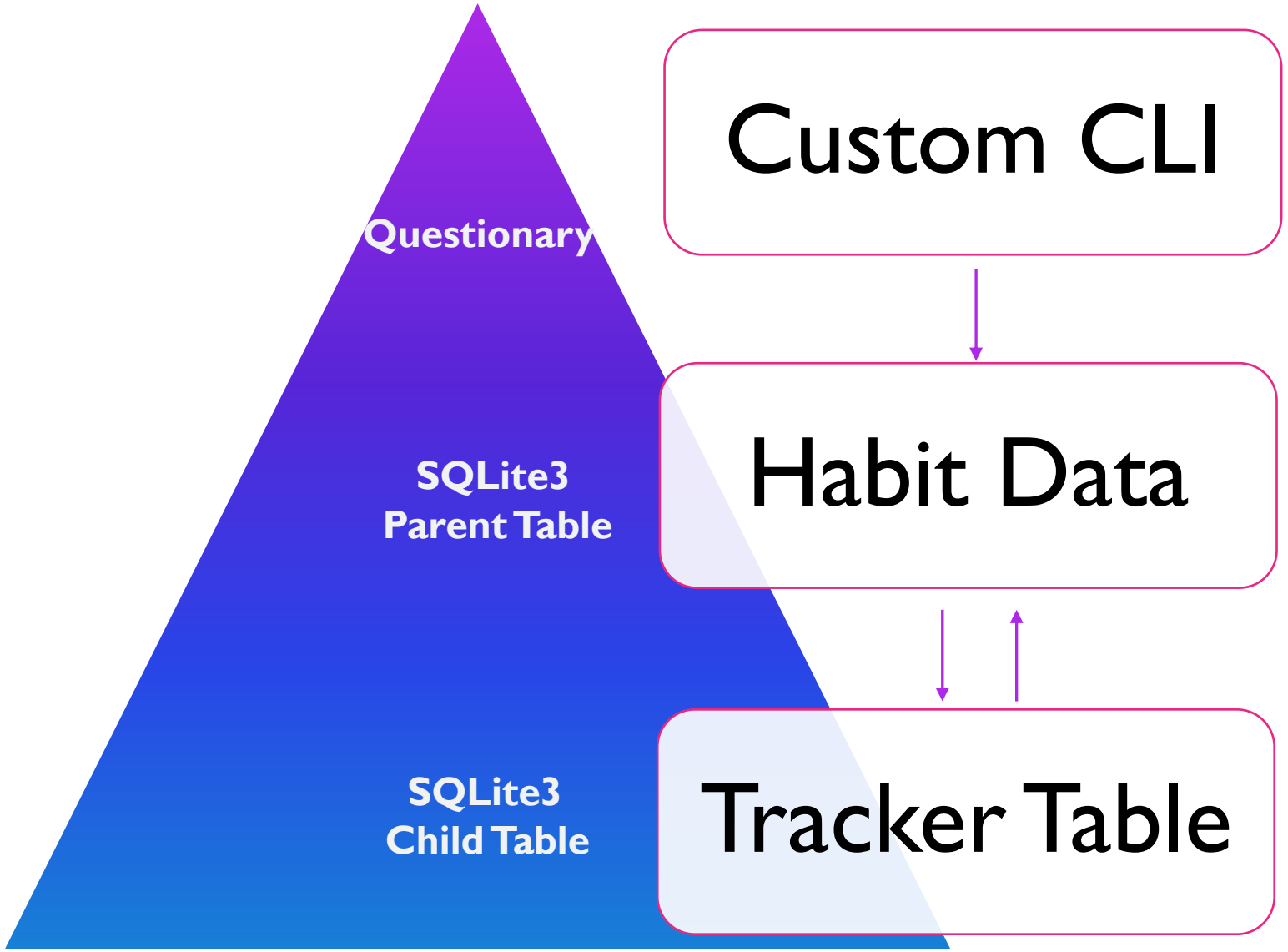
HABIT TRACKER

Design and Implementation Overview

By Tristyn Stein

Core Template



Custom CLI

Habit Data

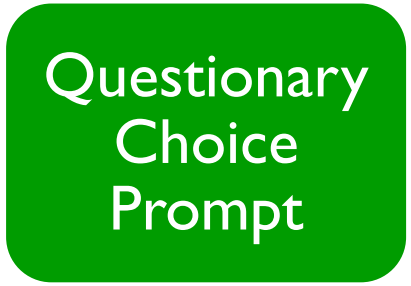
Tracker Table

A custom command line interface allows for users to interact with the application through prompts, negating need for advanced programming knowledge.

