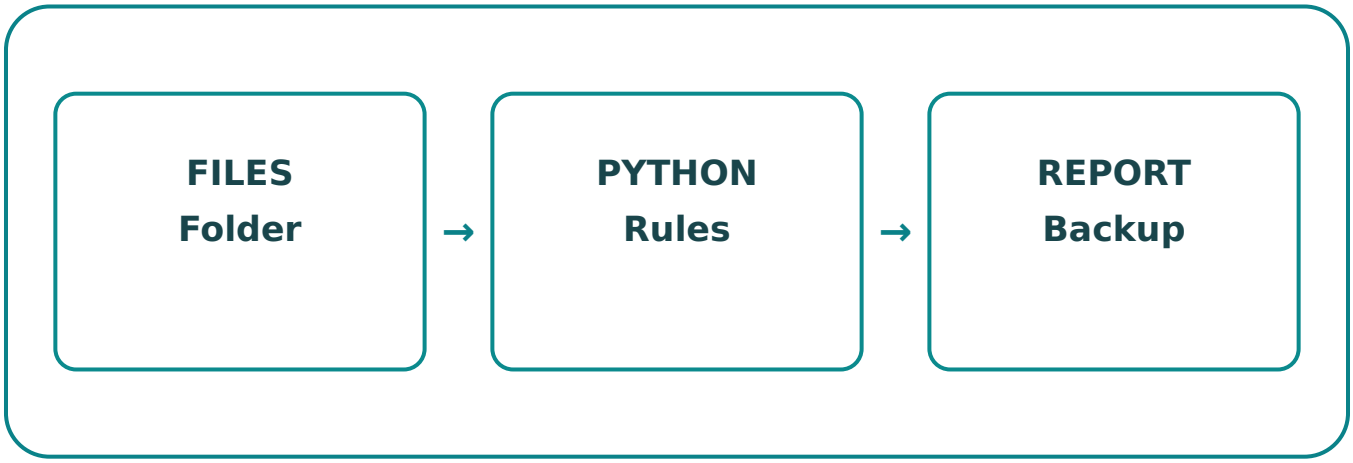
සිංහල පියවරෙන් පියවර • 2026

Python Automation ළමයින්ටත් පහසුවෙන්

CMD එකෙන් files වර්ග කරමු · CSV reports හදමු · ZIP backups ගනිමු



Python basics වල සිට සැබෑ mini automation project එක දක්වා. සෑම පාඩමකම අත්හදා බැලිය හැකි කේත, diagrams සහ පුංචි අභ්‍යාස.

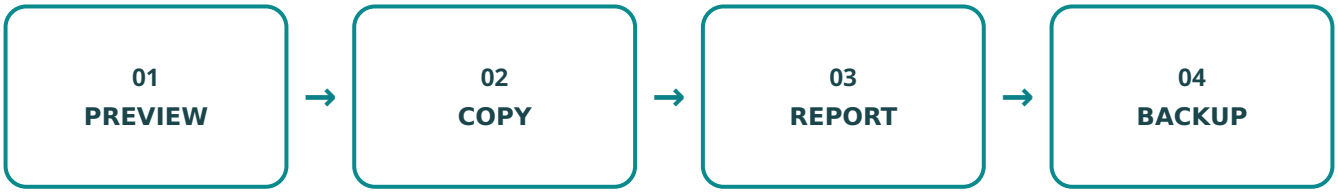
PURE WHITE PAGES | VS CODE | BLACK CMD | SOURCE ZIP | QA + SDLC

EGOTECHWORLD learning series • 2026 edition

PROJECT OVERVIEW

අපි හදන්නේ මොකක්ද?

Little Star Automation Center නම් Python CMD app එකක්. ගුරුවරයාට හෝ ආරම්භක developer කෙනෙකුට sample files හතරක් හරි folders වලට copy කරගන්න, sample sales spreadsheet එකෙන් report එකක් සාදාගන්න, ZIP backup එකක් හදාගන්න පුළුවන්.



Input

demo_inbox/
data/sales.csv

Output

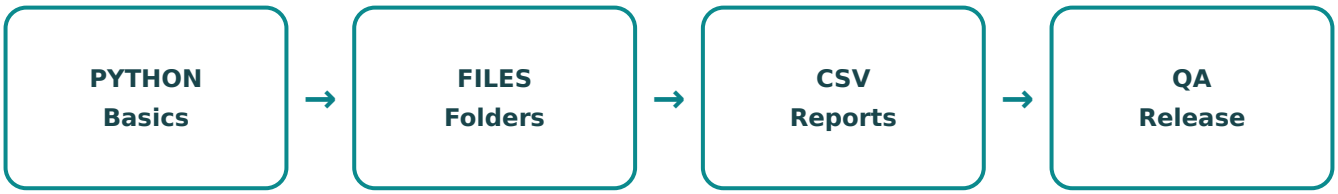
output/organized/
output/reports/
backups/

ආරක්ෂාව

මුල් files delete හෝ move නොකරන learning project එකක්. පුද්ගලික data නොයොදන්න. “Preview” command එකෙන් අරඹන්න.

LEARNING JOURNEY

ඔබේ ඉගෙනුම් මාර්ගය



මෙම පොත පාඩම් කුඩා කොටස් වලට බෙදනවා. එක පිටුවක concept එකක් ඉගෙනගෙන code එක run කරන්න. ඊළඟ පිටුවට යන්නට පෙර ඔබේ මිතුරෙකුට තේරුම්කර දීමට උත්සාහ කරන්න.

අවශ්‍ය දේ: Windows PC · Python 3.12+ · VS Code · CMD · මෙම ZIP project එක. Internet අවශ්‍ය වන්නේ Python/VS Code බාගැනීමට පමණි.

පළමු mini challenge

ඔබේ පරිගණකයේ Python version එක හොයාගන්න. `py --version` දැනටම වැඩ කරනවාද?

CONTENTS • 1/2

පාඩම් අංශ හඳුනාගනිමු

එක අංශයක් අවසන් කළාම code එක ඔබම run කර බලන්න. පහත අංශ සියල්ලේම සිංහල විස්තර, examples සහ tasks ඇත.

01 | Automation කියන්නේ මොකක්ද? 7

02 | Python සහ VS Code සූදානම් කිරීම 17

03 | Python මුල් පියවර 27

04 | Files, folders සහ pathlib 45

05 | Project 1: Safe File Organizer 57

06 | Project 2: CSV Report Automation 69

CONTENTS • 2/2

පාඩම් අංශ හඳුනාගනිමු

එක අංශයක් අවසන් කළාම code එක ඔබම run කර බලන්න. පහත අංශ සියල්ලේම සිංහල විස්තර, examples සහ tasks ඇත.

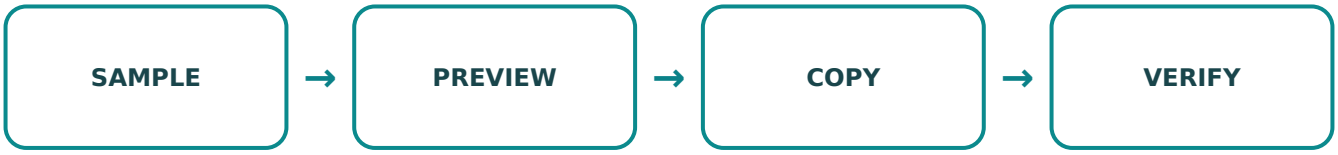
07 | Backups සහ ZIP Automation	82
08 | Windows Task Scheduler	91
09 | SDLC, Testing සහ QA	99
10 | 2026 automation දැනුම හා safe projects	111
11 | Exercises, MCQ සහ final project	123

අවසානයේ සම්පූර්ණ source code appendix, commands cheat sheet සහ official references ඇත.

SAFETY FIRST

කේතය run කිරීමට පෙර

Python automation පරිගණකයේ files වෙනස් කිරීමට පුළුවන්. එම නිසා මුලදී සැබෑ Desktop, Downloads හෝ photos folders භාවිතා නොකරන්න. මෙම ZIP එක සමඟ එන `demo_inbox` සහ `data` folders පමණක් භාවිතා කරන්න.



- ✓ Preview does not modify files.
- ✓ Organize COPIES, never deletes originals.
- ✓ Reports flag invalid rows.
- ✓ Backups need manual restore verification.
- ✗ Never automate real personal files on the first try.

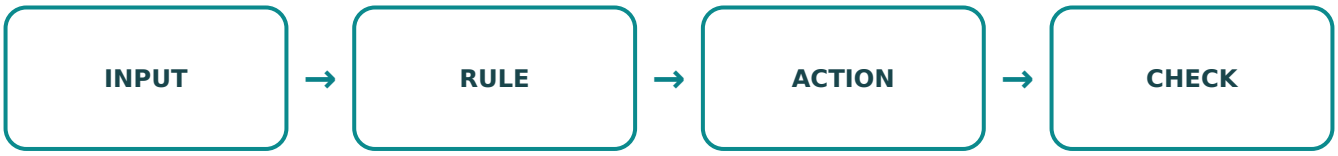
වයස අඩු සිසුන් scheduled jobs වැනි computer settings වෙනස් කිරීමට පෙර වැඩිහිටියෙකුගේ අවසරය ලබාගන්න.

UNIT 01 · LESSON 01 / 10

Automation කියන්නේ කුමක්ද?

එකම වැඩේ දවසින් දවස නැවත කරනවාද? Computer එකට නිශ්චිත පියවර ලියලා දුන්නාම ඒ වැඩේ නැවත කරවාගන්න පුළුවන්. ඒකට automation කියනවා. උදාහරණයක් ලෙස photos, documents සහ CSV files වෙන්කරගැනීම බලමු.

VISUAL SKETCH



මතක තබාගන්න / Expected

Files වර්ග කරලා, ගණන් කරලා, පිළිතුර පරීක්ෂා කිරීම

ඔබේ වාරය • Try it

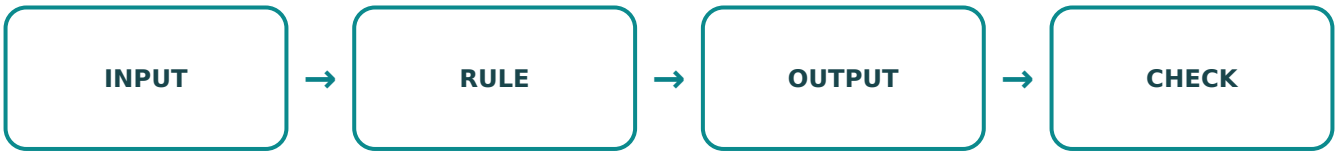
ඔබ ගෙදරදී නැවත නැවත කරන වැඩ තුනක් ලියන්න.

UNIT 01 · LESSON 02 / 10

මිනිසා කරන වැඩ සහ script එක

මිනිසා folder එකක් අරිනවා, file extension බලනවා, තැන තෝරනවා, copy කරනවා. Script එකටත් අනුගමනය කළ යුතු ඒ පියවරම තියෙනවා. Script එකට ඔබ නොකියූ දෙයක් සිතලා කරන්න බැහැ.

VISUAL SKETCH



මතක තබාගන්න / Expected

Computer එකට ක්‍රමවත් instructions අවශ්‍යයි.

ඔබේ වාරය • Try it

වැරදි folder එකකට copy කිරීම වළක්වන්නේ කොහොමද?

UNIT 01 · LESSON 03 / 10

Rule-based automation vs AI

Automation කියන්නේ අනිවාර්යයෙන්ම AI නොවේ. මේ පාඩමේ file organizer එක extension අනුව තීරණය කරන rule-based program එකක්. AI model එකක් train කිරීමක් නැහැ. මූලික automation තේරුම්ගැනීමෙන් පසුව පමණක් AI එකතු කරන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
if filename.endswith(".png"):
    folder = "Images"
else:
    folder = "Other"
```

මතක තබාගන්න / Expected

මෙය ඉගෙනගත් rules දෙකෙන් එකක් තෝරයි.

ඔබේ වාරය • Try it

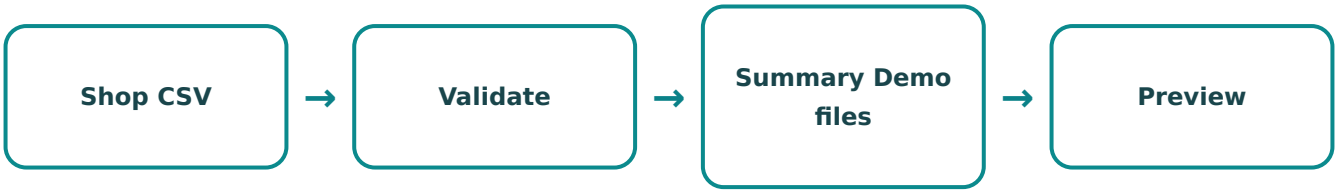
AI නැති automation එකකට උදාහරණයක් දෙන්න.

UNIT 01 · LESSON 04 / 10

කුඩා ව්‍යාපාරයක automation

Little Star Shop එකේ අය දවස අවසානයේ orders.csv ගණන් කරයි; ඒ data එකෙන් report එකක් හදයි; වැදගත් files backup කරයි. මේ වැඩ තුන අපේ project එකෙන් අත්හදා බලනවා. සියල්ල sample data පමණයි.

VISUAL SKETCH



මතක තබාගන්න / Expected

දත්ත ගණනය, file organization, backup වෙනම කාර්යය.

ඔබේ වාරය • Try it

ඔබේ project එකේ outputs නම් කරන්න.

UNIT 01 · LESSON 05 / 10

දැනට අපි නොකරන වැඩ

අපි සැබෑ WhatsApp messages කියවන්නේ නැහැ, email යවන්නේ නැහැ, වෙබ් අඩවි scrape කරන්නේ නැහැ. පළමුව offline sample folder එකක පුහුණු වෙමු. එය දත්ත ආරක්ෂාවට සහ වැරදි වළක්වන්නට උදව් වෙයි.

VISUAL SKETCH

FILE EXPLORER • folder preview

```
demo_inbox/ → safe to experiment
Personal Downloads/ → DO NOT use yet
```

මතක තබාගන්න / Expected

Demo data සමඟ පමණක් පුහුණු වීම හොඳම ආරම්භය.

ඔබේ වාරය • Try it

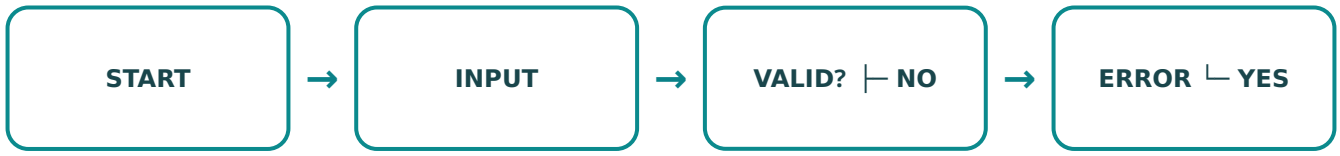
සැබෑ files වෙනස් කිරීමට පෙර backup එකක් අවශ්‍ය ඇයි?

UNIT 01 · LESSON 06 / 10

Automation workflow එක ඇඳීම

Flowchart එකක arrows මඟින් පියවර අනුපිළිවෙළ පෙන්නන්න පුළුවන්. Start සිට input, validation, action සහ review දක්වා තීරණ ලකුණු කරන්න. “Invalid” නම් program එක පැහැදිලි warning එකක් දිය යුතුයි.

VISUAL SKETCH



මතක තබාගන්න / Expected

Fail වුණාමත් ආරක්ෂිතව නවත්වන branch එකක්.

ඔබේ වාරය • Try it

CSV එකක් හිස් නම් workflow එක කරන්නේ මොකක්ද?

UNIT 01 · LESSON 07 / 10

Input සහ Output

Input කියන්නේ program එකට දෙන දේ; output කියන්නේ program එක නිපදවන දේ. Sales CSV එක input එකක්. sales_report.html එක output එකක්. මුල් input file එක නොමැකී තිබීම අපේ project එකේ වැදගත් නීතියක්.

VISUAL SKETCH

FILE EXPLORER • folder preview

```
Input: data/sales.csv
Output: output/reports/sales_report.html
Original CSV: unchanged
```

මතක තබාගන්න / Expected

Original files නොමැකීම අලුතෙන් ඉගෙනගන්නා අයට ආරක්ෂිතයි.

ඔබේ වාරය • Try it

ඔබට වැදගත් output file දෙකක් සඳහන් කරන්න.

UNIT 01 · LESSON 08 / 10

Dry run කියන්නේ මොකක්ද?

“Dry run” කියන්නේ ඇත්තට files copy/move නොකර කලින් සැලැස්ම පෙන්වන ක්‍රමය. අපේ `preview` command එකෙන් යන්න යන තැන බලන්න පුළුවන්. හරි නම් පසුව `organize` command එක දිය හැකියි.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py preview
# Print a plan only - no copying
```

මතක තබාගන්න / Expected

පළමු command එක කිසිවක් copy කරන්නේ නැහැ.

