup යන ක්‍රියාවලිය complete කර පෙන්වන්න.

දැනගත යුතු වැදගත් අදහස

Documentation සහ test results සුරකින්න. තවත් challenge එකක්: shelf number field එකක් database schema එකට එක්කර app එක update කර බලන්න.



✎ **ඔබම කර බලන්න**

ඔබගේ final project screenshots සහ notes එක්කර mini portfolio එකක් හදන්න.

★ මෙය අධ්‍යාපනික self-check certificate activity එකක් මිස පිළිගත් accreditation එකක් නොවේ.

SOURCE CODE APPENDIX

Code Appendix

Project ZIP තුළ සම්පූර්ණ runnable source code එක ඇත

SOURCE

SQL + Python examples

READ EACH FUNCTION • FIND THE SQL • RUN THE APP

ඉදිරි පිටුවල project source files වලින් තෝරාගත් code කොටස් තිබෙනවා. ඒවා තනිවම run කිරීමට වඩා ZIP source එකේ complete files run කරන්න. මේ කොටස code reading practice සඳහායි.

CODE REFERENCE

Source 01 · First tiny SQLite example

File: examples/01_first_table.py • Lines 1-8

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | examples/01_first_table.py

```
1 """First database: make one table, add a record, and read it."""
2 import sqlite3
3
4 with sqlite3.connect(":memory:") as conn:
5     conn.execute("CREATE TABLE snacks (id INTEGER PRIMARY KEY, name TEXT, price INTEGER)")
6     conn.execute("INSERT INTO snacks (name, price) VALUES (?, ?)", ("Biscuit", 80))
7     rows = conn.execute("SELECT id, name, price FROM snacks").fetchall()
8     print(rows)
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 02 · CRUD example

File: examples/02_crud.py • Lines 1-11

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | examples/02_crud.py

```
"""Create, Read, Update, Delete with a temporary database."""
import sqlite3

with sqlite3.connect(":memory:") as conn:
    conn.execute("CREATE TABLE toys(id INTEGER PRIMARY KEY, name TEXT, qty INTEGER)")
    conn.execute("INSERT INTO toys(name,qty) VALUES(?,?)", ("Ball", 4))  # Create
    print("READ", conn.execute("SELECT * FROM toys").fetchall())
    conn.execute("UPDATE toys SET qty=? WHERE name=?", (6, "Ball"))       # Update
    print("UPDATED", conn.execute("SELECT * FROM toys").fetchall())
    conn.execute("DELETE FROM toys WHERE name=?", ("Ball",))              # Delete
    print("DELETED", conn.execute("SELECT * FROM toys").fetchall())
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 03 · Database schema / Books

File: sql/schema.sql • Lines 1-15

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | sql/schema.sql

```
1 -- Little Star Library: three related tables.
2 CREATE TABLE IF NOT EXISTS books (
3     id INTEGER PRIMARY KEY,
4     title TEXT NOT NULL,
5     author TEXT NOT NULL,
6     category TEXT NOT NULL DEFAULT 'General',
7     copies_total INTEGER NOT NULL DEFAULT 1 CHECK(copies_total >= 0),
8     copies_available INTEGER NOT NULL DEFAULT 1
9         CHECK(copies_available >= 0 AND copies_available <= copies_total),
10    created_at TEXT NOT NULL DEFAULT CURRENT_TIMESTAMP
11 );
12
13 CREATE TABLE IF NOT EXISTS members (
14    id INTEGER PRIMARY KEY,
15    name TEXT NOT NULL,
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 04 · Database schema / Members & Loans

File: sql/schema.sql • Lines 16-30

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | sql/schema.sql

```
    joined_at TEXT NOT NULL DEFAULT CURRENT_TIMESTAMP
);

CREATE TABLE IF NOT EXISTS loans (
    id INTEGER PRIMARY KEY,
    book_id INTEGER NOT NULL,
    member_id INTEGER NOT NULL,
    borrowed_on TEXT NOT NULL,
    returned_on TEXT,
    FOREIGN KEY (book_id) REFERENCES books(id) ON DELETE RESTRICT,
    FOREIGN KEY (member_id) REFERENCES members(id) ON DELETE RESTRICT,
    CHECK(returned_on IS NULL OR returned_on >= borrowed_on)
);
CREATE INDEX IF NOT EXISTS idx_loans_book ON loans(book_id);
CREATE INDEX IF NOT EXISTS idx_loans_member ON loans(member_id);
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 05 · library_db.py imports & config