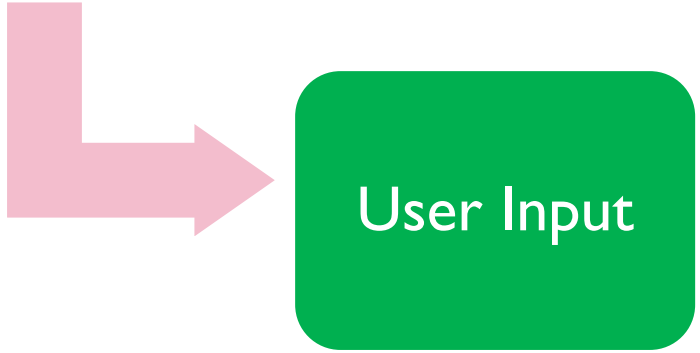
Habit names, descriptions, and periodicities are saved to permanent storage through SQL integration, allowing for referential databases.

A child table uses the habit name as a key to reference the main table, whilst storing date and streak information in a separate, readable table.

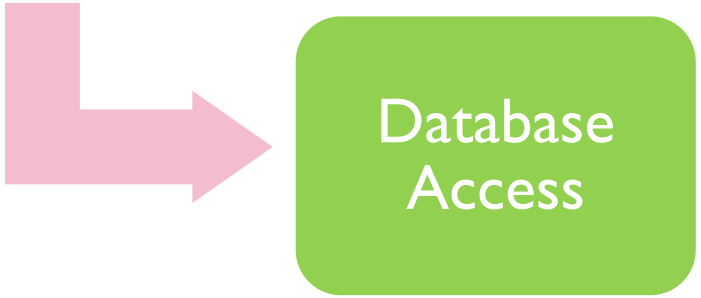
CLI FUNCTION FLOW



- The Python Questionary plugin allows for defined choices to be prompted to the user on the applications main stage. The user can select from the prompts using arrow keys and enter.



- The questionary tool is used to prompt the user for input, such as the name of a habit to look up display data of, or the name or description of a new habit to add.



- Functions check user input against database information and execute commands accordingly, such as writing a new function to the table, or retrieving streak data for a specified habit.

FUNCTIONS

- Add Habit
- Remove Habit
- Mark as Complete
- Show Habits
- Show a Habit's Streak
- Show my highest streak