ඔබේ වාරය • Try it

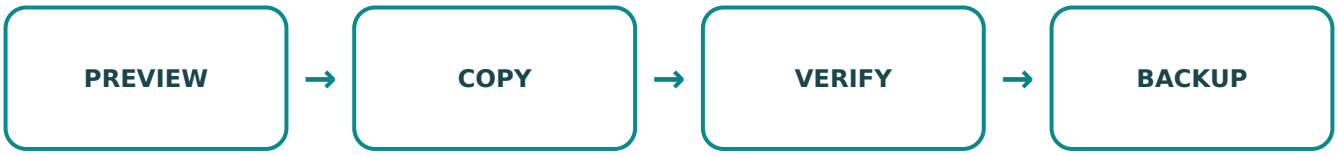
Dry run එකක් නැති automation එකක අවදානම කුමක්ද?

UNIT 01 · LESSON 09 / 10

ආරක්ෂිත නීති 5

1. Test folder එකක් භාවිතා කරන්න. 2. Preview එක බලන්න. 3. වැදගත් originals වෙනස් නොකරන්න. 4. Report එක අතින් පරීක්ෂා කරන්න. 5. Backup එක restore කළ හැකිද බලන්න. Automation එක වේගවත් වුණත් සෑම output එකක්ම නිවැරදි නොවෙයි.

VISUAL SKETCH



මතක තබාගන්න / Expected

"Run" කිරීමෙන් පසුවත් මිනිසාගේ review එක අවශ්‍යයි.

ඔබේ වාරය • Try it

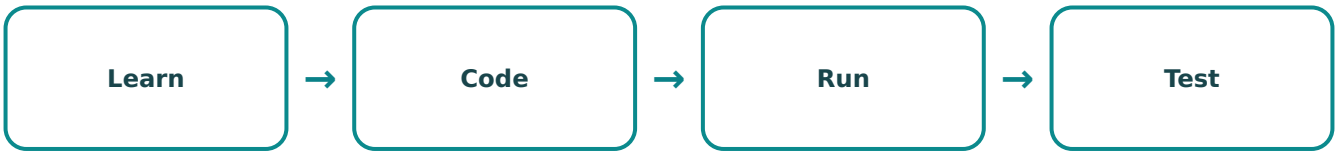
ඔබේම safety checklist එකක් සකස් කරන්න.

UNIT 01 · LESSON 10 / 10

පොත අවසානයේ ඔබට හැකි දේ

Python file එකක් CMD එකෙන් run කිරීම, folders වල filenames කියවීම, CSV data වල වැරදි හඳුනාගැනීම, HTML report එකක් browser එකෙන් බැලීම සහ ZIP backup එකක් ලබාගැනීම. අවසානයේ tests සහ documentation ද සකස් කරමු.

VISUAL SKETCH



මතක තබාගන්න / Expected

කුඩා project එකකින් සම්පූර්ණ software workflow එක ඉගෙනගන්නවා.

ඔබේ වාරය • Try it

අද ඔබට ඉගෙනගන්න ඕනම කොටස තෝරන්න.

UNIT 02 · LESSON 01 / 10

2026දී Python version එක

2026 ඔක්තෝබර් වනවිට Python 3.15 stable release එක ලබාගත හැකියි. මේ පොතේ code Python 3.12 හෝ ඊට නවතම supported versions වල standard library සමඟ වැඩ කරන ලෙස ලියා තිබෙනවා. ඔබේ computer එකේ version එක බලන්න.

RUN / PREVIEW

Windows • Command Prompt

```
py --version
# or
python --version
```

මතක තබාගන්න / Expected

Python 3.12+ එකක් තිබුණොත් හොඳයි.

ඔබේ වාරය • Try it

Version එක ලියාගන්න. Command එක වැඩ නැතිනම් installation බලන්න.

UNIT 02 · LESSON 02 / 10

Windows වල Python install කිරීම

python.org වෙතින් සුදුසු Windows installer එක ලබාගන්න. Installer එකේ PATH / launcher සැකසුම් ගැන කියවන්න. Installation අවසන් කළ පසුව අලුත් CMD window එකක් අරින්න. School computer එකකට install කිරීමට ගුරුවරයාගේ අවසරය ගන්න.

RUN / PREVIEW

file:///.../sales_report.html

https://www.python.org/downloads/

py --version



මතක තබාගන්න / Expected

නිල වෙබ් අඩවියෙන් software ලබාගැනීම ආරක්ෂිතයි.

ඔබේ වාරය • Try it

ඔබට install කරගැනීමට administrator permission අවශ්‍යද?

UNIT 02 · LESSON 03 / 10

VS Code කියන්නේ මොකක්ද?

Visual Studio Code යනු code ලියන්න පුළුවන් editor එකක්. වම් පැත්තෙන් folder tree බලන්න, මැද code ලියන්න, පහළ terminal එකෙන් command run කරන්න පුළුවන්. Python extension එකක් තිබුණොත් syntax highlighting සහ debugging පහසු වෙනවා.

RUN / PREVIEW

Visual Studio Code • Python

```
File > Open Folder
Terminal > New Terminal
File > New Text File
```

මතක තබාගන්න / Expected

VS Code editor එකයි, Python interpreter එකයි දෙකක්.

ඔබේ වාරය • Try it

VS Code සහ CMD අතර වෙනස වචන දෙකකින් කියන්න.

UNIT 02 · LESSON 04 / 10

CMD එකෙන් folder එකට යමු

Windows Start menu එකෙන් Command Prompt විවෘත කරන්න. `cd` command එකෙන් project folder එකට යන්න. Path එකේ spaces තිබේ නම් quotation marks යොදන්න. `dir` මඟින් files බලන්න.

RUN / PREVIEW

Windows • Command Prompt

```
cd "C:\Users\Student\Documents\PythonAutomation"
dir
```

මතක තබාගන්න / Expected

Folder path ඔබේ PC එකේ තැන අනුව වෙනස් වේ.

ඔබේ වාරය • Try it

`dir` command එකෙන් filename දෙකක් කියවන්න.

UNIT 02 · LESSON 05 / 10

Hello World script

Python program එකක් .py file එකක්. `print()` මඟින් text එක CMD එකේ පෙන්වනවා. Code VS Code තුළ ලියලා save කර `py filename.py` ලෙස run කරන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
print("Hello, automation hero!")
```

මතක තබාගන්න / Expected

Hello, automation hero!

> **ඔබේ වාරය • Try it**
> ඔබේම greeting එකක් `print()` කරලා බලන්න.

UNIT 02 · LESSON 06 / 10

Project folder සකස් කරමු

ZIP එක Extract කරලා project folder එක VS Code මඟින් විවෘත කරන්න. app.py, automation/, demo_inbox/, data/, docs/, tests/ යන folders සහ files හඳුනාගන්න. `output/` සහ `backups/` run කළ විට generate වෙයි.

VISUAL SKETCH

FILE EXPLORER • folder preview

```
EGOTECHWORLD_Python_Automation_Source_2026/
--- app.py
--- automation/
--- demo_inbox/
--- data/sales.csv
--- tests/
--- docs/
```

මතක තබාගන්න / Expected

මෙය real project එකේ ආරම්භක folder map එකයි.

ඔබේ වාරය • Try it

`app.py` file එක සොයා මොන folder එකේද කියන්න.

UNIT 02 · LESSON 07 / 10

Hello script එක run කරන්න

Project folder එකට CMD එකෙන් යාමෙන් පසු examples යටතේ ඇති script එක run කරන්න. Python launcher එක නොමැති පරිගණකයක `python` command එක භාවිතා කරන්න.

RUN / PREVIEW

Windows • Command Prompt

```
py examples\01_hello.py
# or
python examples/01_hello.py
```

මතක තබාගන්න / Expected

Hello, little automation hero!

ඔබේ වාරය • Try it

Error එකක් ආවොත් spelling සහ `cd` path පරීක්ෂා කරන්න.

UNIT 02 · LESSON 08 / 10

Traceback කියවමු

Error එකක් පෙනුණාම කලබල නොවන්න. අවසාන line එකේ error type එක බලන්න; file name සහ line number හඳුනාගන්න. FileNotFoundError නම් path වැරදි විය හැකියි; SyntaxError නම් code punctuation නැවත බලන්න.

RUN / PREVIEW

Windows • Command Prompt

```
FileNotFoundError: ...
SyntaxError: ...
NameError: ...
```

මතක තබාගන්න / Expected

Error message එක bug හොයාගන්න උදව් කරන ඉඟියක්.

ඔබේ වාරය • Try it

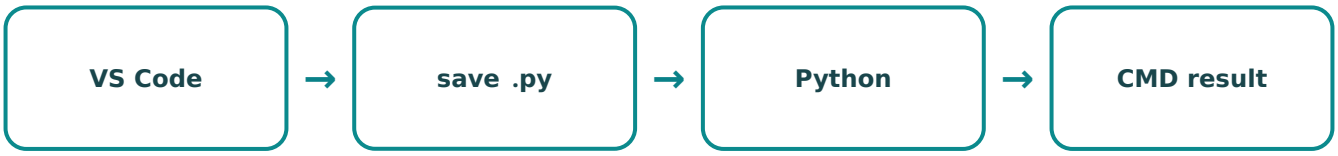
වැරදි filename එකක් දී error message එක කියවන්න.

UNIT 02 · LESSON 09 / 10

Interpreter vs editor

VS Code code ලියන තැන; interpreter එක ඒ code instructions ක්‍රියාත්මක කරන තැන. වෙනත් text editor එකක ලියපු `.py` file එකත් Python interpreter එකෙන් run කළ හැකියි.

VISUAL SKETCH



මතක තබාගන්න / Expected

Editor මාරු කළත් Python code එක එම භාෂාවමයි.

ඔබේ වාරය • Try it

මෙහි interpreter කියන්නේ කොයි කොටසද?

UNIT 02 · LESSON 10 / 10

File extensions සහ Windows

Windows වල file extension එක සැඟවී තිබිය හැකියි. View settings වල filename extensions දක්වන්න. `.py` කියන්නේ Python file; `.csv` කියන්නේ table data; `.html` browser page; `.zip` archive.

VISUAL SKETCH

FILE EXPLORER • folder preview

```
hello.py
orders.csv
sales_report.html
backup.zip
```

මතක තබාගන්න / Expected

File type එක filename එකේ අවසානයෙන් අනුමාන කළ හැකියි.

ඔබේ වාරය • Try it

පහත නම්වල extension එක හඳුනාගන්න: photo.png, notes.txt.

UNIT 03 · LESSON 01 / 18

Variables: නමින් පෙට්ටි

Variable එකක් කියන්නේ data එකක් තබන නම් කළ පෙට්ටියක් වගේ. `shop_name` text එකක්; `orders` integer එකක්. `print()` කරලා දෙකම පෙන්වන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
shop_name = "Little Star"
orders = 3
print(shop_name)
print(orders)
```

මතක තබාගන්න / Expected

```
Little Star
3
```

ඔබේ වාරය • Try it

`orders` අගය 5 කර output එක බලන්න.

UNIT 03 · LESSON 02 / 18

Strings: වචන සමඟ වැඩ

Text අගය strings කියලා හඳුන්වනවා. Quotes ඇතුළේ text ලියනවා. `.lower()` හෝ `.strip()` මඟින් simple text cleaning කරගත හැකියි. File extension එක lower-case කිරීම වැදගත් වන්නේ `.PNG` සහ `.png` එකම වර්ගය නිසා.

RUN / PREVIEW

Visual Studio Code • Python

```
filename = "  PHOTO.PNG  "
clean = filename.strip().lower()
print(clean)
```

මතක තබාගන්න / Expected

photo.png

> **ඔබේ වාරය • Try it**
> Whitespace ඉවත් නොකළොත් මොන ගැටලුවද?

UNIT 03 · LESSON 03 / 18

Numbers: ගණන් කිරීම

Quantity එක integer එකක්. Price එක decimal value එකක්. පසුව currency calculation සඳහා float වෙනුවට Decimal භාවිතා කරනවා; මුලදී `int` ගණනය ඉගෙනගනිමු.

RUN / PREVIEW

Visual Studio Code • Python

```
files = 4
new_files = 2
print(files + new_files)
```

මතක තබාගන්න / Expected

6

ඔබේ වාරය • Try it

Files 10ක් තිබියදී 3ක් අලුතින් ආවොත් මුළු ගණන?

UNIT 03 · LESSON 04 / 18

Boolean: True/False

පිළිතුර ඔව් හෝ නැහැ නම් boolean එකක් භාවිතා කළ හැකියි. File එකක් තිබෙනවාද නැද්ද කියලා condition එකකට දෙන පිළිතුර boolean එකක්.

RUN / PREVIEW

Visual Studio Code • Python

```
is_ready = True
if is_ready:
    print("Start automation")
```

මතක තබාගන්න / Expected

Start automation

> **ඔබේ වාරය • Try it**
> `False` කළොත් print වෙනවාද?

UNIT 03 · LESSON 05 / 18

`if` condition එක

Condition එකක් කියන්නේ “එහෙම වුණොත් මෙහෙම කරන්න” කියන නියමයයි. `.endswith()` මඟින් extension එක පරීක්ෂා කළ හැකියි. අවසානයේ colon සහ ඊළඟ line එකේ indentation වැදගත්.

RUN / PREVIEW

Visual Studio Code • Python

```
file = "drawing.png"
if file.endswith(".png"):
    print("Image file")
else:
    print("Another type")
```

මතක තබාගන්න / Expected

Image file

ඔබේ වාරය • Try it

`.txt` file එකක් දී output එක බලන්න.

UNIT 03 · LESSON 06 / 18

Indentation වැදගත් ඇයි?

Python හි space ප්‍රමාණය code block එක හඳුනාගැනීමට භාවිතා වෙනවා. `if`, `for`, `def` යට code lines එකම මට්ටමකට indent කරන්න. අහඹු spaces නිසා IndentationError එන්න පුළුවන්.

RUN / PREVIEW

Visual Studio Code • Python

```
for number in range(3):
    print(number)
```

මතක තබාගන්න / Expected

```
0
1
2
```

> **ඔබේ වාරය • Try it**
> `print` එක ඉදිරියට spaces 4ක් දී අත්හදා බලන්න.

UNIT 03 · LESSON 07 / 18

Lists: බොහෝ දේ එකට

List එකක filenames ගණනාවක් තබන්න පුළුවන්. `[ ]` brackets භාවිතා කරන්න. `len()` මඟින් items ගණන ලබාගන්න. List එකේ order එකත් ඇත.

RUN / PREVIEW

Visual Studio Code • Python

```
files = ["a.png", "b.txt", "c.csv"]
print(len(files))
print(files[0])
```

මතක තබාගන්න / Expected

```
3
a.png
```

ඔබේ වාරය • Try it

List එකට `d.pdf` එකතු කරන්න.

UNIT 03 · LESSON 08 / 18

Loops: නැවත නැවත වැඩ

Folder එකේ files 100ක් තිබුණත් එකින් එක instruction ලියන්න ඕන නැහැ. `for` loop එකෙන් file එකකට වරක් කරලා එය නැවත කරයි.

RUN / PREVIEW

Visual Studio Code • Python

```
for name in ["a.png", "b.txt"]:
    print("Checking", name)
```

මතක තබාගන්න / Expected

Checking a.png
Checking b.txt

ඔබේ වාරය • Try it

නම් තුනක list එකක loop එකක් ලියන්න.

UNIT 03 · LESSON 09 / 18

`range()` සහ iteration

නියමිත වාර ගණනකට loop එකක් කිරීමට `range()` භාවිතා කරනවා. `range(1, 4)` 1, 2, 3 ලබා දෙයි. මෙය retry නොව මූලික counting concept එකක්.

RUN / PREVIEW

Visual Studio Code • Python

```
for day in range(1, 4):
    print("Day", day)
```

මතක තබාගන්න / Expected

```
Day 1
Day 2
Day 3
```

ඔබේ වාරය • Try it

දින 5ක් පෙන්වන loop එකක් ලියන්න.

UNIT 03 · LESSON 10 / 18

Functions: නැවත භාවිතා කරන කොටස්

එකම code එක තැන් කිහිපයකට copy නොකර function එකක් ලෙස ලියන්න. `def` වචනයෙන් පටන්ගෙන arguments භාවිතා කරන්න. Function එක නිශ්චිත කාර්යයකට වගකිව යුතුයි.

RUN / PREVIEW

Visual Studio Code • Python

```
def label_file(name):
    return "File: " + name

print(label_file("notes.txt"))
```

මතක තබාගන්න / Expected

File: notes.txt

ඔබේ වාරය • Try it

`label_file` වෙනුවට `greet` function එකක් ලියන්න.

