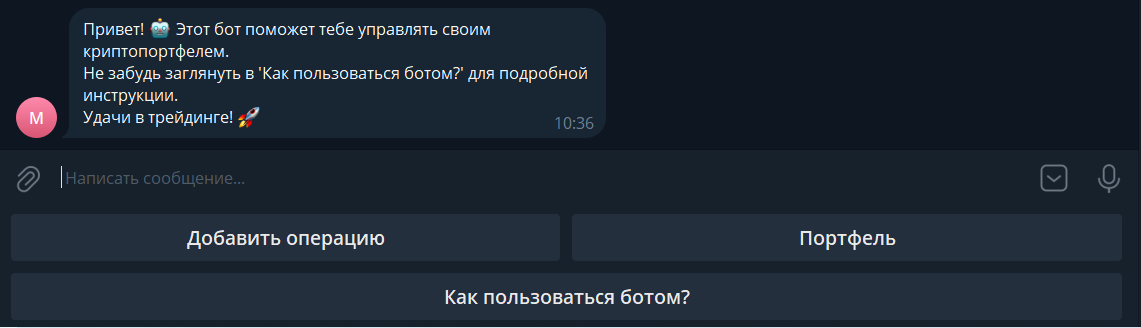
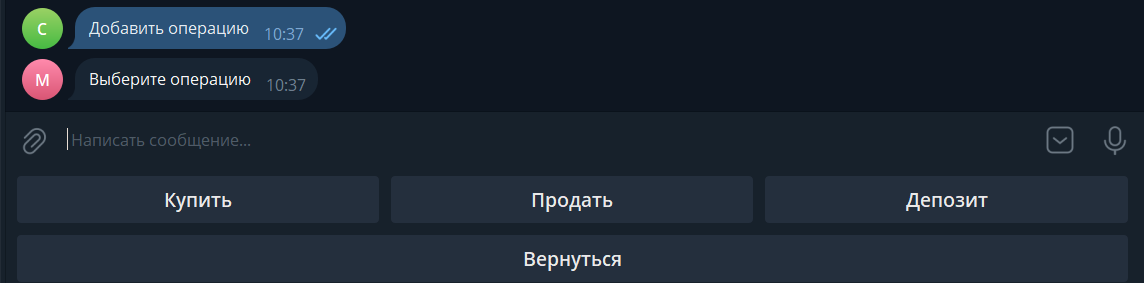
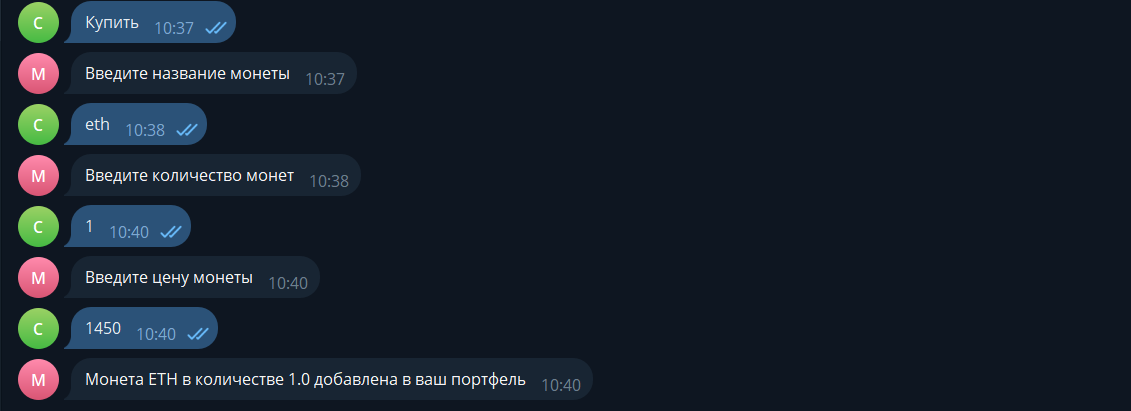
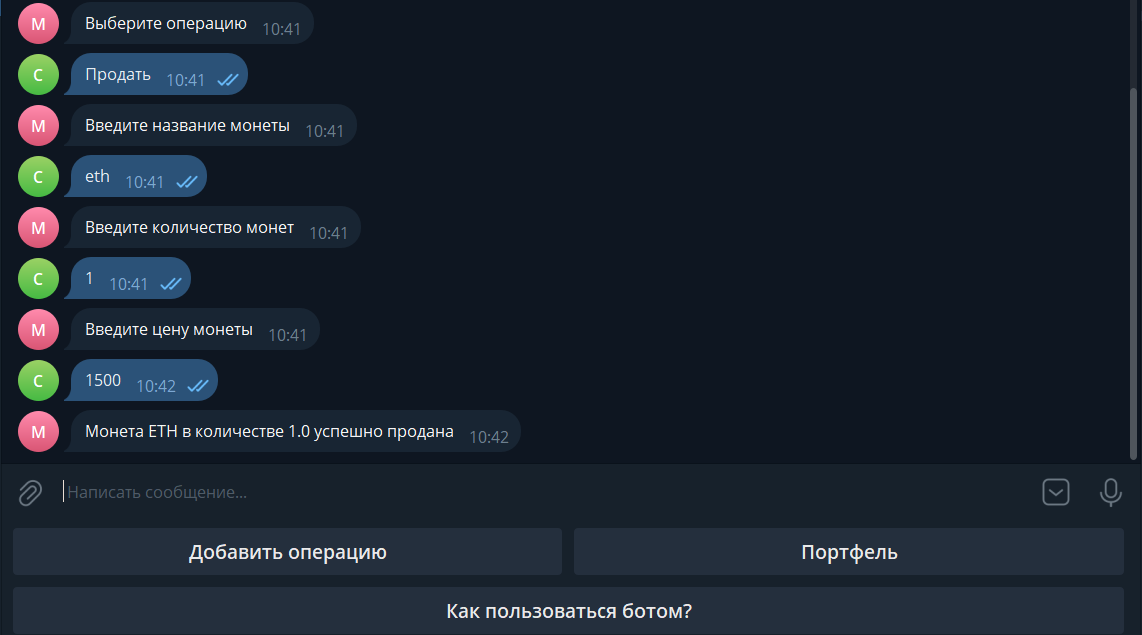
**Отчет по выполненной работе**