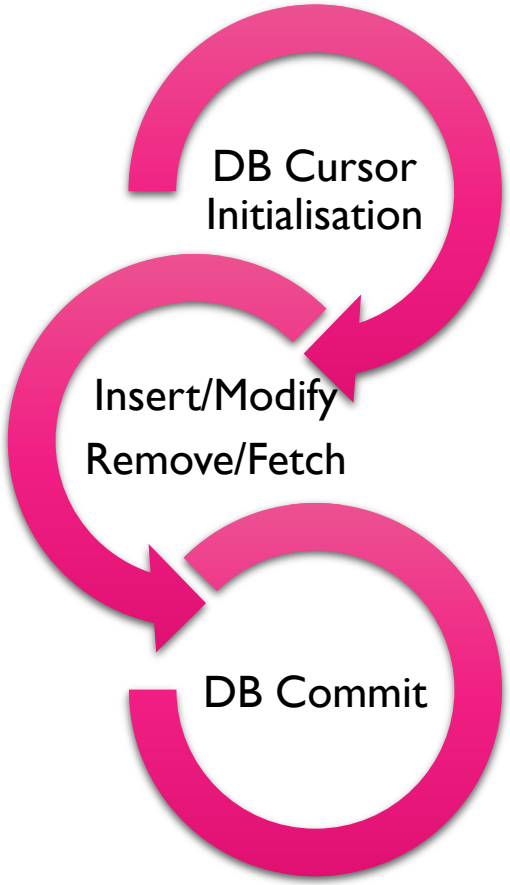
DATA STORAGE AND MANIPULATION

Habit Table

- Habit Name
  - Description
  - Periodicity

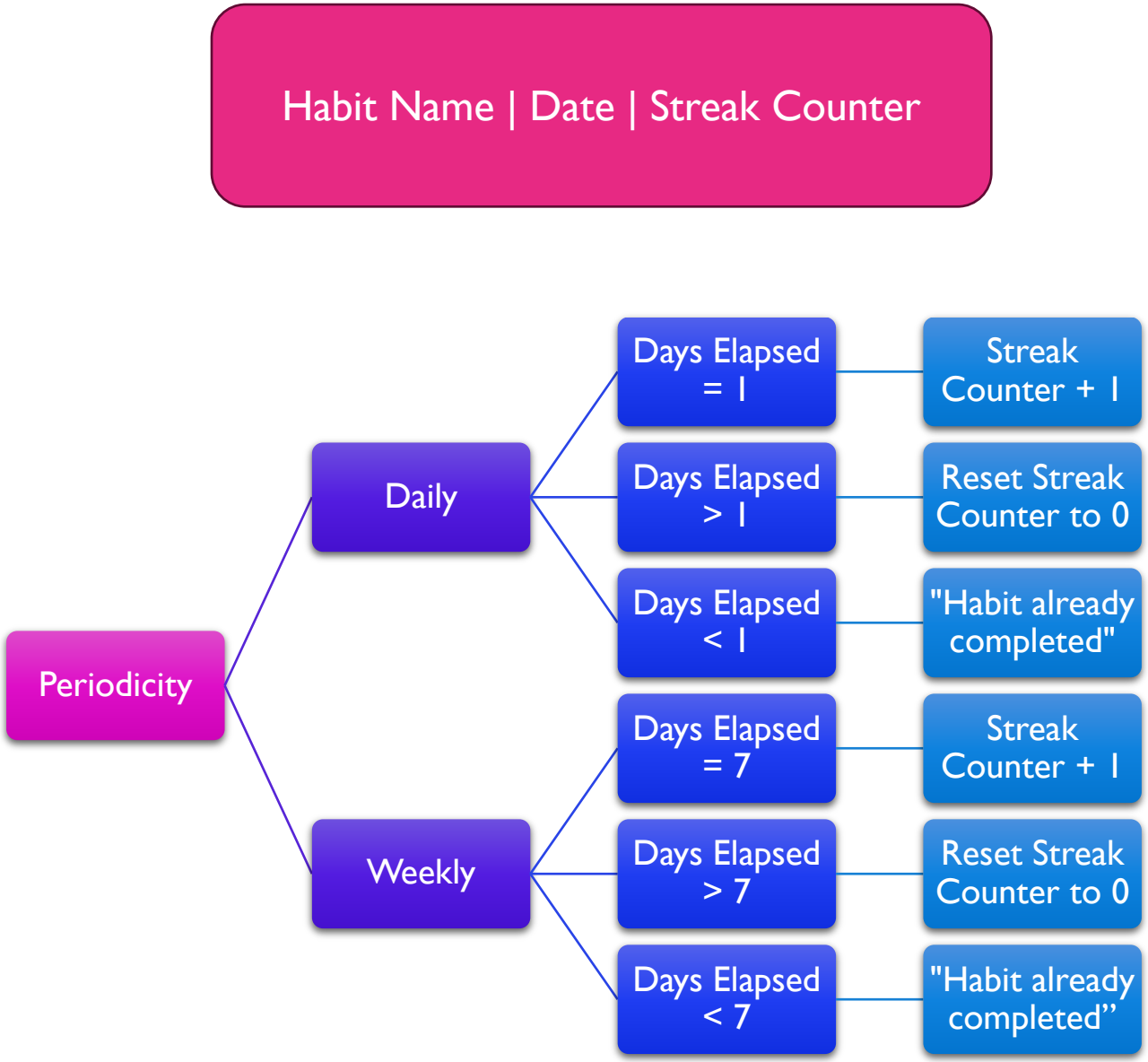
Tracker Table

- Habit Name [Key]
  - Date
  - Streak Counter

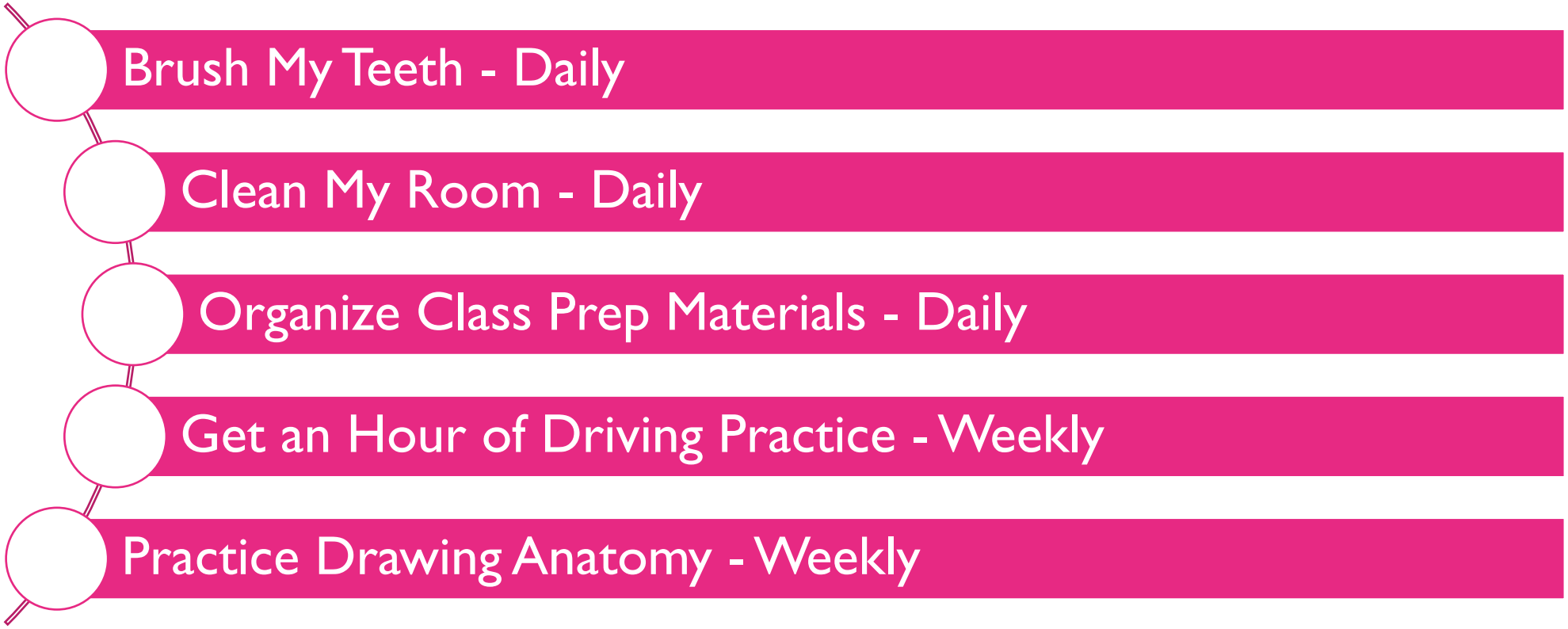


STREAK ANALYSIS

With tracker data in the database, a sqlite3 cursor fetches the last date of completion for a Habit. If the time delta between that date and the current date is greater than the periodicity of the habit, the application automatically writes a new row to the database with the current date and a streak counter value of 0 when launched.

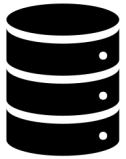


SAMPLE DATA


Brush My Teeth - Daily
Clean My Room - Daily
Organize Class Prep Materials - Daily
Get an Hour of Driving Practice - Weekly
Practice Drawing Anatomy - Weekly

PERMANENCE

The application has built-in checks when calling functions to read data from the database, ensuring the process continues even if a function returns null.



By saving new habit and tracker data in separate rows and calculating streaks through time delta, all data is kept until explicitly deleted. This allows for a history of tracker data to be kept – such as the provided sample data – and used in future additions to the application's capabilities.