UNIT 03 · LESSON 11 / 18

Return සහ print වෙනස

`print()` එක user ට දෙයක් පෙන්වනවා. `return` එක function එකෙන් result එක ආපසු ලබාදෙනවා; ඒ result එක පසුව තවත් calculation එකකට භාවිතා කළ හැකියි.

RUN / PREVIEW

Visual Studio Code • Python

```
def add(a, b):
    return a + b

total = add(2, 3)
print(total)
```

මතක තබාගන්න / Expected

5

ඔබේ වාරය • Try it

Return කරපු අගය තවත් 10කින් වැඩි කරන්න.

UNIT 03 · LESSON 12 / 18

Dictionary: label → value

Dictionary එකක key සහ value pairs තබන්න පුළුවන්. Report එකේ product name එකට total revenue එක map කිරීමට මෙය උපකාරී වෙනවා.

RUN / PREVIEW

Visual Studio Code • Python

```
sales = {"Pencil": 350, "Notebook": 690}
print(sales["Pencil"])
```

මතක තබාගන්න / Expected

350

> **ඔබේ වාරය • Try it**
> `Eraser: 120` pair එක එකතු කරන්න.

UNIT 03 · LESSON 13 / 18

`try` සහ `except`

User වැරදි quantity එකක් type කළාම app එක crash නොවී warning එකක් දිය යුතුයි. `try` block එක උත්සාහ කරයි; `except` එක නිශ්චිත error එක හඳුනාගනියි.

RUN / PREVIEW

Visual Studio Code • Python

```
try:
    quantity = int("abc")
except ValueError:
    print("Enter a whole number")
```

මතක තබාගන්න / Expected

Enter a whole number

> **ඔබේ වාරය • Try it**
> `int("7")` ලෙස වෙනස් කළොත් මොකක්ද?

UNIT 03 · LESSON 14 / 18

Imports සහ standard library

Python සමඟ එන library වල ඉතා වැදගත් tools තියෙනවා. `import csv` CSV සඳහා; `from pathlib import Path` folders සඳහා; `from zipfile import ZipFile` ZIP archive සඳහා. මේ project එකට pip packages අවශ්‍ය නැහැ.

RUN / PREVIEW

Visual Studio Code • Python

```
from pathlib import Path
import csv
from zipfile import ZipFile
```

මතක තබාගන්න / Expected

Standard library = Python සමඟම ලබාගන්නා tools.

ඔබේ වාරය • Try it

මේ imports 3ට එක් එක් වැඩක් ලියන්න.

UNIT 03 · LESSON 15 / 18

`__name__` guard

Python file එක සෘජුව run කරන විට ප්‍රධාන function එක ධාවනය කිරීමට මෙය භාවිතා වෙනවා. වෙනත් file එකකින් import කළ විට menu එක අහම්බෙන් open නොවන ලෙස උදව් වෙයි.

RUN / PREVIEW

Visual Studio Code • Python

```
def main():
    print("App started")

if __name__ == "__main__":
    main()
```

මතක තබාගන්න / Expected

App started

ඔබේ වාරය • Try it

main() යන්න call නොකළොත් program එක මොනවා කරයි?

UNIT 03 · LESSON 16 / 18

Comments: code පිළිබඳ සටහන්

`#` එකෙන් ආරම්භ වන line එක Python execute කරන්නේ නැහැ. දරුවන්ට සහ team membersට purpose එක තේරෙන්න ඉංග්‍රීසි comments ලියමු.

RUN / PREVIEW

Visual Studio Code • Python

```
# A safe preview never changes files.
print("Preview only")
```

මතක තබාගන්න / Expected

Preview only

ඔබේ වාරය • Try it

ඔබේ code එකට තේරුම්ගත හැකි English comment එකක් ලියන්න.

UNIT 03 · LESSON 17 / 18

Input ගැන සලකන්න

`input()` user එකෙන් text ලබාගනියි. ඕනෑම input එකක් නිවැරදි බවට විශ්වාස නොකර පරීක්ෂා කරන්න. මේ demo app එක menu number එකෙන් නිශ්චිත options පමණක් තෝරාගනියි.

RUN / PREVIEW

Visual Studio Code • Python

```
choice = input("Choose 1 or 2: ")
if choice in ("1", "2"):
    print("Valid choice")
```

මතක තබාගන්න / Expected

User input එකක් මත ගන්නා තීරණය validation සමඟ කළ යුතුයි.

ඔබේ වාරය • Try it

`3` ඇතුළත් කළොත් මොනවා වෙයි?

UNIT 03 · LESSON 18 / 18

Python mini test

ඔබ පහත code එක run කිරීමට පෙර result එක කඩදාසියක ලියන්න. පසුව code එක ධාවනය කර ඔබේ පිළිතුර සමඟ සසඳන්න. Prediction → Run → Compare ක්‍රමය හොඳ learning habit එකක්.

RUN / PREVIEW

Visual Studio Code • Python

```
files = ["a.txt", "b.csv", "c.png"]
count = 0
for name in files:
    if name.endswith(".txt"):
        count += 1
print(count)
```

මතක තබාගන්න / Expected

1

ඔබේ වාරය • Try it

`.csv` ද ගණන් කිරීමට condition එක වෙනස් කරන්න.

UNIT 04 · LESSON 01 / 12

Path කියන්නේ ලිපිනයක්

ඔබේ නිවසට ලිපිනයක් තියෙනවා වගේ computer file එකකටත් path එකක් තියෙනවා. Windows වල drive letter සහ backslashes දක්නට ලැබේ. `Path` object එකෙන් paths පහසුවෙන් join කළ හැකියි.

RUN / PREVIEW

Visual Studio Code • Python

```
from pathlib import Path
folder = Path("demo_inbox")
file = folder / "notes.txt"
print(file)
```

මතක තබාගන්න / Expected

demo_inbox\notes.txt (Windows)

ඔබේ වාරය • Try it

ඔබේ folder එකට relative path එකක් ලියන්න.

UNIT 04 · LESSON 02 / 12

Relative vs absolute paths

Relative path එක current working folder එක මත පදනම් වෙයි. Absolute path එක root / drive සිට පටන්ගන්නා සම්පූර්ණ ලිපිනයයි. අපේ project එකේ `__file__` භාවිතයෙන් project root එක සොයාගනියි.

RUN / PREVIEW

Visual Studio Code • Python

```
from pathlib import Path
ROOT = Path(__file__).resolve().parent
print(ROOT)
```

මතක තබාගන්න / Expected

Folder වෙනස් වුණත් script location හඳුනාගන්න පුළුවන්.

ඔබේ වාරය • Try it

`Path.cwd()` සහ `__file__` වෙනස සොයා බලන්න.

UNIT 04 · LESSON 03 / 12

Folder එකක් කියවීම

`iterdir()` folder එකේ ඇතුළත් items එක් එක් එක ලබාදෙනවා. හැම item එකක්ම file නොවිය හැකියි; `is_file()` වලින් files පමණක් තෝරන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
from pathlib import Path
for item in Path("demo_inbox").iterdir():
    if item.is_file():
        print(item.name)
```

මතක තබාගන්න / Expected

notes.txt, orders.csv වැනි නම් CMD තුළ පෙනේ.

ඔබේ වාරය • Try it

Subfolder එකක් එක් කළොත් එය print වෙයිද?

UNIT 04 · LESSON 04 / 12

File suffix එක හඳුනාගන්න

`.suffix` property එකෙන් extension එක ලබාගන්න පුළුවන්. `.lower()` යොදා cases එකට ගළපන්න. `invoice.PDF` සහ `invoice.pdf` දෙකම Documents category එකට දැමිය හැකියි.

RUN / PREVIEW

Visual Studio Code • Python

```
from pathlib import Path
file = Path("PHOTO.PNG")
print(file.suffix.lower())
```

මතක තබාගන්න / Expected

.png

ඔබේ වාරය • Try it

`.docx` suffix එක අත්හදා බලන්න.

UNIT 04 · LESSON 05 / 12

Folder tree sketch එක

Projects වල code, input, output, tests වෙන වෙනම folders තබන එක වැදගත්. Input සහ output එකම folder එකේ තැබුවොත් automation නැවත run කරන විට කලින් සාදපු files නැවත process විය හැකියි.

VISUAL SKETCH

FILE EXPLORER • folder preview

```
project/
--- demo_inbox/
--- data/
--- output/organized/
--- output/reports/
--- backups/
```

මතක තබාගන්න / Expected

Inputs සහ outputs අතර පැහැදිලි සීමාවක් තබන්න.

ඔබේ වාරය • Try it

Output folder එක input තුළ නොතැබිය යුත්තේ ඇයි?

UNIT 04 · LESSON 06 / 12

Files copy vs move

`shutil.copy2()` මුල් file එකත් තබා copy එකක් සෑදිය හැකියි. `move()` නම් මුල් තැනෙන් file එක වෙනස් විය හැකි නිසා ආරම්භක පාඩමේ එය භාවිතා නොකරමු.

RUN / PREVIEW

Visual Studio Code • Python

```
import shutil
shutil.copy2("source.txt", "copy.txt")
```

මතක තබාගන්න / Expected

මෙම උදාහරණය run කිරීමට source.txt පළමුව තිබිය යුතුයි.

ඔබේ වාරය • Try it

Copy සහ move අතර වෙනස පෙන්වන රූපයක් ඇඳන්න.

UNIT 04 · LESSON 07 / 12

Folder එකක් හදමු

නව folder එකක් සෑදීමට `mkdir()` භාවිතා කරනවා. `parents=True` නිසා මැද folders නොමැති වුණත් ඒවා සෑදෙයි. `exist_ok=True` නිසා තිබෙන folder එකක් නිසා error නෑ.

RUN / PREVIEW

Visual Studio Code • Python

```
from pathlib import Path
Path("output/reports").mkdir(
    parents=True, exist_ok=True
)
```

මතක තබාගන්න / Expected

output/reports folder එක සෑදේ.

> **ඔබේ වාරය • Try it**
> `output/images` folder එකක් හදන්න.

UNIT 04 · LESSON 08 / 12

File content කියවීම

Small text file එකක් `read_text()` මඟින් කියවිය හැකියි. නමුත් බොහෝ විශාල files සඳහා streaming ක්‍රම සුදුසුයි. Privacy ඇති files අහම්බෙන් කියවන්න එපා.

RUN / PREVIEW

Visual Studio Code • Python

```
from pathlib import Path
message = Path("demo_inbox/notes.txt")
print(message.read_text(encoding="utf-8"))
```

මතක තබාගන්න / Expected

Hello - this is a sample note.

ඔබේ වාරය • Try it

README file එකක් ඇති නම් එය කියවන්න.

UNIT 04 · LESSON 09 / 12

`exists()` සහ missing files

Automation එකට අවශ්‍ය input folder එක නැතිනම් සාර්ථකව ඉදිරියට නොයා error එකක් පෙන්විය යුතුයි. `exists()` හෝ `is_dir()` මඟින් ඒක බලන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
from pathlib import Path
folder = Path("demo_inbox")
print(folder.is_dir())
```

මතක තබාගන්න / Expected

True (folder එක තිබුණොත්)

ඔබේ වාරය • Try it

Folder name spelling වැරදි කළොත් output එක?

UNIT 04 · LESSON 10 / 12

`resolve()` සහ location validation

Real automation එකේ inbox සහ output folder අහම්බෙන් එකම තැන වුණොත් ගැටලු ඇතිවිය හැකියි. අපේ project එක `resolve()` කර paths සසඳලා nested folders ප්‍රතික්ෂේප කරනවා.

RUN / PREVIEW

Visual Studio Code • Python

```
if destination == inbox:
    raise ValueError("Separate folders needed")
```

මතක තබාගන්න / Expected

මෙය safety guard එකක්, අමතර feature එකක් නොවේ.

ඔබේ වාරය • Try it

`output` එක input තුළ තබා test කලොත් මොකක් වෙයි?

UNIT 04 · LESSON 11 / 12

Symlink සහ hidden surprises

Symbolic link කියන්නේ වෙනත් location එකකට යොමුකරන විශේෂ link එකක්. පළමු automation project එකකදී links follow නොකිරීම හොඳයි. අපේ organizer එක symlink files skip කරනවා.

RUN / PREVIEW

Visual Studio Code • Python

```
if source.is_symlink():
    continue
```

මතක තබාගන්න / Expected

අපේ sample folder එකේ සාමාන්‍ය files පමණක් තියෙනවා.

ඔබේ වාරය • Try it

Unknown file එකක් දුටුවොත් delete කළ යුතුද?

UNIT 04 · LESSON 12 / 12

Filename collisions

එකම නමින් files දෙකක් තිබුණොත් මුල් එක overwrite කිරීම භයානකයි. අපේ organizer එක target file එක වෙනස් නම් `_2`, `_3` වැනි suffix එකක් එකතු කරනවා. එම bytes තිබුණොත් නැවත copy නොකරයි.

VISUAL SKETCH

FILE EXPLORER • folder preview

```
note.txt → Documents/note.txt
note.txt (different) → note_2.txt
```

මතක තබාගන්න / Expected

Overwrite නොකිරීම ආරක්ෂිත file-handling නීතියක්.

ඔබේ වාරය • Try it

Collisions තුනක් තිබුණොත් කුමන නම් යෝජනා කරන්නේද?

UNIT 05 · LESSON 01 / 12

File organizer project එක ඇරඹීම

දැන් අපි සැබෑ project එක run කර බලමු. ZIP එක Extract කර `app.py` ඇති folder එකට CMD එකෙන් යන්න. මුලින් `preview` පමණක් run කරන්න. කිසිම file එකක් වෙනස් නොවෙයි.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py preview
```

මතක තබාගන්න / Expected

copy | notes.txt → Documents/notes.txt වැනි plan එකක්.

> **ඔබේ වාරය • Try it**
> ඔබේ computer එකේ category හතර නම් කරන්න.

UNIT 05 · LESSON 02 / 12

Extension mapping table

File types සඳහා dictionary එකක් තියෙනවා. .png, .jpg Images; .txt, .pdf Documents; .csv, .xlsx Spreadsheets; .py, .html Code. වෙනත් extension එකක් Other folder එකට යයි.

RUN / PREVIEW

DATA / SAMPLE

Images: .png, .jpg, .jpeg, .gif, .webp
Documents: .pdf, .docx, .txt
Spreadsheets: .csv, .xlsx
Code: .py, .html, .css, .js
Other: unknown types

මතක තබාගන්න / Expected

Folder rules පොතෙන් වෙනස් කරගන්න පුළුවන්.

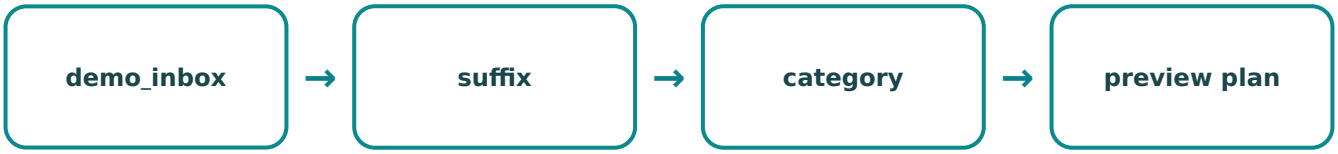
> **ඔබේ වාරය • Try it**
> `.mp3` සඳහා නව category එකක් යෝජනා කරන්න.

UNIT 05 · LESSON 03 / 12

Data flow sketch

Input folder එකේ top-level files තෝරාගෙන, filename extension හඳුනාගෙන, target category folder එක තීරණය කරනවා. Preview එක approval වගේ ක්‍රියා කරයි. Apply command එක පමණක් copy කරනවා.

VISUAL SKETCH



මතක තබාගන්න / Expected

Output එක input එකෙන් වෙනස් folder එකක.

ඔබේ වාරය • Try it

Dry run branch එක flowchart එකට එකතු කරන්න.

UNIT 05 · LESSON 04 / 12

`category_for()` function

එක filename එකක් දී කුමන folder එකට යනවාද කියලා function එක තීරණය කරයි. යම් suffix එක mapping එකේ නැත්නම් Other කියලා ආපසු ලබා දෙයි.

RUN / PREVIEW

Visual Studio Code • Python

```
from automation.organizer import category_for
print(category_for("kitten.png"))
print(category_for("info.txt"))
print(category_for("data.xyz"))
```

මතක තබාගන්න / Expected

Images
Documents
Other

ඔබේ වාරය • Try it

`.PDF` දෙන විට Documents ලැබෙන්නේ ඇයි?

UNIT 05 · LESSON 05 / 12

Preview command එකෙන් බලමු

CMD එකේ preview command එක run කර filename සහ target path එක කියවන්න. `copy` යනු ඉදිරියේදී copy කරන්න නියමිතයි යන්නයි. `unchanged` යනු දැනට එම content තියෙන නිසා නැවත copy නොකරන බවයි.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py preview
```

මතක තබාගන්න / Expected

```
copy      | hello.py -> Code/hello.py
copy      | sample.png -> Images/sample.png
```

ඔබේ වාරය • Try it

Preview කිරීමෙන් පසු output folder සෑදී තිබේද?