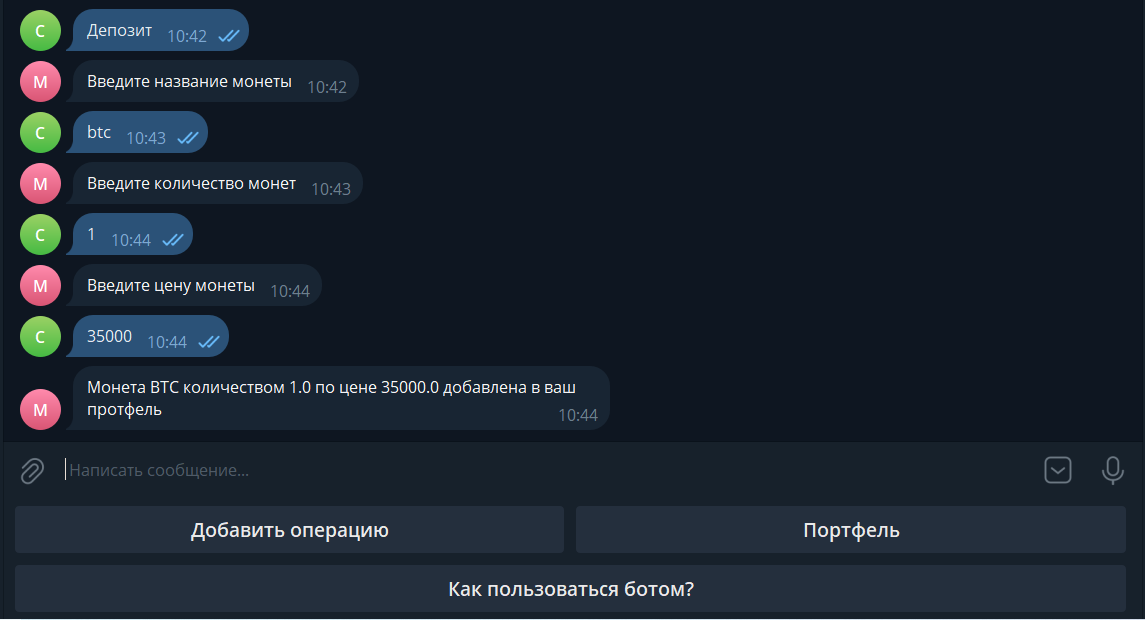
1. **Описание бота:**
   * **Название бота:** @My\_Crypto\_Wallet\_1\_Bot
   * **Язык программирования:** Python
   * **Библиотеки:** aiogram (Telegram бот), SQLite (хранение данных), ccxt (Запросы бирже Binance),
   * **Описание:** Бот предоставляет функционал криптопортфеля для учета финансов пользователя. Вся информация хранится в базе данных SQLite, а взаимодействие с пользователем осуществляется через интерфейс Telegram.
2. **Основной функционал бота:**
   * **/start:** Команда для начала взаимодействия с ботом. Пользователь получает приветственное сообщение от бота. Появляется основная клавиатура с тремя кнопками: "Добавить операцию", "Портфель", "Как пользоваться ботом?".  
     
