STOCK MARKET SIMULATION ENGINE

GETTING STARTED

Start by Importing Stock Market Simulation Engine into your Unity Project

Once that's done, you can open the scene where you want to create your Stock Market

You will see Window tab at the top of your Unity Navigation Bar

Click on **Window**, then find "**Stock Market**" and click the dropdown menu.

Click on **"Create a Market"** from the drop down menu.

A new menu will open, follow the steps carefully from **Step 1** to **Step 4.**

Step 1: Create Stock Market Manager

Creates the Stock Market Manager object in the hierarchy ***REQUIRED***

Step 2: Configure Stock Market Settings

Market Volatility can be set to Low, Normal, or High.

Market Tick Interval is how often the Stock Prices will update.

(If you don't know how to set these, it's recommended you select Normal and 2 Seconds Tick Interval)

Step 3: Add a Stock to the Stock Market:

You will need to name your stock or crypto currency (For Example: Bitcoin)

You will need to set a stock / crypto trading symbol (For Example: BTC)

You will need to set the starting price for your stock or crypto currency (For Example: 95.00)

Step 4: Create Stock Market UI List (Canvas)

This will create a demonstration canvas that supports a list of up to ten stocks.

The Stock Market List Demonstration is a great option if you wish to see quick results.

If you want to integrate the system into your own UI layout, or create your own stock list that supports more than 10 stocks....

Then you will want to continue reading the documentation to see the various ways that you can access stock market information.

HOW TO ACCESS STOCK MARKET DATA

FOR YOUR OWN CUSTOM SCRIPTS

Inside the **Stock_Market_Sim_Engine > [Scripts] > Examples**

You will find **MarketDemonstration.cs**

This script is fully commented with ways to access existing stock market data.

*(This Script is Featured in the Demo Scene - **Market_Example_1**)*

Step 1: Add **using StockMarketSimulationEngine;** to the top of your custom script.

Step 2: Add **public StockMarketManager stockManager;** above your **Start()** function

Step 3: Add **stockManager = GameObject.FindObjectOfType<StockMarketManager>();**

inside of your **Start()** function

What we've done in the first three steps is gain access to the **StockMarketManager** script that exists in your hierarchy once you've completed the initial setup. Now you can review some of the way you can access stock market data below.

SET A STOCK PRICE BY ITS STOCK SYMBOL

stockManager.SetStockPrice("LTC", 64.02f);

*This line of code sets an existing stock/crypto with the symbol **LTC** to a price of **$64.02***

GET A STOCK PRICE BY ITS STOCK SYMBOL

stockManager.GetStockPrice("LTC").ToString("F2")

*This line of code gets an existing stock/crypto with the symbol **LTC** (and formats it into a string)*

GET A STOCK'S PERCENTAGE CHANGE SINCE THE MARKET OPENED

stockManager.GetStockPercentage("LTC"); *(float)*

*This line of code gets the percentage gain/loss of an existing stock/crypto with the symbol **LTC***

(Example: +1.25%)

GET A STOCK'S PRICE CHANGE SINCE THE MARKET OPENED

stockManager.GetStockPriceChange("LTC"); *(float)*

*This line of code gets the price gain/loss of an existing stock/crypto with the symbol **LTC***

(Example: +$2.25)

OPEN THE STOCK MARKET

```
stockManager.SetMarketOpen();
```

This line of code opens the stock market and allows prices to fluctuate.

This line of code resets the Gain/Loss to 0.00% and +$0.00 for each stock since a new market session starts.

CLOSE THE STOCK MARKET

```
stockManager.SetMarketClosed();
```

This line of code closes the stock market and does not allow prices to fluctuate.

THANK YOU FOR PURCHASING

NEED ASSISTANCE OR SUPPORT?

CONTACT: alert@flagmanjeremy.net

FLAGMAN
JEREMY
NETWORK

FlagmanJeremy.net