UNIT 05 · LESSON 06 / 12

Organize command එක

Plan එක නිවැරදි බව තහවුරු වුණාම `organize` command එක run කරන්න. Source files මකා නොදමයි. Code, Documents, Spreadsheets සහ Images folders output යටතේ සෑදෙයි.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py organize
```

මතක තබාගන්න / Expected

Copied 4 files. Originals are unchanged.

ඔබේ වාරය • Try it

`demo_inbox` සහ `output/organized` folders දෙක සසඳන්න.

UNIT 05 · LESSON 07 / 12

නැවත run කිරීම

Automation එක දෙවැනි වර run කරන විට duplicate copies සෑදෙනවාද බලන්න. Same name සහ same bytes තිබුණොත් status `unchanged` වේ. මේකට idempotent behavior යන අර්ථයෙන් සමාන අදහසක් තියෙනවා.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py organize
```

මතක තබාගන්න / Expected

Copied 0 files. Originals are unchanged.

> **ඔබේ වාරය • Try it**
> ඇයි දෙවන වර 0 copy වූයේ?

UNIT 05 · LESSON 08 / 12

`shutil.copy2()`

Python standard library එකේ shutil file copy සඳහා භාවිතා වෙනවා. `copy2()` මඟින් file content සහ සමහර metadata copy කළ හැකියි. එහෙත් original directory permissions, cross-system metadata වැනි දේ 100% එකම වන බවට විශ්වාස නොකරන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
from shutil import copy2
copy2("file.txt", "copy.txt")
```

මතක තබාගන්න / Expected

File දෙකම පසුව තිබිය හැකියි.

ඔබේ වාරය • Try it

Copy එක සහ original එක සමානද කියලා බලන්න.

UNIT 05 · LESSON 09 / 12

SHA-256 content comparison

File names සමාන වුණත් content වෙනස් විය හැකියි. අපේ demo project එක hashlib SHA-256 digest භාවිතයෙන් bytes එකමද බලනවා. මෙය authenticity signature එකක් නොව file content comparison ක්‍රමයක්.

RUN / PREVIEW

Visual Studio Code • Python

```
import hashlib
hashlib.sha256(b"hello").hexdigest()
```

මතක තබාගන්න / Expected

Digest කියන්නේ bytes සඳහා ලැබෙන දිගු code එකක්.

ඔබේ වාරය • Try it

දෙවැනි content එක `hello!` කළොත් digest වෙනස් වෙයිද?

UNIT 05 · LESSON 10 / 12

Collision practice

Output එකේ Documents/notes.txt තිබියදී input එකේ notes.txt වෙනස් කළොත් organizer එක original target overwrite නොකර notes_2.txt වැනි නමක් තෝරයි. පුහුණුවෙන් පසුව sample folder එක restore කරන්න.

VISUAL SKETCH

FILE EXPLORER • folder preview

```
Documents/notes.txt
Documents/notes_2.txt
```

මතක තබාගන්න / Expected

අලුත් file එකට වෙනත් නමක් දීම data loss වළක්වයි.

ඔබේ වාරය • Try it

Collision test case එකක් ලියාගන්න.

UNIT 05 · LESSON 11 / 12

Unknown extensions

Code එක extension එක නොදන්නා file එකක් ලැබුණොත් delete නොකර Other category එකට දමයි. මෙය rule set එකේ default case එකයි. File type නම පමණක් සලකා unsafe files run නොකරන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
category_for("mystery.xyz")
```

මතක තබාගන්න / Expected

Other

> **ඔබේ වාරය • Try it**
> Other folder එකේ files වඩාත් පරීක්ෂා කළ යුත්තේ ඇයි?

UNIT 05 · LESSON 12 / 12

Organize module එකේ සීමාව

මේ demo එක subfolders ඇතුළට යන්නේ නැහැ; top-level files පමණක් බලනවා. Original files copy පමණක් වෙනවා. Live Downloads folder එකකට මේ code එක run කිරීමට පෙර සැලැස්ම, backups, conflicts සහ privacy සලකා බලන්න.

VISUAL SKETCH

FILE EXPLORER • folder preview

```
demo_inbox/notes.txt  → Included
demo_inbox/old/note.txt → Not included
```

මතක තබාගන්න / Expected

මෙම සීමා දැනගැනීම හොඳ developer පුරුද්දක්.

ඔබේ වාරය • Try it

Subfolders process කිරීමෙන් ඇතිවන අලුත් අවදානම් දෙකක්.

UNIT 06 · LESSON 01 / 13

CSV කියන්නේ මොකක්ද?

CSV කියන්නේ comma-separated values. එක් එක් row එක record එකක්. පළමු row එක බොහෝවිට headers වේ. Excel වගේ table එකක් text file එකකින් නිරූපණය කරන සරල ක්‍රමයක්.

RUN / PREVIEW

DATA / SAMPLE

```
date,product,quantity,unit_price
2026-10-01,Pencil,2,50.00
2026-10-01,Notebook,1,230.00
```

මතක තබාගන්න / Expected

Row දෙකම වෙන වෙනම order lines.

ඔබේ වාරය • Try it

Headers හතර සලකුණු කරන්න.

UNIT 06 · LESSON 02 / 13

CSV file එක Python වලින් කියවන්න

`csv.DictReader()` headers අනුව row එක dictionary එකක් බවට පරිවර්තනය කරයි. `newline=""` සහ encoding දීම ආරක්ෂිතයි.

RUN / PREVIEW

Visual Studio Code • Python

```
import csv
with open("data/sales.csv", newline="",
          encoding="utf-8") as file:
    for row in csv.DictReader(file):
        print(row["product"])
```

මතක තබාගන්න / Expected

Pencil, Notebook වැනි items පෙන්නයි.

ඔබේ වාරය • Try it

`quantity` column එක පෙන්වන ලෙස වෙනස් කරන්න.

UNIT 06 · LESSON 03 / 13

Column validation

CSV file එකේ අවශ්‍ය header එකක් නොමැති නම් total ගණනය කරන්න බැහැ. Program එක ඉක්මනින් error එකක් පෙන්විය යුතුයි; වැරදි report එකක් නිර්මාණය නොකරන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
required = {"date", "product",
            "quantity", "unit_price"}
```

මතක තබාගන්න / Expected

මේ සියලු columns CSV එකේ තිබිය යුතුයි.

ඔබේ වාරය • Try it

`unit_price` අහම්බෙන් මැකුවොත් මොකක් වෙයි?

UNIT 06 · LESSON 04 / 13

`quantity × price`

Revenue එක සරලව quantity එක unit_price එකෙන් ගුණ කළ අගයයි. Profit සමාන නෑ; cost සහ expenses අඩු කළ යුතුයි. Example: pencils 2 × Rs 50 = Rs 100.

RUN / PREVIEW

Visual Studio Code • Python

```
quantity = 2
unit_price = 50
revenue = quantity * unit_price
print(revenue)
```

මතක තබාගන්න / Expected

100

ඔබේ වාරය • Try it

Notebooks 3 × Rs 230 සඳහා revenue ගණනය කරන්න.

UNIT 06 · LESSON 05 / 13

Currency සහ Decimal

Money calculations වල float සමහරවිට rounding behavior නිසා අසමත් විය හැකියි. ඒ වෙනුවට `Decimal("50.00")` භාවිතා කරනවා. CSV එකෙන් එන price text එක Decimal බවට වෙනස් කරන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
from decimal import Decimal
price = Decimal("50.00")
print(price * 2)
```

මතක තබාගන්න / Expected

100.00

ඔබේ වාරය • Try it

`0.10 + 0.20` float සහ Decimal භාවිතයෙන් සසඳන්න.

UNIT 06 · LESSON 06 / 13

Bad rows ප්‍රතික්ෂේප කිරීම

Sample CSV එකේ quantity 0 ඇති වැරදි row එකක් තියෙනවා. ඒ row එක valid sale එකක් නොවන නිසා report total එකට එකතු නොකර warning එකක් දක්වයි. Silent errors නොකර review list එකක් තබන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
if quantity <= 0:
    raise ValueError("Invalid quantity")
```

මතක තබාගන්න / Expected

Row 9: Invalid quantity or price

ඔබේ වාරය • Try it

quantity -1 නම් valid ද? ඇයි?

UNIT 06 · LESSON 07 / 13

Grouping by product

Product name එක key එකක් ලෙස dictionary එකක sum එක රැස් කරගන්නවා. Pencil ලැබෙන හැම row එකකටම එහි total එක වැඩිවේ. `defaultdict` මඟින් පළමු අගය 0 ලෙස සලකන්න ලේසියි.

RUN / PREVIEW

Visual Studio Code • Python

```
from collections import defaultdict
totals = defaultdict(int)
totals["Pencil"] += 100
totals["Pencil"] += 200
print(totals["Pencil"])
```

මතක තබාගන්න / Expected

300

ඔබේ වාරය • Try it

Notebook සඳහා 230 + 460 ගණනය කරන්න.

UNIT 06 · LESSON 08 / 13

Real sample: totals බලමු

අපේ data/sales.csv වල valid rows 7ක්, invalid row 1ක් තියෙනවා. Pencil Rs 350, Notebook Rs 690, Eraser Rs 120, Color Pen Rs 250. මේ සියල්ල එකතු කළ sample revenue Rs 1,410.00.

RUN / PREVIEW

DATA / SAMPLE

```
Pencil      Rs 350.00
Notebook    Rs 690.00
Eraser      Rs 120.00
Color Pen   Rs 250.00
--------------------
TOTAL       Rs 1410.00
```

මතක තබාගන්න / Expected

Revenue = Rs 1410.00; profit නොවේ.

> **ඔබේ වාරය • Try it**
> සියලු totals එකතු කර නැවත පරීක්ෂා කරන්න.

UNIT 06 · LESSON 09 / 13

Report command එක ධාවනය කරන්න

Project folder එකේ `py app.py report` run කරන්න. එය `sales_summary.csv` සහ `sales_report.html` files දෙකක් සෑදෙයි. Invalid row warning එක CMD එකේ පෙනේ.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py report
```

මතක තබාගන්න / Expected

```
Summary: output/reports/sales_summary.csv
Browser report: output/reports/sales_report.html
Review: Row 9: Invalid quantity or price
```

ඔබේ වාරය • Try it

Output files දෙකම සොයාගන්න.

UNIT 06 · LESSON 10 / 13

CSV summary output එක

Program එක original CSV වෙනස් නොකර new CSV report එකක් ලියනවා. Header `product,revenue_lkr` වන අතර එක් එක් product එකට total ඇත. Excel හෝ text editor එකකින් open කරන්න.

RUN / PREVIEW

DATA / SAMPLE

```
product,revenue_lkr
Color Pen,250.00
Eraser,120.00
Notebook,690.00
Pencil,350.00
```

මතක තබාගන්න / Expected

CSV එක table විදිහටත් කියවිය හැකියි.

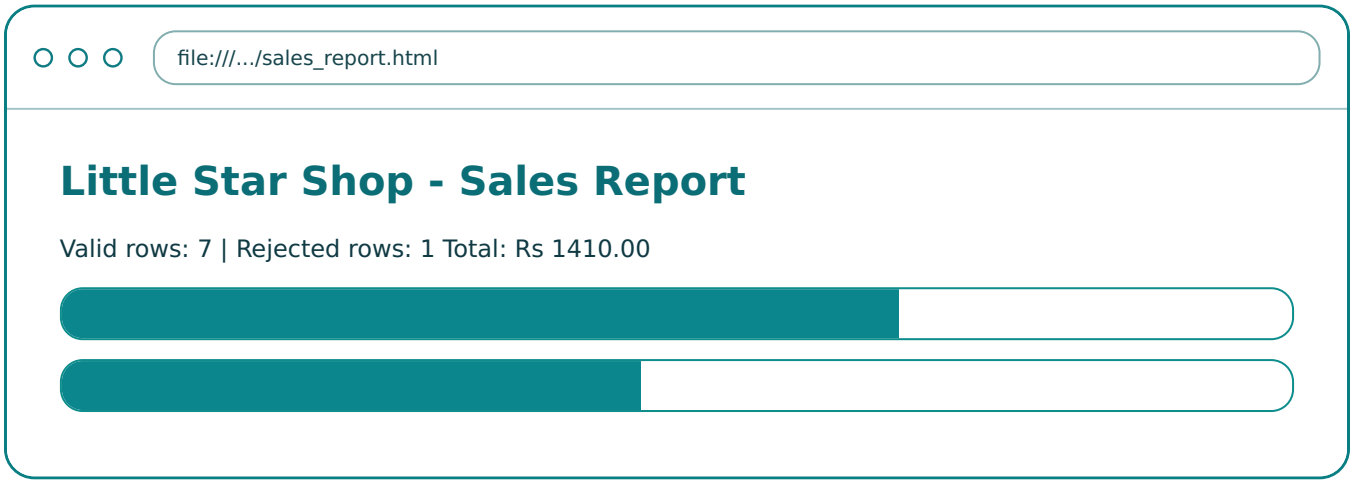
> **ඔබේ වාරය • Try it**
> Notebook value එකේ වැරැද්දක් තිබේද අතින් check කරන්න.

UNIT 06 · LESSON 11 / 13

Chrome HTML report preview

Browser එකෙන් `output/reports/sales_report.html` open කරන්න. මෙය localhost server එකක් නොව local HTML file එකක්. Product rows, total revenue, valid/rejected count පෙන්වයි.

RUN / PREVIEW



මතක තබාගන්න / Expected

පහළ “Revenue is not profit” සටහන බලන්න.

ඔබේ වාරය • Try it

Chrome open කළ විට table එක සම්පූර්ණව පෙනෙනවාද?

UNIT 06 · LESSON 12 / 13

HTML escaping ආරක්ෂාව

CSV එකේ `<script>` වැනි text එකක් product name ලෙස තිබුණොත් එය executable HTML බවට පත් නොවිය යුතුයි. Python `html.escape()` භාවිතයෙන් text safe ලෙස පෙන්විය හැකියි.

RUN / PREVIEW

Visual Studio Code • Python

```
from html import escape
print(escape("<script>"))
```

මතක තබාගන්න / Expected

<script>

ඔබේ වාරය • Try it

HTML preview එකට අපවිත්‍ර input එකක් යොදා බැලීමෙන් QA කිරීම.

UNIT 06 · LESSON 13 / 13

Report results verify කිරීම

Program output එකක් නිවැරදි යැයි අන්ධව විශ්වාස කරන්න එපා. Pencil rows තුන එකතු කර Rs 350ද බලන්න. ඉන්පසු revenue total Rs 1,410ද compare කරන්න. Data analysis එකක math verification අත්‍යවශ්‍යයි.

RUN / PREVIEW

DATA / SAMPLE

100 + 200 + 50 = 350

350 + 690 + 120 + 250 = 1410

මතක තබාගන්න / Expected

Expected output සහ actual output සසඳන්න.

ඔබේ වාරය • Try it

Spreadsheet එකක independent verification එකක් කරන්න.

UNIT 07 · LESSON 01 / 9

Backup කියන්නේ මොකක්ද?

Backup එකක් යනු අහිමි වූ data නැවත ලබාගැනීමට තබාගන්නා වෙනම copy එකක්. ZIP archive එකක් backup එකක් ලෙස භාවිතා කළ හැකි නමුත් එකම drive එකේ විතරක් තිබුණොත් drive failure එකකදී දෙකම අහිමි විය හැකියි.

VISUAL SKETCH



මතක තබාගන්න / Expected

Backup ගැන විශ්වාස කරන්නේ restore test එකෙන් පසුවයි.

ඔබේ වාරය • Try it

Backup folder එක source එකෙන් වෙනම තබන්නේ ඇයි?

UNIT 07 · LESSON 02 / 9

ZipFile code එක

Python `zipfile` module එකෙන් ZIP සාදන්න පුළුවන්. `ZipFile(..., "w")` යනු අලුත් archive එකක් write කරන mode එකයි. අපේ sample source folder එක පමණක් භාවිතා කරමු.

RUN / PREVIEW

Visual Studio Code • Python

```
from zipfile import ZipFile
with ZipFile("demo.zip", "w") as z:
    z.write("data/sales.csv", "sales.csv")
```

මතක තබාගන්න / Expected

demo.zip තුළ sales.csv ඇතුළත් වේ.

ඔබේ වාරය • Try it

Zip output එක extract කර file එක කියවන්න.