   * **Добавить операцию:**



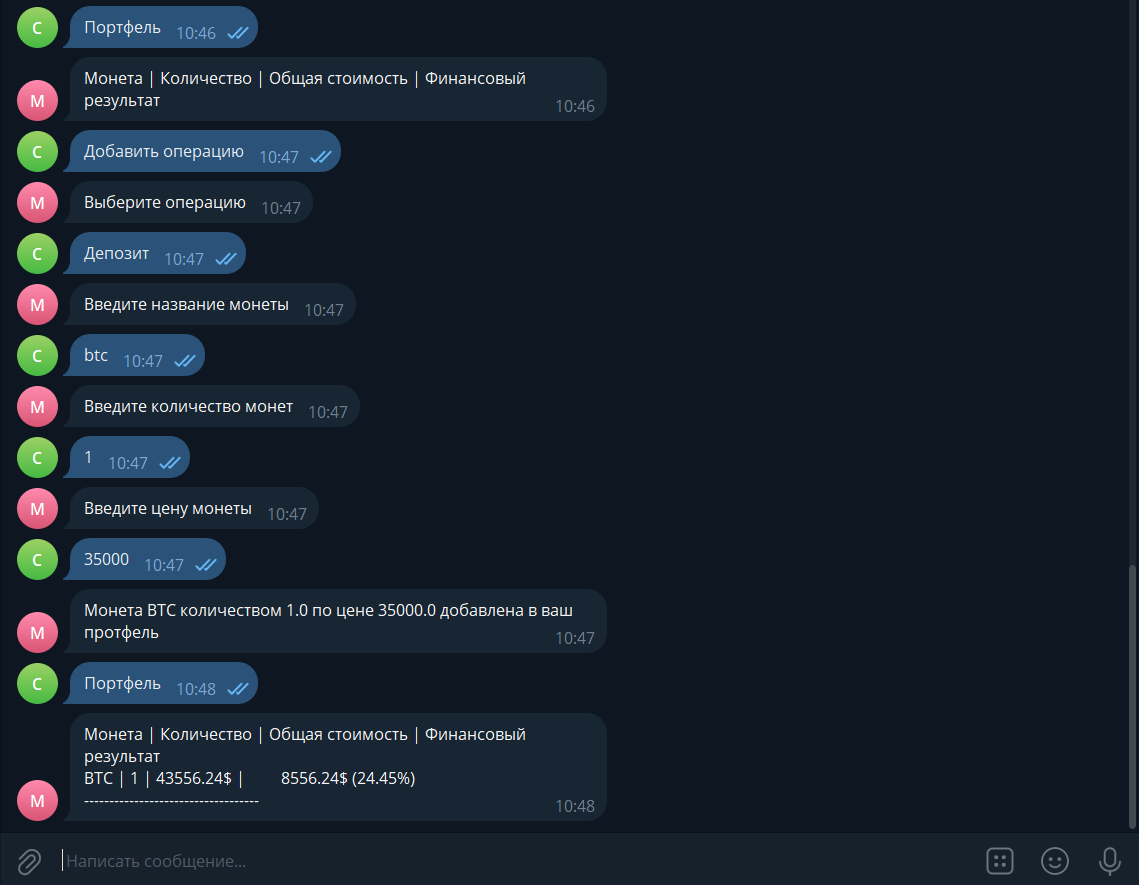
* + - **"Купить":** Покупка новой монеты за счет USDT в портфеле. Указывается название, количество и цена покупки.

