Binance Mock Trading Bot – Project Documentation

📌 Overview

This project is a **simplified command-line trading bot** implemented in Python. It simulates trading on the Binance Futures Testnet by allowing users to place **Market**, **Limit**, and **Stop-Limit** orders using a clean and interactive command-line interface.

✅ Features

- Simulated trading on Binance Futures Testnet
- Supports `Market`, `Limit`, and `Stop-Limit` order types
- Handles both `Buy` and `Sell` orders
- CLI-based input with validation
- Structured output and execution status reporting
- Logs all orders and errors to local files
- Modular code (CLI, logic, mock API separated)
- Easily switchable to real Binance Testnet API

🔧 Technologies Used

- **Python 3.8+**
- Custom-built mock Binance client
- Native `logging` module
- Modular Python scripting

⚠️ Note: Mock Binance Client Used

Due to access issues with the Binance Futures Testnet (API restrictions or region blocks), a **mock implementation** was used to simulate Binance API behavior.

The structure mimics the official client, so the code can be easily switched to real trading when access becomes available.

🧪 Supported Order Types

Order Type	Description
Market	Executes immediately at current market price
Limit	Executes at a user-defined limit price
Stop-Limit	Becomes a limit order when a stop price is hit

▶️ How to Run

From your terminal or command prompt:

```
python main.py
```

You'll be prompted to enter:

- Trading pair (e.g., `BTCUSDT`)
- Order side (`BUY` or `SELL`)
- Order type (`MARKET`, `LIMIT`, or `STOP_LIMIT`)
- Quantity
- Price (based on order type)

📁 Folder Structure

```
binance-mock-bot/
├── main.py
├── bot.py
├── cli.py
├── mock_binance.py
├── utils.py
├── requirements.txt
├── README.md
└── logs/
    ├── order_log.txt
    └── error_log.txt
```

🪵 requirements.txt

```
datetime
```

🔄 Switching to Real Binance API (Optional)

If you get access to Binance Testnet in the future, follow these steps:

Step 1 – Install the Official Library

```
pip install python-binance
```

Step 2 – Replace the Mock Client

In `bot.py`:

```python
from binance.client import Client

class BasicBot:
    def __init__(self, api_key, api_secret, testnet=True):
        self.client = Client(api_key, api_secret)
        if testnet:
            self.client.API_URL = "https://testnet.binancefuture.com/fapi"
```

Step 3 – Place Real Orders

Replace mock `create_order()` with:

```python
self.client.futures_create_order(
    symbol=symbol,
    side=side,
    type=order_type,
    quantity=quantity,
    price=price,           # For LIMIT or STOP_LIMIT
    stopPrice=stop_price,  # For STOP_LIMIT only
    timeInForce="GTC"
)
```

📜 License

This project is open for educational and assessment purposes.