

## Financial AI Bot

### Purpose

**Financial AI Bot** is a Telegram bot designed for personal finance tracking using artificial intelligence. The bot allows users to effortlessly track **income and expenses** by simply chatting with it in natural language. It automatically recognizes transaction details from user messages (amounts, categories, currencies, dates) and saves them to a database. This approach streamlines budgeting: no need to manually fill out spreadsheets—**Financial AI Bot** does it for you, and then helps **generate reports** and **export data** for further analysis. In the end, it saves time and improves accuracy by offering a convenient interface directly in Telegram.

### How It Works

The bot is implemented in Python and uses **OpenAI API** for message understanding, **Telegram Bot API** for communication, and **SQLite** for data storage. Here's how it works step by step:

1. **AI Message Processing.** The user sends a message to the bot describing a financial transaction (e.g., "*Bought coffee for \$10 yesterday*"). The bot forwards the message to OpenAI API, which analyzes it and extracts structured financial data: transaction type (income or expense), amount, currency, date, and category. The bot understands natural language context (e.g., "yesterday", currency symbols) and assigns the correct values.
2. **Saving the Transaction.** Once analyzed, the bot saves the extracted data into a local SQLite database. Each record includes: date, category, amount, currency, and type (income/expense). The database is stored locally (on the machine running the bot), ensuring secure storage of all user transaction history.
3. **Editing and Categories.** The bot supports interactive commands for managing records. For example, the `/edit` command lets users update previously saved transactions: the bot initiates a dialog, lists recent records or allows ID selection, and updates the necessary fields. Similarly, the `/category` command helps manage income/expense categories—users can add, rename, or view them to customize classification. The bot does not enforce a fixed category list—users define their own.
4. **Reports and Exporting.** **Financial AI Bot** can generate financial reports on demand. The `/report` command summarizes statistics for a specific period (e.g., current month), calculating total income and expenses, and grouping by category. Reports are sent as readable text or files (e.g., PDF). For more detailed analysis, the bot supports exporting transactions: the `/export` command allows users to download all or filtered records in **XLSX**, **CSV**, or **JSON** format. The bot uses *pandas* and *openpyxl* for generating tables and *FPDF* for PDF reports.

Bot logic is divided into modules for maintainability. The main file `main.py` handles application startup and command routing. Specific functionality is implemented in modules like `command_edit.py` (edit transactions), `command_export.py` (export), `command_report.py` (reports), `command_category.py` (categories), `command_currency.py` (currencies), and so on. This modular structure simplifies future development and feature extensions.

### How to Use

#### To run the bot:

1. **Install the environment.** Install Python (recommended version 3.8 or higher). Then clone the repository or download the project files to your computer/server.
2. **Install dependencies.** In the project directory, run:

```
pip install -r requirements.txt
```

Required libraries include: openai, python-telegram-bot, pandas, openpyxl, and fpdf. Make sure everything installs correctly.

3. **Configure the .env file.** Create a file named .env in the root of the project and add:

```
TELEGRAM_BOT_TOKEN=your_telegram_token
```

```
OPENAI_API_KEY=your_openai_key
```

**Important:** Do not share or publish this file—it contains private credentials.

4. **Run the application.** Once everything is set up, start the bot:

```
python main.py
```

Make sure the machine has a stable internet connection, as the bot relies on Telegram and OpenAI APIs.

5. **Chat with the bot in Telegram.** Find your bot (using the name you registered via BotFather) and send /start to initiate. You can immediately begin entering transactions in free form. Example:

"Received salary of \$5000 on May 5"

The bot will detect this as an income, amount \$5000, currency USD, date May 5, and potentially categorize it as "salary". It will confirm that the transaction has been saved.

Another example:

"Bought groceries for \$80 today"

The bot understands this as an expense and adds it accordingly.

#### **Available Telegram commands:**

- /start — Initializes the bot and sends a welcome message.
- /report — Generates a summary report for a given period, including totals and category breakdowns.
- /export — Allows exporting your transaction data in XLSX, CSV, or JSON.
- /edit — Lets you edit an existing transaction through a guided dialogue.
- /category — Manages your personal categories (add, delete, rename).
- /currency — Sets or changes your default currency.

Aside from commands, the bot also accepts natural language messages with financial content. This means you can use the bot intuitively, without needing technical knowledge or navigating complex menus.

#### **Important Notes**

- **.env file is mandatory.** Make sure .env exists and contains valid credentials. Without it, the bot will not start.
- **Local data storage.** All data is stored in a local SQLite database (e.g., finance.db). It is recommended to periodically back up this file.
- **Internet connection required.** The bot needs access to the internet to communicate with Telegram servers and OpenAI.
- **Privacy and costs.** Your financial messages are processed by OpenAI's servers. While the bot doesn't share this data further, it is sent externally for analysis. Also note: using OpenAI API may incur costs—ensure your OpenAI account has sufficient balance.

- **User-managed categories and currencies.** You are responsible for setting your own categories and preferred currency. This helps improve report accuracy and classification quality.
- **Cross-platform compatibility.** The bot works on Windows, macOS, and Linux, as long as Python is properly installed.

## Requirements

To run Financial AI Bot, you need:

- **Python 3.8+**
- **Python Libraries:**
  - openai
  - python-telegram-bot
  - pandas
  - openpyxl
  - fpdf
- **Service Credentials:**
  - Telegram Bot Token (via BotFather)
  - OpenAI API Key (from [OpenAI platform](#))
- **SQLite** (bundled with Python; no separate installation required)

Once all dependencies are installed and environment variables set, you can start using **Financial AI Bot** to automate and simplify your personal finance tracking—all within a familiar chat interface.