You are provided a .net standard library that mock a pricing system.

The library has a class named “TradingApi” which implement interface:

public interface ITradingApi

{

/// <summary>

/// Get full list of symbol definitions, only two symbols are supported: “AUDUSD” and “CL-OIL”.

/// </summary>

/// <returns>Symbols</returns>

IEnumerable<Symbol> GetSymbols();

/// <summary>

/// If there is an in-coming price and the symbol of the price is subscribed by SubscribeSymbol <see cref="SubscribeSymbol"/> an event will fire, the pricing data is passed as event data.

/// </summary>

event EventHandler<Price> OnPricing;

/// <summary>

/// Subscribe a symbol to receive prices though OnPricing event handler

/// </summary>

/// <param name="symbolName">name of a symbol</param>

void SubscribeSymbol(string symbolName);

/// <summary>

/// Unsubscribe a symbol

/// </summary>

/// <param name="symbolName">name of a symbol</param>

void UnsubscribeSymbol(string symbolName);

}

public struct Price

{

public string Symbol { get; }

public decimal Bid { get; }

public decimal Ask { get; }

public DateTime Time { get; }

}

public class Symbol

{

public string Name { get; set; }

public string BaseCurrency { get; set; }

public int Digits { get; set; }

}

public class PriceStats

{

public decimal Open { get; set; }

public decimal Close { get; set; }

public decimal Min { get; set; }

public decimal Max { get; set; }

}

Your task is to full fill the following requirements:

1. Create a Web API using .net or .net core that implement the followings:
2. GET /api/symbols
   * Return a list of all symbols
3. GET /api/symbols/{name}
   * Return a single **Symbol** object with the {name}
4. GET /api/price/{symbol}
   * Return the **latest** **Price** of a {symbol}
5. Get /api/price/{symbol}/history
   * Return the **PriceStats** that contains **Open, Close, Min and Max**, for the **current** **Minute**, for a given symbol
     1. Open - The **first** Bid price in the minute
     2. Close – the **last** Bid price in the minute
     3. Min – the **lowest** Bid price in the minute
     4. Max – the **highest** Bid price in the minute
   * A extension method will help you