




# Binance Futures CLI Bot – Report

---

 Developed by: Abqariyah

 Project: Binance USDT-M Futures Trading Bot (Mock Version)

 Date: 11 July 2025

## Objective

To build a command-line Binance Futures bot that simulates placing:

- Market Orders
- Limit Orders
- Stop-Limit Orders

with structured logging and input validation, using mock mode for testing.

## Features Implemented

Feature	Description
Market Order	Simulates immediate buy/sell at market price
Limit Order	Places simulated buy/sell at specific price
Stop-Limit Order	Simulates stop trigger before placing limit order
Logging	`bot.log` records every action and error with timestamps
CLI Commands	Each order module is fully operable from terminal
Mock Mode	Safe for users without Binance API/KYC

## Screenshots to Include

1. Market Order: `python src/market_orders.py BTCUSDT BUY 0.01`

```
PS C:\Users\Zahid\Downloads\Abqariyah_binance_bot> python src/market_orders.py BTCUSDT BUY 0.01
✅ [SIMULATED] Market order: BUY 0.01 BTCUSDT
```

2. Limit Order: `python src/limit_orders.py BTCUSDT SELL 0.01 60000`

```
PS C:\Users\Zahid\Downloads\Abqariyah_binance_bot> python src/limit_orders.py BTCUSDT SELL 0.01 60000
✅ [SIMULATED] Limit order: SELL 0.01 BTCUSDT @ 60000.0
```

3. Stop-Limit Order: `python src/advanced/stop_limit.py BTCUSDT SELL 0.01 58000 57900`

```
PS C:\Users\Zahid\Downloads\Abqariyah_binance_bot> python src/advanced/stop_limit.py BTCUSDT SELL 0.01 58000 57900
✅ [SIMULATED] Stop-Limit order: SELL 0.01 BTCUSDT | Stop: 58000.0, Limit: 57900.0
```

## **Binance API & Mock Mode**

Since Binance now enforces strict identity verification (Aadhaar/PAN), the bot was built using mock mode to simulate realistic trades without needing access to Binance servers.

Mock mode helped achieve:

- Real command-line testing
- Safe development
- Assignment completion without personal risk

## **Future Improvements**

- Add TWAP or Grid strategies
- Enable real API integration once KYC is completed
- Improve exception handling and CLI input parsing