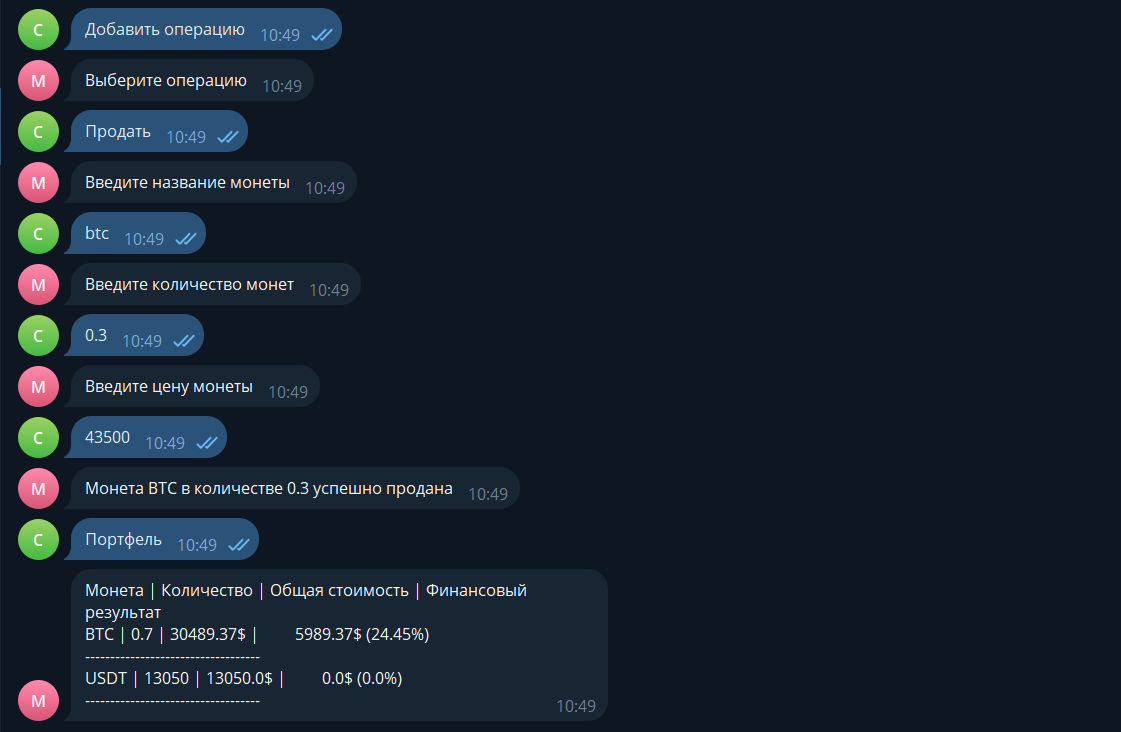


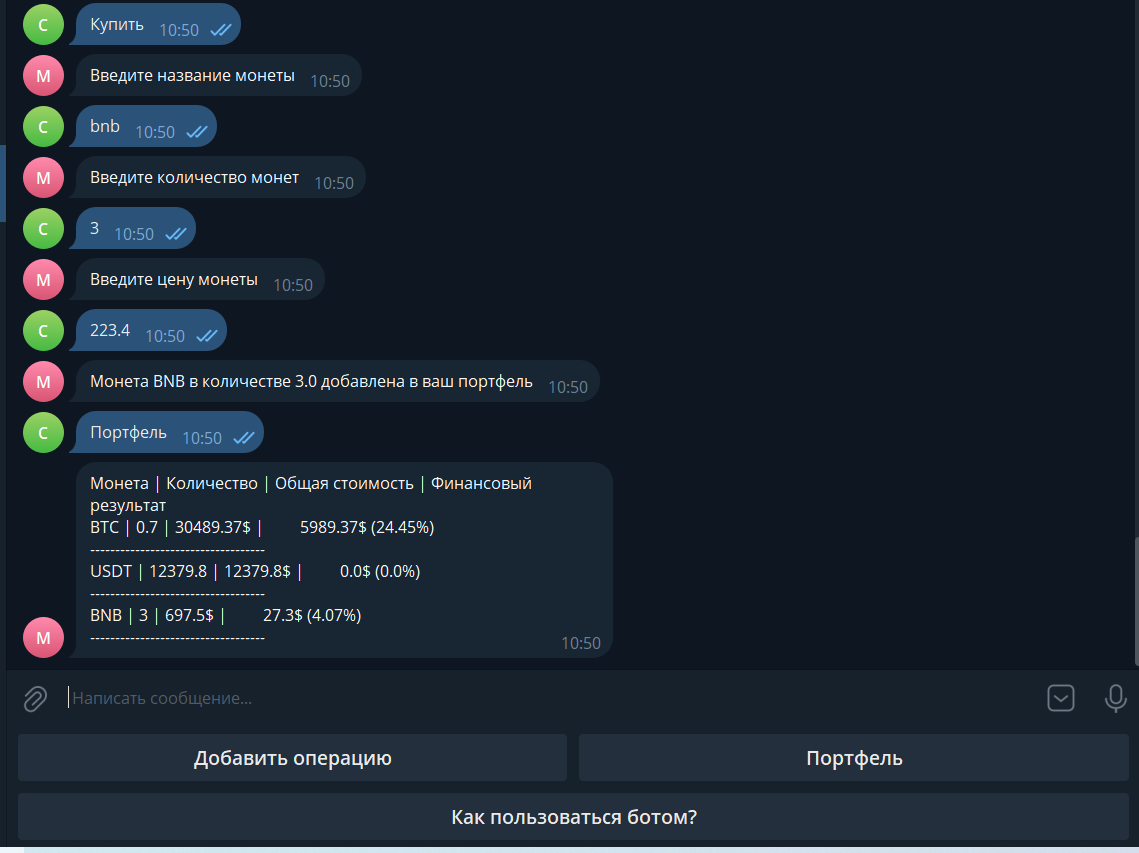
* + - **"Продать":** Продажа имеющейся монеты за USDT. Указывается название монеты, количество и цена продажи.  
      