File: library_db.py • Lines 1-37

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | library_db.py

```
"""Safe, beginner-friendly SQLite operations for Little Star Library.

Python standard library only. Data in all examples is fictitious.
"""
from __future__ import annotations

import csv
from contextlib import contextmanager
from datetime import date
from html import escape
from pathlib import Path
import sqlite3
from typing import Any

ROOT = Path(__file__).resolve().parent
DEFAULT_DB = ROOT / "little_star_library.db"
SCHEMA = ROOT / "sql" / "schema.sql"

class LibraryError(ValueError):
    """Friendly error message for an invalid library action."""


@contextmanager
def connect(db_path: str | Path = DEFAULT_DB):
    """Open, commit/rollback, and always close the SQLite connection."""
    conn = sqlite3.connect(str(db_path))
    try:
        conn.row_factory = sqlite3.Row
        conn.execute("PRAGMA foreign_keys = ON")
        conn.execute("PRAGMA busy_timeout = 5000")
        with conn:
            yield conn
    finally:
        conn.close()


def init_db(db_path: str | Path = DEFAULT_DB, seed: bool = True) -> None:
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 06 · list_books() & SQL search

File: library_db.py • Lines 61-71

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | library_db.py

```
61 def list_books(db_path: str | Path = DEFAULT_DB, keyword: str = "") -> list[dict[str, Any]]:
62     """Search titles using a placeholder; never build SQL with user input."""
63     with connect(db_path) as conn:
64         rows = conn.execute(
65             """SELECT id, title, author, category, copies_total, copies_available
66                FROM books WHERE title LIKE ? ORDER BY id""",
67             (f"%{keyword.strip()}%",),
68         ).fetchall()
69         return [dict(row) for row in rows]
70
71
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 07 · Add book and member

File: library_db.py • Lines 72-100

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | library_db.py

```
 72 def add_book(title: str, author: str, category: str, copies: int,
 73              db_path: str | Path = DEFAULT_DB) -> int:
 74     title, author = title.strip(), author.strip()
 75     if not title or not author:
 76         raise LibraryError("Book title and author are required.")
 77     if not isinstance(copies, int) or copies < 0:
 78         raise LibraryError("Copies must be a whole number of zero or more.")
 79     with connect(db_path) as conn:
 80         cur = conn.execute(
 81             """INSERT INTO books(title, author, category, copies_total, copies_available)
 82                VALUES(?, ?, ?, ?, ?)""",
 83             (title, author, category.strip() or "General", copies, copies),
 84         )
 85         return int(cur.lastrowid)
 86
 87
 88 def add_member(name: str, db_path: str | Path = DEFAULT_DB) -> int:
 89     if not name.strip():
 90         raise LibraryError("Member name cannot be empty.")
 91     with connect(db_path) as conn:
 92         cur = conn.execute("INSERT INTO members(name) VALUES(?)", (name.strip(),))
 93         return int(cur.lastrowid)
 94
 95
 96 def list_members(db_path: str | Path = DEFAULT_DB) -> list[dict[str, Any]]:
 97     with connect(db_path) as conn:
 98         return [dict(r) for r in conn.execute("SELECT id, name FROM members ORDER BY id")]
 99
100
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 08 · Borrow logic (part 1)

File: library_db.py • Lines 101-119

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | library_db.py

```
101 def borrow_book(book_id: int, member_id: int,
102                 db_path: str | Path = DEFAULT_DB,
103                 on_date: str | None = None) -> int:
104     """Borrow and decrement copies in one atomic transaction."""
105     day = on_date or date.today().isoformat()
106     with connect(db_path) as conn:
107         conn.execute("BEGIN IMMEDIATE")
108         member = conn.execute("SELECT id FROM members WHERE id=?", (member_id,)).fetchone()
109         if member is None:
110             raise LibraryError("Member does not exist.")
111         book = conn.execute(
112             "SELECT copies_available FROM books WHERE id=?", (book_id,)
113         ).fetchone()
114         if book is None:
115             raise LibraryError("Book does not exist.")
116         if book["copies_available"] < 1:
117             raise LibraryError("No available copies; try another book.")
118         changed = conn.execute(
119             """UPDATE books SET copies_available=copies_available-1
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 09 · Borrow logic (part 2)

File: library_db.py • Lines 120-130

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | library_db.py

```
120                 WHERE id=? AND copies_available>0""", (book_id,)
121         )
122         if changed.rowcount != 1:
123             raise LibraryError("This book is no longer available.")
124         cur = conn.execute(
125             "INSERT INTO loans(book_id, member_id, borrowed_on) VALUES(?, ?, ?)",
126             (book_id, member_id, day),
127         )
128         return int(cur.lastrowid)
129
130
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 10 · Return logic

File: library_db.py • Lines 131-152

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | library_db.py