UNIT 07 · LESSON 03 / 9

Datetime සහ filename

ඕනෑම backup එකකට අලුත් timestamp එකක් දීමෙන් පරණ backup එක overwrite නොවෙයි. `datetime.now()` හා `strftime()` මඟින් දින සහ වේලා string එකක් සකස් කරමු.

RUN / PREVIEW

Visual Studio Code • Python

```
from datetime import datetime
name = datetime.now().strftime("%Y%m%d_%H%M%S")
print(name)
```

මතක තබාගන්න / Expected

20261031_103000 වැනි value එකක්

ඔබේ වාරය • Try it

ඔබේ local time අනුව value එක වෙනස් වෙයිද?

UNIT 07 · LESSON 04 / 9

Backup command එක

`py app.py backup` command එකෙන් `data/` folder එක ZIP කර `backups/` folder එකේ තබයි. Archive එක තුළ sales.csv තියෙනවාද බලන්න. “Backup created” දැකීම පමණක් සෑහෙන්නේ නැහැ.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py backup
```

මතක තබාගන්න / Expected

Backup: backups/little_star_backup_...zip

ඔබේ වාරය • Try it

ZIP එක Extract කර sales.csv අරින්න.

UNIT 07 · LESSON 05 / 9

Archive validation

`ZipFile.testzip()` එකෙන් archive member එකක් corrupt ද කියලා මූලික check එකක් කරන්න පුළුවන්. එය පමණක් real-world recovery guarantee එකක් නොවේ; සැබෑ restore test එකත් කරන්න.

RUN / PREVIEW

Visual Studio Code • Python

```
from zipfile import ZipFile
with ZipFile("demo.zip") as z:
    print(z.testzip())
```

මතක තබාගන්න / Expected

None යනු corrupt member එකක් හමු නොවූ බවයි.

ඔබේ වාරය • Try it

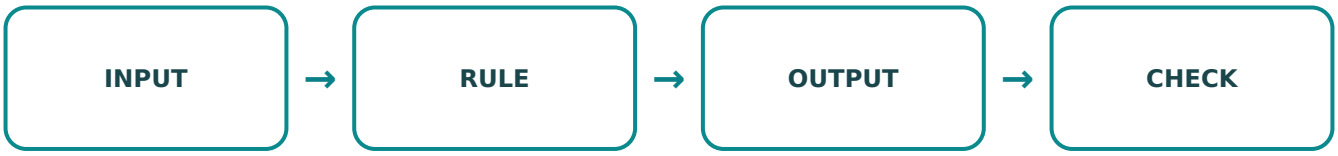
අලුත් ZIP එක restore කරන පියවර ලියන්න.

UNIT 07 · LESSON 06 / 9

3-2-1 backup නියමය

වැදගත් data සඳහා copies 3ක්, media වර්ග 2ක්, offsite copy 1ක් කියන 3-2-1 මූලධර්මය ජනප්‍රියයි. අපේ sample project එකේ ZIP එක මේ නියමය සම්පූර්ණ කරනවා කියලා නොසිතන්න.

VISUAL SKETCH



මතක තබාගන්න / Expected

School project එකටත් sample data loss ඉගෙනීම වැදගත්.

ඔබේ වාරය • Try it

ඔබේ project එකට offsite backup එකක් තැබිය හැකි ආරක්ෂිත තැන කුමක්ද?

UNIT 07 · LESSON 07 / 9

Restore rehearsal

Backup එක extract කර වෙනම `restore_test` folder එකක file එක හොඳින් open වෙනවාද බලන්න. Data එක සසඳා තහවුරු කරන්න. Live project එකේ original file එක වෙනුවට test backup එකක් දමන්න එපා.

VISUAL SKETCH



මතක තබාගන්න / Expected

Restore කිරීම අත්හදා බැලීම = හොඳ QA practice එකක්.

ඔබේ වාරය • Try it

තවම backup තිබුණත් restore fail විය හැකි හේතු දෙකක්.

UNIT 07 · LESSON 08 / 9

Backup privacy සහ encryption

ZIP file එක සාමාන්‍යයෙන් password-protected නොවෙයි. Private customer details එකතු නොකරන්න. Production systems වල storage permissions, encrypted backups සහ retention policy අවශ්‍ය විය හැකියි.

RUN / PREVIEW

DATA / SAMPLE

Demo only: made-up sales data
Never: real passwords in ZIP

මතක තබාගන්න / Expected

Security කියන්නේ coding එකට පසුව කරන අමතර වැඩක් නොවේ.

ඔබේ වාරය • Try it

සැබෑ customer email list ZIP කරන්න කලින් සලකන්න ඕන දේවල්?

UNIT 07 · LESSON 09 / 9

Backup repetition සහ storage

දිනපතා ZIP සෑදුවොත් disk space වැඩි වෙයි. Retention policy එකකින් කුමන copies තබාගන්නවාද යන්න සැලසුම් කරන්න. අපේ learning app එක backup ස්වයංක්‍රීයව delete කරන්නේ නැහැ.

VISUAL SKETCH

FILE EXPLORER • folder preview

```
Day 1 → backup_1.zip
Day 2 → backup_2.zip
Day 3 → backup_3.zip
```

මතක තබාගන්න / Expected

Delete කිරීමට වෙනම approval එකක් අවශ්‍යයි.

ඔබේ වාරය • Try it

Backup storage පිරුණොත් notification එකක් දීමේ වාසිය?

UNIT 08 · LESSON 01 / 8

Scheduled automation කියන්නේ?

සෑම දිනකම පෙ.ව. 8ට report එකක් හදන්න ඕනනම් Windows Task Scheduler එකට trigger එකක් දිය හැකියි. Automation එක run වීම computer on, account permissions සහ task settings මත රඳා පවතියි.

VISUAL SKETCH



මතක තබාගන්න / Expected

පළමු වර manual test එක සාර්ථක කර පසුව schedule කරන්න.

ඔබේ වාරය • Try it

Computer off නම් task එක නියමිත වෙලාවට run වෙයිද?

UNIT 08 · LESSON 02 / 8

`where python` මඟින් path සොයමු

Task Scheduler එකට Python executable එකේ සම්පූර්ණ path එක අවශ්‍ය විය හැකියි. Windows CMD හි `where python` භාවිතා කරන්න. Store alias එකක් හමු වුණොත් actual installed python.exe එක සොයාගන්න.

RUN / PREVIEW

Windows • Command Prompt

```
where python
where py
```

මතක තබාගන්න / Expected

C:\Users\...\python.exe වැනි path එකක්.

ඔබේ වාරය • Try it

ඔබේ පරිගණකයේ result එක ලියාගන්න.

UNIT 08 · LESSON 03 / 8

Create Basic Task පියවර

Start menu > Task Scheduler > Create Basic Task වෙත යන්න. Name: Little Star Daily Report. Trigger: Daily. Action: Start a program. ගුරුවරයාගේ/වැඩිහිටියෙකුගේ අවසරය ඇතිව පමණක් මෙය කරන්න.

RUN / PREVIEW

file:///.../sales_report.html

Name: Little Star Daily Report

Trigger: Daily Action: Start a program

මතක තබාගන්න / Expected

මෙම පියවර සිසුන් සඳහා optional exercise එකක්.

ඔබේ වාරය • Try it

Task Scheduler නම පැහැදිලි වීමේ වාසිය?

UNIT 08 · LESSON 04 / 8

Program / Arguments / Start in

Program/script field එකට full `python.exe` path එක. Add arguments field එකට quote කළ app.py path එක සහ `report`. Start in field එකට project folder path එක. PC අනුව paths වෙනස් වේ.

RUN / PREVIEW

DATA / SAMPLE

Program: C:\...\python.exe
Arguments: "C:\...\app.py" report
Start in: C:\...\Project

මතක තබාගන්න / Expected

Path වල spaces තිබේ නම් quotes නිවැරදිද බලන්න.

ඔබේ වාරය • Try it

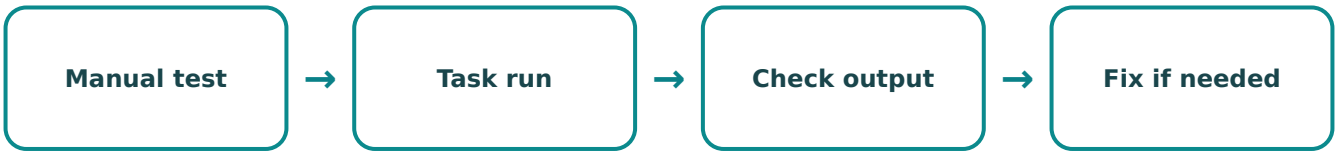
Program field එකට Python path වෙනුවට script path පමණක් දිය හැකිද?

UNIT 08 · LESSON 05 / 8

Task එක පරීක්ෂා කිරීම

Task එක save කර right click > Run භාවිතා කරන්න. Last Run Result, output report timestamp සහ report content බලන්න. Run වූවා කියන status එක තිබුණත් output එක verify කරන්න.

VISUAL SKETCH



මතක තබාගන්න / Expected

Scheduled runs ද QA අවශ්‍යයි.

ඔබේ වාරය • Try it

Report එක අලුතින් සෑදී තිබේද කියලා සොයන්න.

UNIT 08 · LESSON 06 / 8

Task disable කරන හැටි

වැරදි output එනවා නම් Task Scheduler හි task එක Disable කරන්න. Production process එකක් නැවත නැවත fail වුණත් automatic run වීම නතර කිරීම වැදගත්. මුලින් safe report task එක පමණක් schedule කරන්න.

RUN / PREVIEW

file:///.../sales_report.html

Task Scheduler → Task → Disable

Sample browser preview



මතක තබාගන්න / Expected

Control නැති automation එකක් අවදානම්.

ඔබේ වාරය • Try it

ඔබ Task එක නතර කිරීමට පියවර 2ක් ලියන්න.

UNIT 08 · LESSON 07 / 8

Automation logs ගැන හැඳින්වීම

වැඩේ කවදා run වුණාද, error ආවද කියලා logs හොඳයි. අපේ beginner demo එක CMD වෙත messages print කරනවා; advanced version එකකට Python `logging` module එකක් එකතු කළ හැකියි.

RUN / PREVIEW

Visual Studio Code • Python

```
import logging
logging.basicConfig(level=logging.INFO)
logging.info("Report started")
```

මතක තබාගන්න / Expected

Logs එකකට passwords හෝ personal data නොලියන්න.

ඔබේ වාරය • Try it

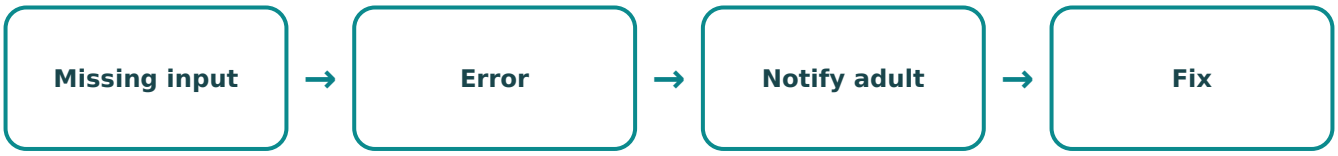
Timestamp සමඟ log එකක් උදව් වෙන්නේ කොහොමද?

UNIT 08 · LESSON 08 / 8

Failure වෙනුවෙන් සැලැස්මක්

Input CSV එක නොමැති නම් script එක fail වෙන්න පුළුවන්. Scheduler එකේ Last Run Result හෝ log බලන්න. Error messages සඟවා `success` කියලා පෙන්වීම හොඳ නැහැ. Fix කර පසුව rerun කරන්න.

VISUAL SKETCH



මතක තබාගන්න / Expected

ප්‍රතිඵල පරීක්ෂාව නැති schedule එකකින් හානි විය හැකියි.

ඔබේ වාරය • Try it

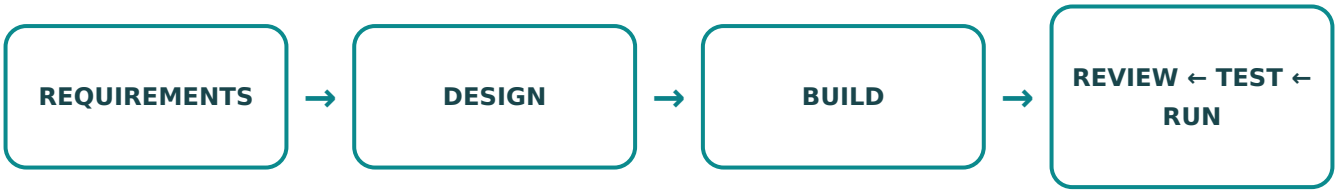
File missing scenario එකකට recovery steps ලියන්න.

UNIT 09 · LESSON 01 / 12

SDLC කියන්නේ?

Software Development Life Cycle එක requirements සිට design, code, testing, release සහ maintenance දක්වා software එකක් නිර්මාණය කරන ක්‍රියාවලියයි. කුඩා Python automation එකකටත් මේ පියවර වැදගත්.

VISUAL SKETCH



මතක තබාගන්න / Expected

Step එකක් skip කිරීමෙන් bug වැඩි විය හැකියි.

ඔබේ වාරය • Try it
අපේ app එකේ Requirement එකක් ලියන්න.

UNIT 09 · LESSON 02 / 12

SRS ලියන හැටි

Software Requirements Specification (SRS) එකට “App එක කරන්න ඕන මොකක්ද?” කියන පැහැදිලි පිළිතුර ලියනවා. අපේ docs/SRS.md එකේ FR01 Preview, FR02 Copy, FR05 CSV+HTML reports වැනි requirements තියෙනවා.

RUN / PREVIEW

DATA / SAMPLE

FR01: Preview without change
FR02: Copy, do not delete original
FR05: Create CSV and HTML reports

මතක තබාගන්න / Expected

Requirement එක test කළ හැකි වාක්‍යයක් විය යුතුයි.

ඔබේ වාරය • Try it

“System එක ලස්සන විය යුතුයි” testable ලෙස නැවත ලියන්න.

UNIT 09 · LESSON 03 / 12

Functional vs nonfunctional

Functional requirement එකක් app එක කරන වැඩ විස්තර කරයි. Nonfunctional requirement එකක් performance, compatibility, usability, privacy වැනි quality එකක් විස්තර කරයි.

RUN / PREVIEW

DATA / SAMPLE

Functional: Make a CSV report
Nonfunctional: Runs offline on Python 3.12+

මතක තබාගන්න / Expected

අවශ්‍යතා දෙවර්ගය වෙනස් කර සටහන් කරන්න.

ඔබේ වාරය • Try it

No external packages යන්න කුමන category එකටද?

UNIT 09 · LESSON 04 / 12

Test case එකක්

Test case එකක් තුළ ID, precondition, steps, expected result තිබෙනවා. “Preview → no folder created” කියන test එක automated සහ manual ක්‍රම දෙකටම කළ හැකියි.

RUN / PREVIEW

DATA / SAMPLE

TC01 | Action: py app.py preview
Expected: plan shown; nothing copied
Actual: _____
Result: PASS / FAIL

මතක තබාගන්න / Expected

Test එක නැවත කළ හැකි ලෙස ලියන්න.

ඔබේ වාරය • Try it

TC02 සඳහා expected result එක ලියන්න.

UNIT 09 · LESSON 05 / 12

Positive testing

නිවැරදි data එකක් දී program එකක් අපේක්ෂිත output ලබාදෙනවාද බලන පරීක්ෂාව positive test කියනවා. 2 pencils × Rs 50 = Rs 100 යන්න උදාහරණයක්.

RUN / PREVIEW

DATA / SAMPLE

Input: Pencil, 2, 50.00

Expected revenue: 100.00

මතක තබාගන්න / Expected

Expected result කලින්ම ලියා තිබීම වැදගත්.

ඔබේ වාරය • Try it

Notebook 2×230 සඳහා test එක සකස් කරන්න.

UNIT 09 · LESSON 06 / 12

Negative testing

වැරදි data එකක් දී program එක ආරක්ෂිතව ප්‍රතිචාර දක්වනවාද බලමු. Quantity 0, missing column, nested folders, duplicate filename වැනි data භාවිතා කරන්න.

RUN / PREVIEW

DATA / SAMPLE

Input: quantity = 0

Expected: reject row + warning

Not expected: quietly report a sale

මතක තබාගන්න / Expected

Validation rules negative tests මඟින් ඔප්පු වෙනවා.

ඔබේ වාරය • Try it

Negative test දෙකක් ලියන්න.

UNIT 09 · LESSON 07 / 12

Automated unit tests

Python සමඟ ඇති `unittest` framework එකෙන් functions වෙන වෙනම පරීක්ෂා කළ හැකියි. Project එකේ category, collisions, report validation, HTML escaping සහ backup behavior සඳහා tests තියෙනවා.

RUN / PREVIEW

Windows • Command Prompt