    - **"Депозит":** Добавление новой монеты в портфель. Указывается название, количество и цена монеты в момент приобретения.



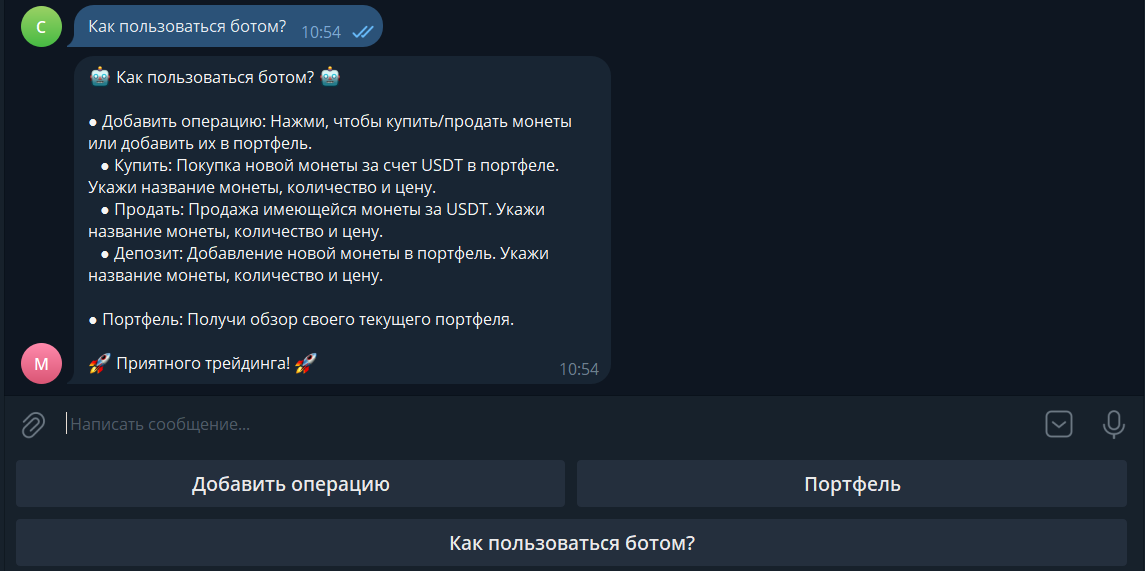
* + **Портфель:** Получение обзора текущего состояния портфеля.







* + **Как пользоваться ботом?:** Краткое руководство по использованию функционала бота.



1. **Файлы и структура проекта:**
   * **main.py:** Основной файл бота.
   * **keyboards.py**: Клавиатуры для бота
   * **config.py:** Системные настройки бота
   * **check\_token.py:** Модуль проверки монеты на бирже
   * **create\_db.py:** Модуль создание и обновление базы данных
   * **parse\_token\_price.py:** Модуль получения актуального курса валют и обновление базы данных
   * **crypto.db:** База данных**Начало формы**