Real Time Feed

Background

The server storing the code for an important real-time data processing feed has had a catastrophic malfunction. As a developer on the data team, you have been tasked with rewriting the process before the stock exchange starts trading in an hour.

Input

1. Data will be provided as comma-delimited strings representing a price quote.
2. The first row will be an integer representing the number of quotes that will be following.
3. All quotes will flow in ascending order of timestamp.
4. Timestamp for the quotes can span multiple days.

Row Schema:

Date	Time	Symbol	Price
Date of the quote in YYYY-mm-dd format	Time of the quote in HH:MM:SS format	Stock Ticker for the instrument	Current price

Sample input:

```
8
2017-01-03,16:18:50,AAPL,142.64
2017-01-03,16:25:22,AMD,13.86
2017-01-03,16:25:25,AAPL,141.64
2017-01-03,16:25:28,AMZN,845.61
2017-01-03,16:28:50,AAPL,140.64
2017-01-03,16:29:59,FB,140.34
2017-01-04,16:29:32,AAPL,143.64
2017-01-04,16:30:50,AAPL,141.64
```

Output

For this problem, assume the following:

1. The exchange starts trading daily at 09:30:00 hrsand closes at 16:30:00 hrsevery day. Any quotes outside this time window are invalid and should be ignored.

   Desired behavior:

   a) After exchange closes at 16:30:00 for each trading day, print
      1. Trading Day = <Date>

2. Last Quote Time = <Time of the last quote received before 16:30:00>
3. Number of valid quotes received for the day
4. Most active hour (maximum valid quotes per hour received during the trading day). If the maximum number of valid quotes per hour occurs for more than one hour, pick the earliest hour of the day.
5. Most active symbol (maximum valid quotes per symbol received during the trading day).If the maximum number of valid quotes per symbol occurs for more than one symbol, pick the first symbol (sorted alphabetically).

b) Calculate and print the following data for each Symbol as a comma-delimiter string. Rows should be printed in alphabetical order based on Symbol
   i. Time: Most recent timestamp for that Symbol in YYYY-mm-dd HH:MM:SS format
   ii. Symbol
   iii. High: Maximum Price that occurred for that Symbol during the trading day.
   iv. Low: Minimum Price that occurred for that Symbol during the trading day.

Sample output:

```
Trading Day = 2017-01-03
Last Quote Time = 16:29:59
Number of valid quotes = 6
Most active hour = 16
Most active symbol = AAPL
2017-01-03 16:28:50,AAPL,142.64,140.64
2017-01-03 16:25:22,AMD,13.86,13.86
2017-01-03 16:25:28,AMZN,845.61,845.61
2017-01-03 16:29:59,FB,140.34,140.34
Trading Day = 2017-01-04
Last Quote Time = 16:29:32
Number of valid quotes = 1
Most active hour = 16
Most active symbol = AAPL
2017-01-04 16:29:32,AAPL,143.64,143.64
```

Important Guidelines:

1. Please try to avoid use of any packages outside the standard Python library (for e.g. pandas) while writing the code.
2. While a complete and correct solution will be appreciated, we will be giving equal weightage to solutions that have a good design behind the implementation and a proper structure. Code that is production quality, easy to maintain and also easily extensible (if requirements change) will be duly appreciated.