**“Command-Line Password Manager”**

A secure Python CLI application that stores and retrieves encrypted passwords using a master key. Built with AES encryption and `getpass`, it keeps your credentials safe and local.

**Features**

- Master password authentication

- Remember multiple site credentials securely

- AES encryption using Fernet (`cryptography`)

- Stores data in a local JSON vault file

- View saved passwords only after successful login

- Easy to use, secure, and portable

**Technologies Used**

- Python 3.x

- `cryptography` (for AES encryption)

- `getpass` (for secure password entry)

- `json` (for local data storage)

**Getting Started**

git clone https://github.com/your-username/cli-password-manager.git

cd cli-password-manager

python -m venv .venv

.venv\Scripts\activate # Or source .venv/bin/activate (Linux/Mac)

pip install cryptographypython password\_manager.py

**Install dependencies**

pip install cryptography

**Run the application**

main.py

**“Expense & Budget Tracker”**

**Expense & Budget Tracker**

A GUI-based Expense & Budget Tracker built with Python.

It helps users track their daily expenses, set monthly budgets, and visualize spending patterns — all through a clean and simple interface.

**Project Overview**

This project is a personal finance tracker designed to help individuals:

- Log daily expenses by category

- Set and monitor monthly budgets

- Get insights with weekly/monthly reports

- View category-wise expense breakdowns

- Generate visual reports using pie charts

**Built With**

- Python 3

- Tkinter— for GUI

- CSV — for simple local data storage

- Matplotlib — for visualizing expenses

**Features**

| Feature | Description |

|----------------------|-----------------------------------------------------------------------------|

| Add Expenses | Log daily spending with category, amount, and notes |

| Set Monthly Budget | Define a budget and compare actual expenses against it |

| View Reports | Show total expenses, remaining budget, and category-wise breakdown |

| Pie Chart | Visual breakdown of where your money is going |

| CSV Storage | Simple and portable data persistence using a CSV file |

### Install Dependencies

pip install matplotlib

**Dynamic Web Scraper + Export to Excel**

**Dynamic Web Scraper + Export to Excel**

This is a dynamic web scraper built using \*\*Python and Selenium\*\*.

It extracts book data (title and price) from [Books to Scrape](https://books.toscrape.com), supports pagination, and automatically exports the collected data to an Excel file.

-Scrapes book titles and prices from multiple pages

- Supports pagination for dynamic data

- Exports scraped data to `scraped\_books.xlsx`

-Easy to customize for other sites

-Works with Selenium and ChromeDriver

**Technologies Used**

- Python 3.x

- Selenium WebDriver

- OpenPyXL (for Excel export)

- Chrome & ChromeDriver

**Requirements**

- Python 3.7 or higher installed

- Google Chrome browser installed

- Matching version of ChromeDriver

- pip (Python package manager)

**Getting Started**

1. Clone or download this repository:

git clone https://github.com/your-username/DynamicWebScraperExporttoExcel.git

cd DynamicWebScraperExporttoExcel

**Install Dependencies**

pip install selenium openpyxl

**Run the App**

dynamic\_web.py