Binance Futures Trading Bot – Project Report
Author: SANKET S B
Date: 17/08/2025
Conatct:7411178329

Introduction

This project is a Python-based trading bot for Binance USDT-M Futures Testnet.
It allows users to place Market and Limit orders using a Command-Line Interface (CLI).
The bot supports input validation, error handling, and detailed logging of all orders.
It was developed as part of the Junior Python Developer hiring task.

Features

- ✔️ Market Orders – Execute instantly at current market price

- ✔️ Limit Orders – Execute at a user-defined price

- ✔️ Input validation for symbol, price, and quantity

- ✔️ Logging of orders, executions, and errors (`bot.log`)

- ✔️ Command-line interface for easy usage

How to Run
Install Python

- Ensure Python **3.8+** is installed.

- Check version:

```
python --version
```

Clone/Download the Project

If you have the `.zip` file:

```
unzip sanket_binance_bot.zip
cd sanket_binance_bot
```

If from GitHub:

```
git clone https://github.com/yourusername/sanket-binance-bot.git
cd sanket-binance-bot
```

Install Dependencies

Install required libraries:

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

Configure Binance API Keys

1. Register on **Binance Futures Testnet**:
   https://testnet.binancefuture.com

2. Generate API Key & Secret.

3. Add them into an `.env` file in your project root:

```
API_KEY=your_testnet_api_key
API_SECRET=your_testnet_api_secret
```

4. The bot will read keys from this file automatically.

---

5. Run the Bot

Market Order (executes immediately at market price):

```
python src/market_orders.py BTCUSDT BUY 0.01
```

● (venv) sanket@sanket:~/Downloads/Binance$ python src/market_orders.py BTCUSDT BUY 0.01
✅ Order placed successfully: {'orderId': 5593213547, 'symbol': 'BTCUSDT', 'status': 'NEW', 'clientOrderId': 'x-Cb7ytekJ1a1d8497066205c59bcefc', 'price': '0.00', 'avgPrice': '0.00', 'origQty': '0.010', 'executedQty': '0.000', 'cumQty': '0.000', 'cumQuote': '0.00000', 'timeInForce': 'GTC', 'type': 'MARKET', 'reduceOnly': False, 'closePosition': False, 'side': 'BUY', 'positionSide': 'BOTH', 'stopPrice': '0.00', 'workingType': 'CONTRACT_PRICE', 'priceProtect': False, 'origType': 'MARKET', 'priceMatch': 'NONE', 'selfTradePreventionMode': 'EXPIRE_MAKER', 'goodTillDate': 0, 'updateTime': 1755448542391}

Limit Order (executes only when the market hits your set price):

```
python src/limit_orders.py BTCUSDT SELL 0.01 60000
```

● (venv) sanket@sanket:~/Downloads/Binance$ python src/limit_orders.py BTCUSDT SELL 0.01 60000
✅ Limit order placed successfully: {'orderId': 5593156939, 'symbol': 'BTCUSDT', 'status': 'NEW', 'clientOrderId': 'x-Cb7ytekJ9440516803ea4be3fa5345', 'price': '60000.00', 'avgPrice': '0.00', 'origQty': '0.010', 'executedQty': '0.000', 'cumQty': '0.000', 'cumQuote': '0.00000', 'timeInForce': 'GTC', 'type': 'LIMIT', 'reduceOnly': False, 'closePosition': False, 'side': 'SELL', 'positionSide': 'BOTH', 'stopPrice': '0.00', 'workingType': 'CONTRACT_PRICE', 'priceProtect': False, 'origType': 'LIMIT', 'priceMatch': 'NONE', 'selfTradePreventionMode': 'EXPIRE_MAKER', 'goodTillDate': 0, 'updateTime': 1755447071286}

---

6. Check Logs

- The bot creates a file:

  `bot.log`

- It contains all order requests, errors, and execution results.