```
py -m unittest discover -s tests -v
```

මතක තබාගන්න / Expected

9 tests සාර්ථකව run වෙයි (provided source).

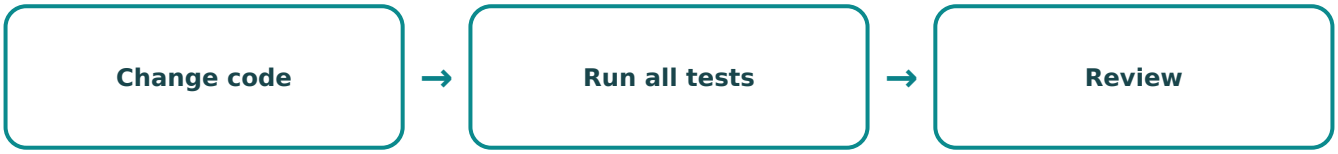
> **ඔබේ වාරය • Try it**
> තවත් test එකක් එකතු කළොත් කුමන feature එකටද?

UNIT 09 · LESSON 08 / 12

Regression testing

වැඩ කරන feature එකක් වෙනස් කළාම පරණ feature බිඳී නැද්ද බලන්න සම්පූර්ණ tests නැවත run කරන එක regression testing. `.webp` category mapping වෙනස් කළාට පසුව report tests පවා run කරමු.

VISUAL SKETCH



මතක තබාගන්න / Expected

One small change can affect other parts.

ඔබේ වාරය • Try it

New extension එකක් එකතු කළ පසු tests list එකක් ලියන්න.

UNIT 09 · LESSON 09 / 12

Bug report ලියමු

Bug report එකකට steps to reproduce, expected, actual සහ severity ඇතුළත් කරන්න. හොඳ bug report එකක් වෙනත් developer කෙනෙකුට ගැටලුව නැවත දැකගන්න උදව් කරනවා.

RUN / PREVIEW

DATA / SAMPLE

BUG-01: Wrong category
Steps: Add test.webp, run preview
Expected: Images
Actual: Other
Severity: Medium

මතක තබාගන්න / Expected

උදාහරණයක් පමණයි; project mapping එකේ .webp තියෙනවා.

ඔබේ වාරය • Try it

ඔබේම imaginary bug report එකක් ලියන්න.

UNIT 09 · LESSON 10 / 12

QA vs Testing

Testing යනු bugs හඳුනාගන්නා ක්‍රියාවලියක්. QA යනු requirements, process, reviews, testing සහ improvement ඇතුළු quality සඳහා වූ පුළුල් වැඩපිළිවෙළක්. Developer සහ QA දෙපාර්ශ්වයටම ගුණාත්මක බව වැදගත්.

VISUAL SKETCH



මතක තබාගන්න / Expected

QA කියන්නේ test case run කිරීම පමණක් නොවේ.

> **ඔබේ වාරය • Try it**
> SRS review එක QA වැඩක්ද? ඇයි?

UNIT 09 · LESSON 11 / 12

Release checklist

Project එක බෙදාදීමට පෙර README කියවිය හැකිද, files අවශ්‍ය සියල්ල ඇතුළත්ද, tests pass ද, outputs නිවැරදිද, ZIP extract වෙයිද බලන්න. Real data හෝ secrets accidentally ZIP එකට ඇතුළත් නොවිය යුතුයි.

RUN / PREVIEW

DATA / SAMPLE

- ☐ Code runs
- ☐ Tests pass
- ☐ Samples included
- ☐ README explains commands
- ☐ No private information

මතක තබාගන්න / Expected

Documentation එකක් නැති source ZIP එකක් නවකයෙකුට අපහසුයි.

ඔබේ වාරය • Try it

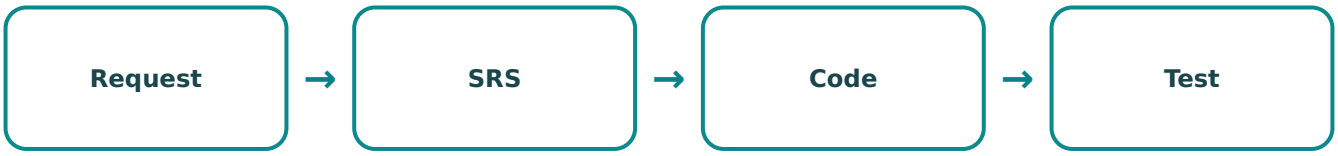
ඔබේම acceptance checklist එක සකස් කරන්න.

UNIT 09 · LESSON 12 / 12

Maintenance සහ change requests

Users නව file type එකක් support කරන්න ඉල්ලුවොත් requirements update කර design බලලා code/test/doc එකට වෙනස්කම් දාන්න. නව feature එකක් කියන්නේ code line එකක් පමණක් නොවේ.

VISUAL SKETCH



මතක තබාගන්න / Expected

Change එක end-to-end කළ යුතුයි.

ඔබේ වාරය • Try it

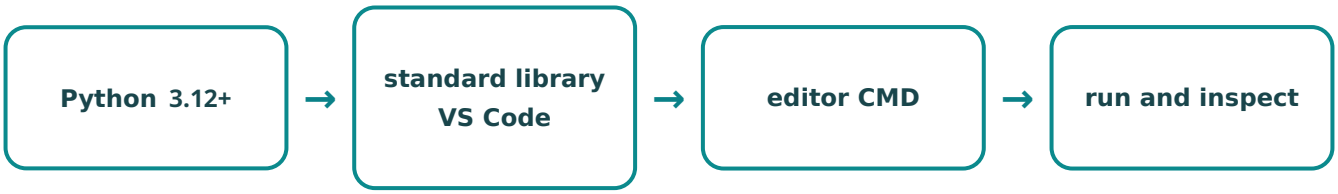
`.mp3` category එකක් එකතු කරන roadmap එක ලියන්න.

UNIT 10 · LESSON 01 / 12

Python 2026 ගැන දැනගමු

2026 ඔක්තෝබර් මස Python 3.15 stable release එකක් ලෙස තිබෙනවා. හැබැයි මේ පොතට විශේෂ අලුත් language features අවශ්‍ය නැහැ. පරණ, ප්‍රසිද්ධ pathlib/csv/zipfile APIs භාවිතයෙන් සරල හා විශ්වාසදායක code ලියමු.

VISUAL SKETCH



මතක තබාගන්න / Expected

Version එකට වඩා code තේරුම්ගැනීම වැදගත්.

ඔබේ වාරය • Try it

ඔබ භාවිතා කරන Python version එක ලියන්න.

UNIT 10 · LESSON 02 / 12

Automation vs Machine Learning

Rule-based organizer එක fixed extensions අනුව decision ගනියි. Machine Learning system එක data examples වලින් patterns ඉගෙනගනියි. දෙකම ප්‍රයෝජනවත් නමුත් සෑම automation task එකකටම ML අවශ්‍ය නැහැ.

RUN / PREVIEW

DATA / SAMPLE

Automation: if .png → Images
ML: train on examples → prediction

මතක තබාගන්න / Expected

කුඩා, පැහැදිලි rules ඇති task එකකට plain Python හොඳයි.

ඔබේ වාරය • Try it

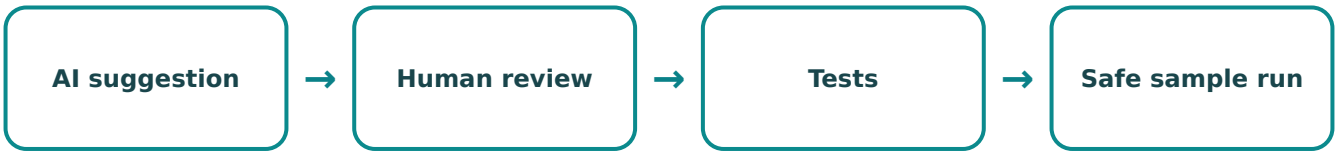
Auto email classifier එකකට ML කවදා උපකාරී වේද?

UNIT 10 · LESSON 03 / 12

2026 AI coding assistants

AI tools code suggestions දීමට උපකාරී විය හැකියි. නමුත් එම code එක filenames delete/move කරනවාද, internet වෙත data යවනවාද කියලා මිනිසා review කළ යුතුයි. පළමුව dry run සහ tests කරන්න.

VISUAL SKETCH



මතක තබාගන්න / Expected

AI-generated automation එකක් blind run නොකරන්න.

ඔබේ වාරය • Try it

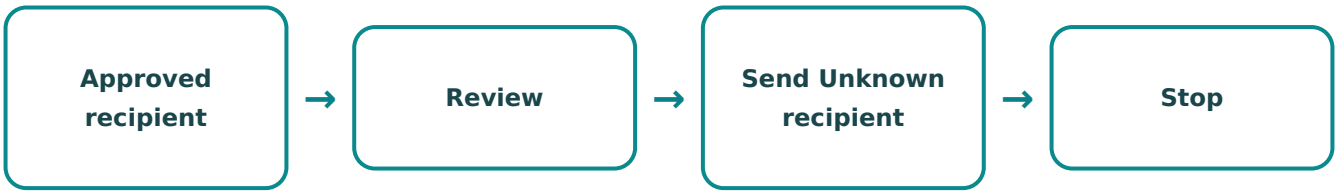
AI එකක් දුන් script එකක් run කිරීමට පෙර checks තුනක්.

UNIT 10 · LESSON 04 / 12

Email automation සීමා

පසුව reminders යැවීම ඉගෙනගන්න පුළුවන්. නමුත් credentials, opt-in, mail limits සහ privacy සැලකිය යුතුයි. මේ beginner project එක email නොයවයි. Unsolicited mass mailing/harvesting සඳහා automation නොකරන්න.

VISUAL SKETCH



මතක තබාගන්න / Expected

Permission නොමැති පුද්ගල තොරතුරු process නොකරන්න.

ඔබේ වාරය • Try it

Permission සහ privacy කියන්නේ මොකක්ද?

UNIT 10 · LESSON 05 / 12

Web automation සීමා

Browser automation වෙබ් pages වල buttons click කිරීම වගේ වැඩට උපකාරී. නමුත් robots rules, site terms, rate limits සහ accounts ගැන සැලකිලිමත් වන්න. මේ පොතේ internet requests හෝ scraping නැහැ.

RUN / PREVIEW

DATA / SAMPLE

Public documentation → Read

Private accounts → Authorized only

මතක තබාගන්න / Expected

නීති සහ permissions තේරුම්ගැනීමත් engineering skill එකක්.

ඔබේ වාරය • Try it

Login details share නොකළ යුත්තේ ඇයි?

UNIT 10 · LESSON 06 / 12

Privacy by design

Sample CSV එකේ නම, email, phone නොව products සහ prices පමණයි. සැබෑ customer data භාවිතා කරන විට minimum necessary data පමණක් රැස්කරන්න. Logs/ZIP files වල sensitive values නොදමන්න.

VISUAL SKETCH



මතක තබාගන්න / Expected

පුහුණු වීමට synthetic data ප්‍රමාණවත්.

ඔබේ වාරය • Try it

Phone number නොමැති sample order schema එකක් යෝජනා කරන්න.

UNIT 10 · LESSON 07 / 12

Mini challenge: Morning report

දිනපතා දත්ත summary එකක් අවශ්‍ය shop එකකට ඔබ developer කියලා හිතන්න. SRS තුළ date, sample CSV, report file සහ invalid rows behavior සටහන් කරලා manual run එක පෙන්වන්න.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py report
```

මතක තබාගන්න / Expected

Output report Chrome එකෙන් කියවන්න.

> **ඔබේ වාරය • Try it**
> Automation එක user requirement එකට ගැළපේද කියලා චෙක් කරන්න.

UNIT 10 · LESSON 08 / 12

Mini challenge: Classroom files

Teacher කෙනෙක් photos සහ homework files වර්ග කිරීමට ඉල්ලනවා. Code changes නැතිව demo_inbox තුළ files හතරක් දාලා preview කරන්න. Output paths සත්‍යද බලන්න. Private student files නොයොදන්න.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py preview
py app.py organize
```

මතක තබාගන්න / Expected

Sample input පමණක් පාවිච්චි කරන්න.

ඔබේ වාරය • Try it

Images කොපමණ copy වුණාද ගණන් කරන්න.

UNIT 10 · LESSON 09 / 12

Mini challenge: Backup restore

Zip backup එකක් සෑදූ පසු restore_test folder එකට extract කරන්න. Source CSV සහ restored CSV line count compare කරන්න. Backup policy එකට “monthly test” අංගයක් එක් කරන්න.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py backup
# Then extract ZIP manually
```

මතක තබාගන්න / Expected

Backup successful message එකට වඩා recovery test එක වටිනවා.

ඔබේ වාරය • Try it

Restore test result එක PASS/FAIL ලෙස සටහන් කරන්න.

UNIT 10 · LESSON 10 / 12

Mini challenge: Error hunt

CSV එකේ quantity field එකට abc දමා report run කරන්න. Warning row එක හොයාගන්න. පසුව sample data මුල් තත්ත්වයට යළි සකස් කරන්න. Validation මඟින් incorrect output වළක්වාගන්නා විදිහ පැහැදිලි කරන්න.

RUN / PREVIEW

DATA / SAMPLE

```
2026-10-06,Pencil,abc,50.00
```

මතක තබාගන්න / Expected

Row rejected; report error ලැයිස්තුවේ දැක්වේ.

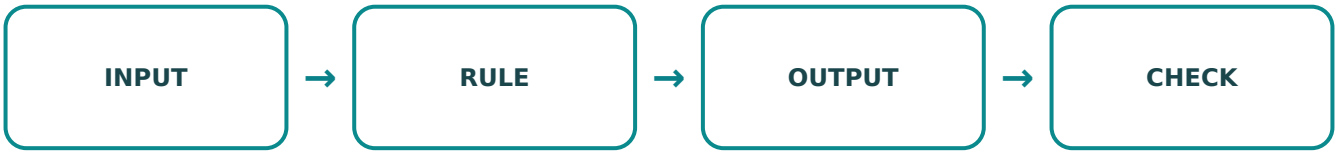
> **ඔබේ වාරය • Try it**
> Expected සහ actual results දෙක ලියන්න.

UNIT 10 · LESSON 11 / 12

Mini project: Requirement interview

මිතුරෙකු automation එකක් ඉල්ලනවා කියලා රඟපෑමක් කරන්න. “මොන files?”, “කොහේද?”, “කවදාද?”, “Delete කරන්නද Copy කරන්නද?”, “වැරදුණොත්?” වැනි ප්‍රශ්න 5ක් අහන්න.

VISUAL SKETCH



මතක තබාගන්න / Expected

Good requirements are discovered through good questions.

ඔබේ වාරය • Try it

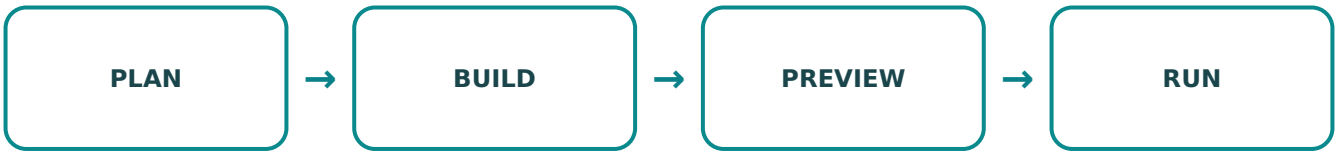
වැඩේට user consent ලැබුණාද කියලා කොහොමද තහවුරු කරන්නේ?

UNIT 10 · LESSON 12 / 12

ඉගෙනගත් දේ මතක තබාගමු

ඔබ මේ වනවිට input, rule, condition, loop, function, file paths, CSV, reports, backup, scheduling, tests, QA සහ safety ගැන ඉගෙනගෙන ඉන්නවා. එකම program එකෙන් මේ සියල්ල සම්බන්ධ වෙනවා.

VISUAL SKETCH



මතක තබාගන්න / Expected

Automation = time saving + checks + responsibility.

ඔබේ වාරය • Try it

ඔබේ අලුත් automation project එකේ මූලික idea එක ලියන්න.