```
131 def return_book(loan_id: int, db_path: str | Path = DEFAULT_DB,
132                 on_date: str | None = None) -> None:
133     """Mark loan returned and restore one copy together."""
134     day = on_date or date.today().isoformat()
135     with connect(db_path) as conn:
136         conn.execute("BEGIN IMMEDIATE")
137         loan = conn.execute(
138             "SELECT book_id, borrowed_on, returned_on FROM loans WHERE id=?", (loan_id,)
139         ).fetchone()
140         if loan is None:
141             raise LibraryError("Loan does not exist.")
142         if loan["returned_on"] is not None:
143             raise LibraryError("This loan was already returned.")
144         if day < loan["borrowed_on"]:
145             raise LibraryError("Return date cannot be before borrow date.")
146         conn.execute("UPDATE loans SET returned_on=? WHERE id=?", (day, loan_id))
147         conn.execute(
148             "UPDATE books SET copies_available=copies_available+1 WHERE id=?",
149             (loan["book_id"],),
150         )
151
152
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 11 · JOIN and category report

File: library_db.py • Lines 153-175

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | library_db.py

```
153 def list_loans(db_path: str | Path = DEFAULT_DB,
154                active_only: bool = False) -> list[dict[str, Any]]:
155     clause = "WHERE l.returned_on IS NULL" if active_only else ""
156     with connect(db_path) as conn:
157         return [dict(r) for r in conn.execute(f"""
158             SELECT l.id, b.title, m.name, l.borrowed_on, l.returned_on
159             FROM loans l
160             JOIN books b ON b.id = l.book_id
161             JOIN members m ON m.id = l.member_id
162             {clause} ORDER BY l.id
163         """)]
164
165
166 def category_report(db_path: str | Path = DEFAULT_DB) -> list[dict[str, Any]]:
167     with connect(db_path) as conn:
168         return [dict(r) for r in conn.execute("""
169             SELECT category, COUNT(*) AS book_titles,
170                    SUM(copies_total) AS total_copies,
171                    SUM(copies_available) AS available_copies
172             FROM books GROUP BY category ORDER BY category
173         """)]
174
175
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 12 · CSV & backup helpers

File: library_db.py • Lines 176-202

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | library_db.py

```
176 def export_csv(output_path: str | Path, db_path: str | Path = DEFAULT_DB) -> Path:
177     out = Path(output_path)
178     out.parent.mkdir(parents=True, exist_ok=True)
179     with out.open("w", newline="", encoding="utf-8-sig") as file:
180         writer = csv.DictWriter(file, fieldnames=[
181             "id", "title", "author", "category", "copies_total", "copies_available"
182         ])
183         writer.writeheader()
184         writer.writerows(list_books(db_path))
185     return out
186
187
188 def backup_database(output_path: str | Path,
189                     db_path: str | Path = DEFAULT_DB) -> Path:
190     """Use the SQLite backup API instead of copying a possibly active file."""
191     out = Path(output_path)
192     out.parent.mkdir(parents=True, exist_ok=True)
193     with connect(db_path) as source:
194         destination = sqlite3.connect(str(out))
195         try:
196             with destination:
197                 source.backup(destination)
198         finally:
199             destination.close()
200     return out
201
202
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 13 · HTML report helper

File: library_db.py • Lines 203-233

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | library_db.py

```
203 def export_html(output_path: str | Path,
204                db_path: str | Path = DEFAULT_DB) -> Path:
205     """Create an offline report that can be opened in Chrome."""
206     out = Path(output_path)
207     out.parent.mkdir(parents=True, exist_ok=True)
208     books = list_books(db_path)
209     category = category_report(db_path)
210     rows = "\n".join(
211         "<tr>" + "".join(f"<td>{escape(str(b[key]))}</td>" for key in
212         ["id", "title", "category", "copies_total", "copies_available"]) + "</tr>"
213         for b in books
214     )
215     summary = " ".join(
216         f"<li>{escape(str(x['category']))}: {x['available_copies']} available</li>"
217         for x in category
218     )
219     doc = f"""<!doctype html><html lang="en"><head><meta charset="utf-8">
220     <meta name="viewport" content="width=device-width,initial-scale=1">
221     <title>Little Star Library Report</title><style>
222     body{{font:16px Arial,sans-serif;margin:32px auto;max-width:850px;padding:0 20px;
223     background:#fff;color:#183d45}}h1{{color:#087a7d}}
224     table{{border-collapse:collapse;width:100%}}th,td{{padding:10px;border:1px solid
#9fbab8;text-align:left}}
225     th{{background:#fff}}footer{{margin-top:30px;border-top:1px solid #9fbab8;padding-top:10px}}
226     </style></head><body><h1>Little Star Library</h1>
227     <p>Practice data only. Report generated from SQLite.</p>
228     <table><thead><tr><th>ID</th><th>Title</th><th>Category</th>
229     <th>Total</th><th>Available</th></tr></thead><tbody>{rows}</tbody></table>
230     <h2>Available copies by category</h2><ul>{summary}</ul>
231     <footer>egotechworld.com | Learning report</footer></body></html>"""
232     out.write_text(doc, encoding="utf-8")
233     return out
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 14 · CMD helpers

File: app.py • Lines 1-27

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | app.py

```
"""Windows CMD menu for the Little Star Library database."""
from pathlib import Path
from library_db import (
    LibraryError, init_db, list_books, add_book, add_member, list_members,
    borrow_book, return_book, list_loans, category_report,
    export_csv, backup_database, export_html,
)

ROOT = Path(__file__).resolve().parent


def print_rows(items):
    if not items:
        print("No records found.")
    for item in items:
        print(" | ".join(f"{key}: {value}" for key, value in item.items()))


def number(label):
    """Get a positive ID using a simple beginner input check."""
    value = int(input(label).strip())
    if value < 1:
        raise ValueError("Please use a positive ID.")
    return value


def run():
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 15 · CMD menu (part 1)

File: app.py • Lines 28-54

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | app.py

```
28  init_db()
29  print("Little Star Library - egotechworld.com")
30  while True:
31      print("\n1 Books  2 Search  3 Add book  4 New member")
32      print("5 Borrow 6 Return  7 Active loans  8 Categories")
33      print("9 Export CSV  10 Backup  11 Chrome report  0 Exit")
34      choice = input("Choose: ").strip()
35      try:
36          if choice == "1":
37              print_rows(list_books())
38          elif choice == "2":
39              print_rows(list_books(keyword=input("Search title: ")))
40          elif choice == "3":
41              title = input("Title: ")
42              author = input("Author: ")
43              category = input("Category: ")
44              copies = int(input("Copies: "))
45              print("Saved book ID:", add_book(title, author, category, copies))
46          elif choice == "4":
47              print_rows(list_members())
48              print("New member ID:", add_member(input("Name: ")))
49          elif choice == "5":
50              print_rows(list_members())
51              print_rows(list_books())
52              loan_id = borrow_book(number("Book ID: "), number("Member ID: "))
53              print("Borrowed successfully; Loan ID:", loan_id)
54          elif choice == "6":
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

CODE REFERENCE

Source 16 · CMD menu (part 2)

File: app.py • Lines 55-78

Code එක කියවන විට SQL commands හඳුනාගන්න. Functions, parameters සහ error handling එකට සම්බන්ධ වන විදිහ නිරීක්ෂණය කරන්න.

VS Code | app.py

```
55                print_rows(list_loans(active_only=True))
56                return_book(number("Loan ID to return: "))
57                print("Book returned successfully.")
58            elif choice == "7":
59                print_rows(list_loans(active_only=True))
60            elif choice == "8":
61                print_rows(category_report())
62            elif choice == "9":
63                print("CSV saved:", export_csv(ROOT / "outputs" / "books.csv"))
64            elif choice == "10":
65                print("Backup saved:", backup_database(ROOT / "outputs" / "library_backup.db"))
66            elif choice == "11":
67                print("Open this in Chrome:", export_html(ROOT / "outputs" / "report.html"))
68            elif choice == "0":
69                print("Bye! Keep learning databases.")
70                break
71            else:
72                print("Choose a number from 0 to 11.")
73        except (LibraryError, ValueError) as error:
74            print("Please check your input:", error)
75
76
77 if __name__ == "__main__":
78     run()
```

✎ **Code Reading Challenge**

මේ code කොටස තුළ `SELECT`, `INSERT`, `UPDATE`, `?`, function definitions හෝ validations අඩංගුද? ඒවාට උදාහරණයක් ලියන්න.

★ සම්පූර්ණ runnable version එක සමඟ source ZIP ලබාදී ඇත. Code segments අතර imports සහ surrounding logic තිබිය හැක.

FINAL CHECKLIST

ඔබගේ Database Project එක සාර්ථකද?

දැනගත් දේවල් අදහස් කර බලන්න • 2026

සාර්ථකව කළ යුතු වැඩ	ඔබේ Tick
Python හා SQLite run කළා	☐
Books table කියවා rows/columns හඳුනාගත්තා	☐
SELECT / INSERT / WHERE queries ලිව්වා	☐
PK / FK හා JOIN තේරුම්ගත්තා	☐
Book එකක් borrow සහ return කළා	☐
CSV සහ HTML report සෑදුවා	☐
Database backup එකක් verify කළා	☐
Unit tests run කළා	☐
Real personal data භාවිතා නොකළා	☐

මේ checklist එක සම්පූර්ණ කිරීමෙන් පසු තවත් කුඩා project එකක් සිතාගන්න: Class timetable, Toy inventory, Plant watering log වැනි දෙයක්.

NEXT STEP

MySQL හෝ PostgreSQL ඉගෙනගෙන, ඒ database එක PHP / Python / Java backend සමඟ සම්බන්ධ කරන විදිහ අවබෝධ කරගන්න.