UNIT 11 · LESSON 01 / 14

MCQ 01 | Automation කියන්නේ?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. එකම වැඩේ නිශ්චිත instructions අනුව නැවත කිරීම
B. ගීතයක් record කිරීම

මතක තබාගන්න / Expected

පිළිතුර: A. එකම වැඩේ නිශ්චිත instructions අනුව නැවත කිරීම

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 02 / 14

MCQ 02 | `print()` කරන්නේ?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. Text result පෙන්වීම
B. Folder delete කිරීම

මතක තබාගන්න / Expected

පිළිතුර: A. Text result පෙන්වීම

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 03 / 14

MCQ 03 | `.csv` කියන්නේ?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. Table-like text data

B. Image file

මතක තබාගන්න / Expected

පිළිතුර: A. Table-like text data

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 04 / 14

MCQ 04 | `for` loop එක?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. නැවත කරන වැඩ

B. Computer restart කරන command

මතක තබාගන්න / Expected

පිළිතුර: A. නැවත කරන වැඩ

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 05 / 14

MCQ 05 | `Path.suffix`?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. File extension
B. File content

මතක තබාගන්න / Expected

පිළිතුර: A. File extension

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 06 / 14

MCQ 06 | Dry run?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. මුලින් preview; වෙනස්කමක් නැහැ
B. මුල් file delete කිරීම

මතක තබාගන්න / Expected

පිළිතුර: A. මුලින් preview; වෙනස්කමක් නැහැ

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 07 / 14

MCQ 07 | `shutil.copy2`?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. File copy කිරීම
B. Email යැවීම

මතක තබාගන්න / Expected

පිළිතුර: A. File copy කිරීම

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 08 / 14

MCQ 08 | `Decimal` මුදල් සඳහා?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. Precision control පහසු නිසා
B. Audio play කරන නිසා

මතක තබාගන්න / Expected

පිළිතුර: A. Precision control පහසු නිසා

ඔබේ වාරය • Try it
ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 09 / 14

MCQ 09 | Invalid quantity 0?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. Reject සහ warning

B. Zero revenue sale එකක් ලෙස දැමීම

මතක තබාගන්න / Expected

පිළිතුර: A. Reject සහ warning

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 10 / 14

MCQ 10 | Backup verification?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. ZIP extract / restore පරීක්ෂාව
B. Filename පමණක් බලන්න

මතක තබාගන්න / Expected

පිළිතුර: A. ZIP extract / restore පරීක්ෂාව

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 11 / 14

MCQ 11 | Task Scheduler trigger?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. කාලය හෝ event එකක්

B. Python comments

මතක තබාගන්න / Expected

පිළිතුර: A. කාලය හෝ event එකක්

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 12 / 14

MCQ 12 | QA සහ testing?

පිළිතුර තෝරාගන්න. පසුව ඇයි කියලා වචන දෙක තුනකින් පැහැදිලි කරන්න. තනි ප්‍රශ්නයකට පිළිතුර මතක තබාගැනීමට වඩා ඒක test කර බැලීම වටිනවා.

RUN / PREVIEW

QUESTION / OPTIONS

A. QA පුළුල් ක්‍රියාවලියකි
B. QA කියන්නේ code අලංකාර කිරීමයි

මතක තබාගන්න / Expected

පිළිතුර: A. QA පුළුල් ක්‍රියාවලියකි

ඔබේ වාරය • Try it

ඔබේම තවත් වැරදි පිළිතුරක් (option C) නිර්මාණය කරන්න.

UNIT 11 · LESSON 13 / 14

Final project - Demo Day

Little Star Automation Center project එක අලුතින් Extract කර preview, organize, report, backup ක්‍රමයෙන් run කරන්න. Report එක Chrome වලින් පෙන්වන්න. SRS හි requirements සහ QA checklist එකට actual result ලියන්න.

RUN / PREVIEW

Windows • Command Prompt

```
py app.py preview
py app.py organize
py app.py report
py app.py backup
py -m unittest discover -s tests -v
```

මතක තබාගන්න / Expected

4 copied (first run), 7 valid CSV rows, 1 rejected, 9 tests PASS.

> **ඔබේ වාරය • Try it**
> අවසානයේ ඔබේ demo video එකක් පෞද්ගලිකව record කර බලන්න.

UNIT 11 · LESSON 14 / 14

Certificate checklist

පහත කාර්යයන් ඔබම පරීක්ෂා කරන්න. ආරම්භයේදී කිසිවක් නොදැන සිටියත් මේ steps සියල්ල කරන්න පුළුවන් නම් ඔබ පළමු automation project එක නිම කළා කියලා සලකන්න.

RUN / PREVIEW

DATA / SAMPLE

- ☐ Python version checked
- ☐ Hello script run
- ☐ Preview reviewed
- ☐ Files copied safely
- ☐ Report verified
- ☐ Backup restored
- ☐ Tests run

මතක තබාගන්න / Expected

Certificate ලබාදීම කළමනාකරු/ගුරුවරයාගේ තීරණයයි.

ඔබේ වාරය • Try it

ඔබ සම්පූර්ණ කළ boxes tick කරන්න.

QUICK REFERENCE

Commands Cheat Sheet

Windows CMD එකේ `app.py` ඇති folder එකේ සිට පහත commands run කරන්න. Python launcher එක නැති නම් `py` වෙනුවට `python` යොදාගන්න.

Windows • Command Prompt

```
py --version
py examples\01_hello.py
py app.py preview
py app.py organize
py app.py report
py app.py backup
py app.py all
py -m unittest discover -s tests -v
```

Browser report: output/reports/sales_report.html
Originals remain in demo_inbox. Default `py app.py` opens a simple menu.

PRACTICAL WORK

Sample Data සහ Expected Output

පළමු run එකේ sample files හතරක් copy වෙනවා. Sales data වල rows 8ක් තිබෙන අතර valid 7ක්; quantity 0 ඇති row 1ක් reject වෙනවා. Total revenue Rs 1,410.00. මේ අගයන් test result ලෙස යොදාගන්න.

DATA / SAMPLE

Pencil: Rs 350.00
Notebook: Rs 690.00
Eraser: Rs 120.00
Color Pen: Rs 250.00
Total Revenue: Rs 1410.00

Independent verification

CSV එකේ order values අතින් ගණන් කර report output එකට ගළපන්න. Profit නොව revenue බව මතක තබාගන්න.

FULL SOURCE CODE • app.py

app.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–16 / 69. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • app.py • PART 1

```
"""Little Star Automation Center - safe Python learning project."""
import argparse
from pathlib import Path

from automation.backups import create_backup
from automation.organizer import apply_plan, plan_files
from automation.reports import write_report

ROOT = Path(__file__).resolve().parent
INBOX = ROOT / "demo_inbox"
ORGANIZED = ROOT / "output" / "organized"
REPORTS = ROOT / "output" / "reports"
BACKUPS = ROOT / "backups"
DATA = ROOT / "data" / "sales.csv"


```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • app.py

app.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 17–32 / 69. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • app.py • PART 2

```
 17  def show_plan() -> list:
 18      """Print a plan; nothing is changed."""
 19      plan = plan_files(INBOX, ORGANIZED)
 20      if not plan:
 21          print("Inbox is empty. Add sample files to demo_inbox first.")
 22      for item in plan:
 23          print(f"{item.status:9} | {item.source.name} -> {item.target.parent.name}/
{item.target.name}")
 24      return plan
 25
 26
 27  def run_command(command: str) -> None:
 28      if command == "preview":
 29          show_plan()
 30      elif command == "organize":
 31          plan = show_plan()
 32          print(f"Copied {apply_plan(plan)} files. Originals are unchanged.")
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • app.py

app.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 33–48 / 69. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • app.py • PART 3

```
33      elif command == "report":
34          summary, webpage, errors = write_report(DATA, REPORTS)
35          print(f"Summary: {summary}\nBrowser report: {webpage}")
36          for error in errors:
37              print("Review:", error)
38      elif command == "backup":
39          print("Backup:", create_backup(ROOT / "data", BACKUPS))
40      elif command == "all":
41          run_command("organize")
42          run_command("report")
43          run_command("backup")
44      else:
45          raise ValueError("Unknown command")
46
47
48  def main() -> None:
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • app.py

app.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 49–64 / 69. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • app.py • PART 4

```
 49      parser = argparse.ArgumentParser(description="Safe, beginner-friendly automation demo")
 50      parser.add_argument("command", nargs="?", choices=["preview", "organize", "report", "backup",
"all"])
 51      options = parser.parse_args()
 52      if options.command:
 53          run_command(options.command)
 54          return
 55      print("\n=== LITTLE STAR AUTOMATION CENTER ===")
 56      print("1 Preview file plan (safe)\n2 Copy organized files\n3 Create sales report")
 57      print("4 Backup sample data\n5 Run all\n0 Exit")
 58      choice = input("Choose 0-5: ").strip()
 59      menu = {"1": "preview", "2": "organize", "3": "report", "4": "backup", "5": "all"}
 60      if choice in menu:
 61          run_command(menu[choice])
 62      elif choice == "0":
 63          print("Bye!")
 64      else:
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • app.py

app.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 65–69 / 69. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • app.py • PART 5

```
65          print("Try one of the shown menu numbers.")
66
67
68  if __name__ == "__main__":
69      main()
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/organizer.py

automation/organizer.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–16 / 80. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/organizer.py • PART 1

```
"""Make a safe file organization plan; copy only when requested."""
from dataclasses import dataclass
from pathlib import Path
import hashlib
import shutil

EXTENSIONS = {
    "Images": {".jpg", ".jpeg", ".png", ".gif", ".webp"},
    "Documents": {".pdf", ".docx", ".txt"},
    "Spreadsheets": {".csv", ".xlsx"},
    "Code": {".py", ".html", ".css", ".js"},
}


@dataclass(frozen=True)
class FileAction:
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/organizer.py

automation/organizer.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 17–32 / 80. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/organizer.py • PART 2

```
17      source: Path
18      target: Path
19      status: str  # "copy" or "unchanged"
20
21
22  def category_for(filename: str) -> str:
23      """Use a file's last extension to choose a folder."""
24      extension = Path(filename).suffix.lower()
25      for category, extensions in EXTENSIONS.items():
26          if extension in extensions:
27              return category
28      return "Other"
29
30
31  def same_file_contents(first: Path, second: Path) -> bool:
32      """Compare bytes safely, not names alone."""
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/organizer.py

automation/organizer.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 33–48 / 80. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/organizer.py • PART 3

```
33      if not second.is_file() or first.stat().st_size != second.stat().st_size:
34          return False
35      def digest(path: Path) -> bytes:
36          h = hashlib.sha256()
37          with path.open("rb") as file:
38              for chunk in iter(lambda: file.read(65536), b""):
39                  h.update(chunk)
40          return h.digest()
41      return digest(first) == digest(second)
42
43
44  def plan_files(inbox: Path, destination: Path) -> list[FileAction]:
45      """Preview top-level files, skipping symlinks and directories."""
46      inbox, destination = inbox.resolve(), destination.resolve()
47      if not inbox.is_dir():
48          raise FileNotFoundError(f"Inbox not found: {inbox}")
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/organizer.py

automation/organizer.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 49–64 / 80. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/organizer.py • PART 4

```
 49      if destination == inbox or inbox in destination.parents or destination in inbox.parents:
 50          raise ValueError("Inbox and output must be separate folders, not nested.")
 51      planned: list[FileAction] = []
 52      reserved: set[Path] = set()
 53      for source in sorted(inbox.iterdir(), key=lambda p: p.name.casefold()):
 54          if source.is_symlink() or not source.is_file():
 55              continue
 56          folder = destination / category_for(source.name)
 57          candidate = folder / source.name
 58          count = 2
 59          while candidate in reserved or (candidate.exists() and not same_file_contents(source,
candidate)):
 60              candidate = folder / f"{source.stem}_{count}{source.suffix}"
 61              count += 1
 62          status = "unchanged" if candidate.exists() and same_file_contents(source, candidate) else
"copy"
 63          planned.append(FileAction(source, candidate, status))
 64          reserved.add(candidate)
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/organizer.py

automation/organizer.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 65–80 / 80. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/organizer.py • PART 5

```
65      return planned
66
67
68  def apply_plan(plan: list[FileAction]) -> int:
69      """Copy files, never delete or move the originals."""
70      copied = 0
71      for item in plan:
72          if item.status != "copy":
73              continue
74          item.target.parent.mkdir(parents=True, exist_ok=True)
75          # A newly appearing target is not overwritten.
76          if item.target.exists():
77              raise FileExistsError(f"Refusing to overwrite: {item.target}")
78          shutil.copy2(item.source, item.target)
79          copied += 1
80      return copied
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/reports.py

automation/reports.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–16 / 63. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/reports.py • PART 1

```
"""Create a clean report from made-up sales data."""
from collections import defaultdict
from decimal import Decimal, InvalidOperation
from html import escape
from pathlib import Path
import csv

REQUIRED = {"date", "product", "quantity", "unit_price"}


def calculate_sales(source: Path) -> tuple[dict[str, Decimal], list[str], int]:
    """Read and validate sales; keep rejected row numbers for review."""
    totals: dict[str, Decimal] = defaultdict(lambda: Decimal("0.00"))
    errors: list[str] = []
    valid_count = 0
    with source.open("r", newline="", encoding="utf-8-sig") as stream:
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/reports.py

automation/reports.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 17–32 / 63. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/reports.py • PART 2

```
17          reader = csv.DictReader(stream)
18          if not reader.fieldnames or not REQUIRED.issubset(reader.fieldnames):
19              raise ValueError("CSV needs date, product, quantity, unit_price columns")
20          for number, row in enumerate(reader, start=2):
21              try:
22                  name = (row["product"] or "").strip()
23                  quantity = int(row["quantity"] or "")
24                  price = Decimal(row["unit_price"] or "")
25                  if not name or not (row["date"] or "").strip():
26                      raise ValueError("Missing date or product")
27                  if quantity <= 0 or not price.is_finite() or price < 0:
28                      raise ValueError("Invalid quantity or price")
29                  totals[name] += quantity * price
30                  valid_count += 1
31              except (ValueError, TypeError, InvalidOperation) as exc:
32                  errors.append(f"Row {number}: {exc}")
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/reports.py

automation/reports.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 33–48 / 63. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/reports.py • PART 3

```
33      return dict(totals), errors, valid_count
34
35
36  def write_report(input_csv: Path, report_dir: Path) -> tuple[Path, Path, list[str]]:
37      """Save a summary CSV and browser-viewable HTML report."""
38      totals, errors, count = calculate_sales(input_csv)
39      report_dir.mkdir(parents=True, exist_ok=True)
40      csv_path = report_dir / "sales_summary.csv"
41      html_path = report_dir / "sales_report.html"
42      with csv_path.open("w", newline="", encoding="utf-8") as output:
43          writer = csv.writer(output)
44          writer.writerow(["product", "revenue_lkr"])
45          for name, total in sorted(totals.items()):
46              writer.writerow([name, f"{total:.2f}"])
47      rows = "\n".join(
48          f"<tr><td>{escape(name)}</td><td>Rs {total:.2f}</td></tr>"
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/reports.py

automation/reports.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 49–63 / 63. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/reports.py • PART 4

```
 49          for name, total in sorted(totals.items())
 50      )
 51      total_revenue = sum(totals.values(), Decimal("0.00"))
 52      html = f'''<!DOCTYPE html><html lang="en"><head><meta charset="utf-8">
 53  <title>Little Star Sales Report</title><style>
 54  body{{font:18px Arial;max-width:750px;margin:35px auto;color:#123b42;background:white}}
 55  table{{border-collapse:collapse;width:100%}}th,td{{border:1px solid #9bb;padding:12px;text-
align:left}}
 56  th{{color:#076975}}strong{{color:#087b80}}
 57  </style></head><body><h1>Little Star Shop - Sales Report</h1>
 58  <p>Valid rows: {count} | Rejected rows: {len(errors)}</p>
 59  <table><thead><tr><th>Product</th><th>Revenue (LKR)</th></tr></thead>
 60  <tbody>{rows}</tbody></table><h2>Total: <strong>Rs {total_revenue:.2f}</strong></h2>
 61  <p>Sample data only - revenue is not profit.</p></body></html>'''
 62      html_path.write_text(html, encoding="utf-8")
 63      return csv_path, html_path, errors
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/backups.py

automation/backups.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–16 / 26. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/backups.py • PART 1

```
"""Backup folders into a timestamped ZIP archive."""
from datetime import datetime
from pathlib import Path
from zipfile import ZIP_DEFLATED, ZipFile


def create_backup(folder: Path, output_dir: Path) -> Path:
    """Create a backup without following folder symlinks."""
    folder = folder.resolve()
    output_dir = output_dir.resolve()
    if not folder.is_dir():
        raise FileNotFoundError(f"Folder not found: {folder}")
    if output_dir == folder or folder in output_dir.parents:
        raise ValueError("Keep backup output outside the source folder.")
    output_dir.mkdir(parents=True, exist_ok=True)
    stamp = datetime.now().strftime("%Y%m%d_%H%M%S_%f")
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • automation/backups.py

automation/backups.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 17–26 / 26. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • automation/backups.py • PART 2

```
 17      target = output_dir / f"little_star_backup_{stamp}.zip"
 18      with ZipFile(target, "w", ZIP_DEFLATED) as archive:
 19          for file in sorted(folder.rglob("*")):
 20              if file.is_file() and not file.is_symlink() and not any(p.is_symlink() for p in
file.parents if p != folder.parent):
 21                  archive.write(file, arcname=file.relative_to(folder))
 22      with ZipFile(target) as check:
 23          broken = check.testzip()
 24          if broken:
 25              raise IOError(f"Backup failed validation: {broken}")
 26      return target
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • examples/01_hello.py

examples/01_hello.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–2 / 2. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • examples/01_hello.py • PART 1

```
1  # Python says hello to a new programmer.
2  print("Hello, little automation hero!")
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • examples/02_variables.py

examples/02_variables.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–4 / 4. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • examples/02_variables.py • PART 1

```
1  # A variable is a named box for storing data.
2  shop_name = "Little Star Shop"
3  orders = 3
4  print(shop_name, "processed", orders, "orders")
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • examples/03_loops.py

examples/03_loops.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–4 / 4. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • examples/03_loops.py • PART 1

```
1  # Repeat a task for each filename.
2  files = ["drawing.png", "notes.txt", "sales.csv"]
3  for filename in files:
4      print("Check:", filename)
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • examples/04_functions.py

examples/04_functions.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–5 / 5. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • examples/04_functions.py • PART 1

```
1  # A function is a reusable mini-machine.
2  def greet_customer(name):
3      return "Hello, " + name + "!"
4
5  print(greet_customer("Nila"))
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • examples/05_pathlib.py

examples/05_pathlib.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–6 / 6. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • examples/05_pathlib.py • PART 1

```
1  # Show file names without changing files.
2  from pathlib import Path
3  folder = Path(__file__).resolve().parent.parent / "demo_inbox"
4  for file in folder.iterdir():
5      if file.is_file():
6          print(file.name, file.suffix.lower())
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • examples/06_csv_read.py

examples/06_csv_read.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–7 / 7. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • examples/06_csv_read.py • PART 1

```
1  # Read the sample sales spreadsheet as text data.
2  from pathlib import Path
3  import csv
4  file = Path(__file__).resolve().parent.parent / "data" / "sales.csv"
5  with file.open(newline="", encoding="utf-8") as stream:
6      for row in csv.DictReader(stream):
7          print(row["product"], row["quantity"])
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • examples/07_zip_basics.py

examples/07_zip_basics.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–7 / 7. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • examples/07_zip_basics.py • PART 1

```
1  # Create a tiny ZIP from safe example files.
2  from pathlib import Path
3  from zipfile import ZipFile
4  folder = Path(__file__).resolve().parent.parent
5  with ZipFile(folder / "tiny_demo.zip", "w") as archive:
6      archive.write(folder / "data" / "sales.csv", "sales.csv")
7  print("tiny_demo.zip created")
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • tests/test_automation.py

tests/test_automation.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 1–15 / 97. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • tests/test_automation.py • PART 1

```
"""Run: python -m unittest discover -s tests -v"""
from decimal import Decimal
from pathlib import Path
from tempfile import TemporaryDirectory
from zipfile import ZipFile
import unittest

from automation.organizer import category_for, plan_files, apply_plan
from automation.reports import calculate_sales, write_report
from automation.backups import create_backup


class OrganizerTests(unittest.TestCase):
    def test_categories(self):
        self.assertEqual(category_for("photo.PNG"), "Images")
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • tests/test_automation.py

tests/test_automation.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 16–30 / 97. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • tests/test_automation.py • PART 2

```
16          self.assertEqual(category_for("budget.csv"), "Spreadsheets")
17          self.assertEqual(category_for("mystery.zzz"), "Other")
18
19      def test_preview_does_not_copy_and_apply_does(self):
20          with TemporaryDirectory() as tmp:
21              base = Path(tmp)
22              inbox, output = base / "inbox", base / "out"
23              inbox.mkdir()
24              (inbox / "story.txt").write_text("hi")
25              plan = plan_files(inbox, output)
26              self.assertFalse(output.exists())
27              self.assertEqual(apply_plan(plan), 1)
28              self.assertEqual((output / "Documents" / "story.txt").read_text(), "hi")
29              self.assertTrue((inbox / "story.txt").exists())
30              self.assertEqual(apply_plan(plan_files(inbox, output)), 0)
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • tests/test_automation.py

tests/test_automation.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 31–45 / 97. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • tests/test_automation.py • PART 3

```
31
32      def test_name_collision_never_overwrites(self):
33          with TemporaryDirectory() as tmp:
34              base = Path(tmp)
35              inbox = base / "inbox"; output = base / "output"
36              inbox.mkdir(); (output / "Documents").mkdir(parents=True)
37              (inbox / "note.txt").write_text("new")
38              (output / "Documents" / "note.txt").write_text("old")
39              actions = plan_files(inbox, output)
40              self.assertEqual(actions[0].target.name, "note_2.txt")
41              apply_plan(actions)
42              self.assertEqual((output / "Documents" / "note.txt").read_text(), "old")
43
44      def test_nested_folders_disallowed(self):
45          with TemporaryDirectory() as tmp:
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • tests/test_automation.py

tests/test_automation.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 46–60 / 97. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • tests/test_automation.py • PART 4

```
 46              base = Path(tmp); inbox = base / "inbox"; inbox.mkdir()
 47              with self.assertRaises(ValueError):
 48                  plan_files(inbox, inbox / "out")
 49
 50
 51  class ReportTests(unittest.TestCase):
 52      def test_report_calculations_and_invalid_rows(self):
 53          with TemporaryDirectory() as tmp:
 54              base = Path(tmp)
 55              data = base / "sales.csv"
 56
data.write_text("date,product,quantity,unit_price\n2026-01-01,Pen,2,50\n2026-01-02,Pen,1,30\n2026-01-03,
Pen,0,90\n")
 57              totals, errors, count = calculate_sales(data)
 58              self.assertEqual(count, 2)
 59              self.assertEqual(totals["Pen"], Decimal("130"))
 60              self.assertEqual(len(errors), 1)
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • tests/test_automation.py

tests/test_automation.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 61–75 / 97. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • tests/test_automation.py • PART 5

```
61              csv_file, html_file, _ = write_report(data, base / "reports")
62              self.assertTrue(csv_file.is_file())
63              self.assertIn("130.00", html_file.read_text())
64
65      def test_html_escapes_product_names(self):
66          with TemporaryDirectory() as tmp:
67              base = Path(tmp); csv_file = base / "data.csv"
68              csv_file.write_text("date,product,quantity,unit_price\n2026-01-01,<script>,1,10\n")
69              _, report, _ = write_report(csv_file, base / "report")
70              self.assertIn("&lt;script&gt;", report.read_text())
71
72      def test_reject_missing_headers(self):
73          with TemporaryDirectory() as tmp:
74              path = Path(tmp) / "oops.csv"
75              path.write_text("name,price\nPen,10\n")
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • tests/test_automation.py

tests/test_automation.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 76–90 / 97. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • tests/test_automation.py • PART 6

```
76              with self.assertRaises(ValueError):
77                  calculate_sales(path)
78
79
80  class BackupTests(unittest.TestCase):
81      def test_archive_contains_file(self):
82          with TemporaryDirectory() as tmp:
83              base = Path(tmp); source = base / "source"; source.mkdir()
84              (source / "readme.txt").write_text("backup me")
85              target = create_backup(source, base / "backups")
86              with ZipFile(target) as archive:
87                  self.assertEqual(archive.read("readme.txt"), b"backup me")
88
89      def test_backup_folder_not_nested(self):
90          with TemporaryDirectory() as tmp:
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

FULL SOURCE CODE • tests/test_automation.py

tests/test_automation.py

පහත code lines ඔබට ලබාදුන් ZIP project එකේම තියෙනවා. Lines 91–97 / 97. සම්පූර්ණ file එක run කිරීමට ZIP එක භාවිතා කරන්න.

Visual Studio Code • tests/test_automation.py • PART 7

```
91              path = Path(tmp)
92              with self.assertRaises(ValueError):
93                  create_backup(path, path / "backups")
94
95
96  if __name__ == "__main__":
97      unittest.main()
```

මෙය file එකේ කොටසක් පමණි. Appendix එකෙන් අනුපිළිවෙළට කියවන්න; source ZIP එකේ සම්පූර්ණ working files ලබාදී තිබෙනවා.

DOCUMENTATION • PART 1/2

Software Requirements Specification

මෙය source project එකේ docs/SRS.md ලේඛනයේ කියවිය හැකි කොටසක්. සම්පූර්ණ ලේඛනය ZIP එකෙන් ලබාගන්න.

```
# Student SRS - Little Star Automation Center

**Goal:** save time by organizing demo files, summarizing shop sales, and creating a ZIP backup.

## Users
Beginner students and teachers, Windows CMD users.

## Functional requirements
- FR01: Preview classification of top-level demo files.
- FR02: Copy files into extension categories only after explicit action.
- FR03: Never overwrite an existing different file.
- FR04: Read CSV sales and reject bad quantities and prices.
- FR05: Produce sales_summary.csv and sales_report.html.
- FR06: Archive sample data to a separate ZIP folder.
- FR07: Show errors instead of silently discarding problems.

## Nonfunctional requirements
- NFR01: Python 3.12+, standard library only.
```

Documentation exercise

ඔබගේම project එකකට අදාළ කර requirement එකක් හෝ test එකක් තව එකතු කරන්න.

DOCUMENTATION • PART 2/2

Software Requirements Specification

මෙය source project එකේ docs/SRS.md ලේඛනයේ කියවිය හැකි කොටසක්. සම්පූර්ණ ලේඛනය ZIP එකෙන් ලබාගන්න.

```
- NFR02: Local/offline operation; no external API or database.
- NFR03: Clear messages and absolute output paths.
- NFR04: Demo data only; originals preserved by organizer.

## Out of scope
Email sending, production scheduling, internet scraping, real customer information, guaranteed backup/recovery.

## Acceptance criteria
Run `python -m unittest discover -s tests -v` and inspect output folders and HTML manually.
```

Documentation exercise

ඔබගේම project එකකට අදාළ කර requirement එකක් හෝ test එකක් තව එකතු කරන්න.

DOCUMENTATION • PART 1/2

QA Test Plan

මෙය source project එකේ docs/QA_TEST_PLAN.md ලේඛනයේ කියවිය හැකි කොටසක්. සම්පූර්ණ ලේඛනය ZIP එකෙන් ලබාගන්න.

```
# QA Test Plan - Python Automation

| ID | Action | Expected result |
|---|---|---|
| T01 | `py app.py preview` | List only; no output folder created |
| T02 | `py app.py organize` | 4 copied sample files in classified folders |
| T03 | Run organize again | 0 additional copies, no overwrite |
| T04 | Add a second file with matching name | `_2` collision suffix used |
| T05 | `py app.py report` | CSV and HTML reports created |
| T06 | Invalid quantity 0 in sample | Rejected row reported |
| T07 | Open report HTML in Chrome | Sales table renders correctly |
| T08 | `py app.py backup` | Valid ZIP created in backups/ |
| T09 | Extract ZIP to separate temporary folder | Data can be read |
```

Documentation exercise

ඔබගේම project එකකට අදාළ කර requirement එකක් හෝ test එකක් තව එකතු කරන්න.

DOCUMENTATION • PART 2/2

QA Test Plan

මෙය source project එකේ docs/QA_TEST_PLAN.md ලේඛනයේ කියවිය හැකි කොටසක්. සම්පූර්ණ ලේඛනය ZIP එකෙන් ලබාගන්න.

| T10 | `py -m unittest discover -s tests -v` | Automated suite passes |

Bug record template: ID | Steps to reproduce | Expected | Actual | Severity | Fix | Retest.

Release checklist: Source intact; tests pass; sample data only; backup restore checked; runbook updated.

Documentation exercise

ඔබගේම project එකකට අදාළ කර requirement එකක් හෝ test එකක් තව එකතු කරන්න.

DOCUMENTATION • PART 1/1

SDLC & Maintenance

මෙය source project එකේ docs/SDLC.md ලේඛනයේ කියවිය හැකි කොටසක්. සම්පූර්ණ ලේඛනය ZIP එකෙන් ලබාගන්න.

```
# SDLC Walkthrough

1. Requirements - Ask what repeated work needs automating.
2. Design - Draw `Input -> Preview -> Copy/Report/Backup -> Review`.
3. Implement - Build small Python functions before joining them in app.py.
4. Test - Try normal, empty, wrong-input, and collision scenarios.
5. Release - Hand over a ZIP with instructions and sample data.
6. Maintain - Collect errors, review logs, make backups, and rerun tests after changes.

Change request example: Add `.webp` to Images. Update tests and user guide; do not patch code without review.
```

Documentation exercise

ඔබගේම project එකකට අදාළ කර requirement එකක් හෝ test එකක් තව එකතු කරන්න.

DOCUMENTATION • PART 1/2

Scheduling Runbook

මෙය source project එකේ docs/WINDOWS_SCHEDULER.md ලේඛනයේ කියවිය හැකි කොටසක්. සම්පූර්ණ ලේඛනය ZIP එකෙන් ලබාගන්න.

```
# Windows Task Scheduler - manual setup

1. Run `py app.py preview` and test all tasks manually first.
2. Use Start > Task Scheduler > Create Basic Task.
3. Trigger: Daily at an appropriate safe time.
4. Action: Start a program.
5. Program/script: full path to your Python executable (find with `where python`).
6. Add arguments: `"C:\FULL\PATH\TO\app.py" report` (start with report, not all).
7. Start in: full folder holding `app.py` (without quotes when using this GUI field).
8. Finish, then right-click the task > Run, examine Last Run Result and generated files.
9. Stop/disable the task if it behaves unexpectedly.
```

Documentation exercise

ඔබගේම project එකකට අදාළ කර requirement එකක් හෝ test එකක් තව එකතු කරන්න.

DOCUMENTATION • PART 2/2

Scheduling Runbook

මෙය source project එකේ docs/WINDOWS_SCHEDULER.md ලේඛනයේ කියවිය හැකි කොටසක්. සම්පූර්ණ ලේඛනය ZIP එකෙන් ලබාගන්න.

Do NOT put `py app.py report` as the executable name without checking how the wizard interprets arguments. On machines where `python` is a Microsoft Store alias, use the actual installed python.exe path. Task Scheduler may run under different permissions/accounts.

For kids, scheduling is an adult-supervised optional extension. Do not schedule file-moving jobs on real personal folders.

Documentation exercise

ඔබගේම project එකකට අදාළ කර requirement එකක් හෝ test එකක් තව එකතු කරන්න.

OFFICIAL RESOURCES

තවත් ඉගෙනගන්න • 2026

Python standard library එක ගැන නිල documentation බලන්න. Python 3.15.2 (October 2026) ලබා ගත හැකියි; නමුත් මේ project එක Python 3.12+ සඳහා සරල APIs මත පදනම් වෙනවා.

https://www.python.org/downloads/
https://docs.python.org/3/library/pathlib.html
https://docs.python.org/3/library/csv.html
https://docs.python.org/3/library/shutil.html
https://docs.python.org/3/library/zipfile.html
https://docs.python.org/3/library/unittest.html
https://docs.python.org/3/library/decimal.html
https://learn.microsoft.com/windows/win32/taskschd/task-scheduler-start-page

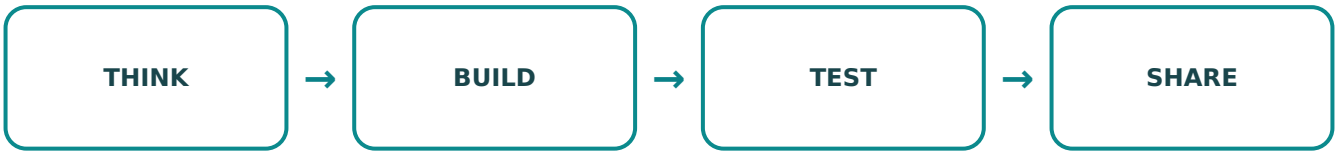
2026 ඔක්තෝබර් 31 වන විට ප්‍රකාශිත නිල releases අනුව සටහන් සකස් කළා. Website UI සහ installation පියවර ඔබේ Windows version අනුව වෙනස් විය හැකියි.

> **ඔබේ ඉදිරි ගමන**
> ඊළඟ automation idea එක: CSV report එකට bar chart එකක්, email නොයවන reminder preview එකක්, හෝ local folder photo-counter එකක්.

THE END · KEEP LEARNING

ඔබ දැන් Python automation developer කෙනෙක්!

කුඩා steps අලුතින් නිර්මාණය කරන්න. සෑම වෙනසක්ම `preview → test → verify` ක්‍රමයට යොදන්න. Automation එකේ අරමුණ මිනිසුන්ගේ වැඩ පහසු කිරීමයි, ඔවුන්ගේ data අනතුරට දැමීම නොවේ.



Book: Python Automation Sinhala 2026
Example: Little Star Automation Center
Technology: Python standard library + Windows CMD
Website: egotechworld.com

Source code and documentation are included as a separate ZIP.