**Stock Archives**

Name: Channaveer Patil

Phone #: 9945773086

Email: [channaveer.mpatil@gmail.com](mailto:channaveer.mpatil@gmail.com)

All responses are in the color

**Stock Archives**

**[Problem Statement]**

Develop a pseudo Full Stack Web Application for Stock Archives. A data dump is provided which contains the stocks related historical data from 2005 - 2016. Your web app should be an archive of historical stocks trading data. Users should be able to view history of desired stock prices for any point in time.

**[Ideal Behaviour]**

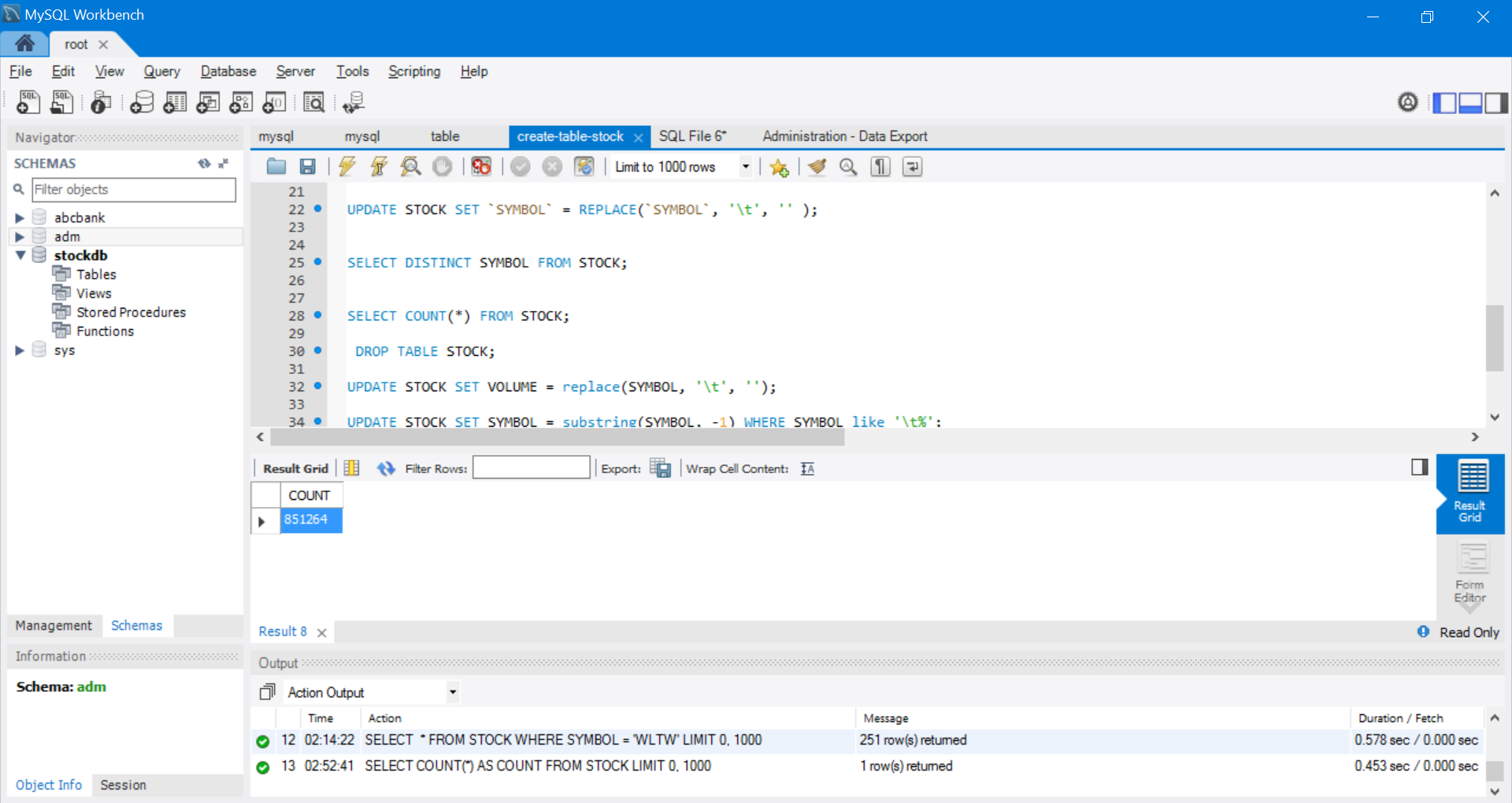
**On Back End**:

— Write a backend which exports APIs with all the data provided in the dump below.

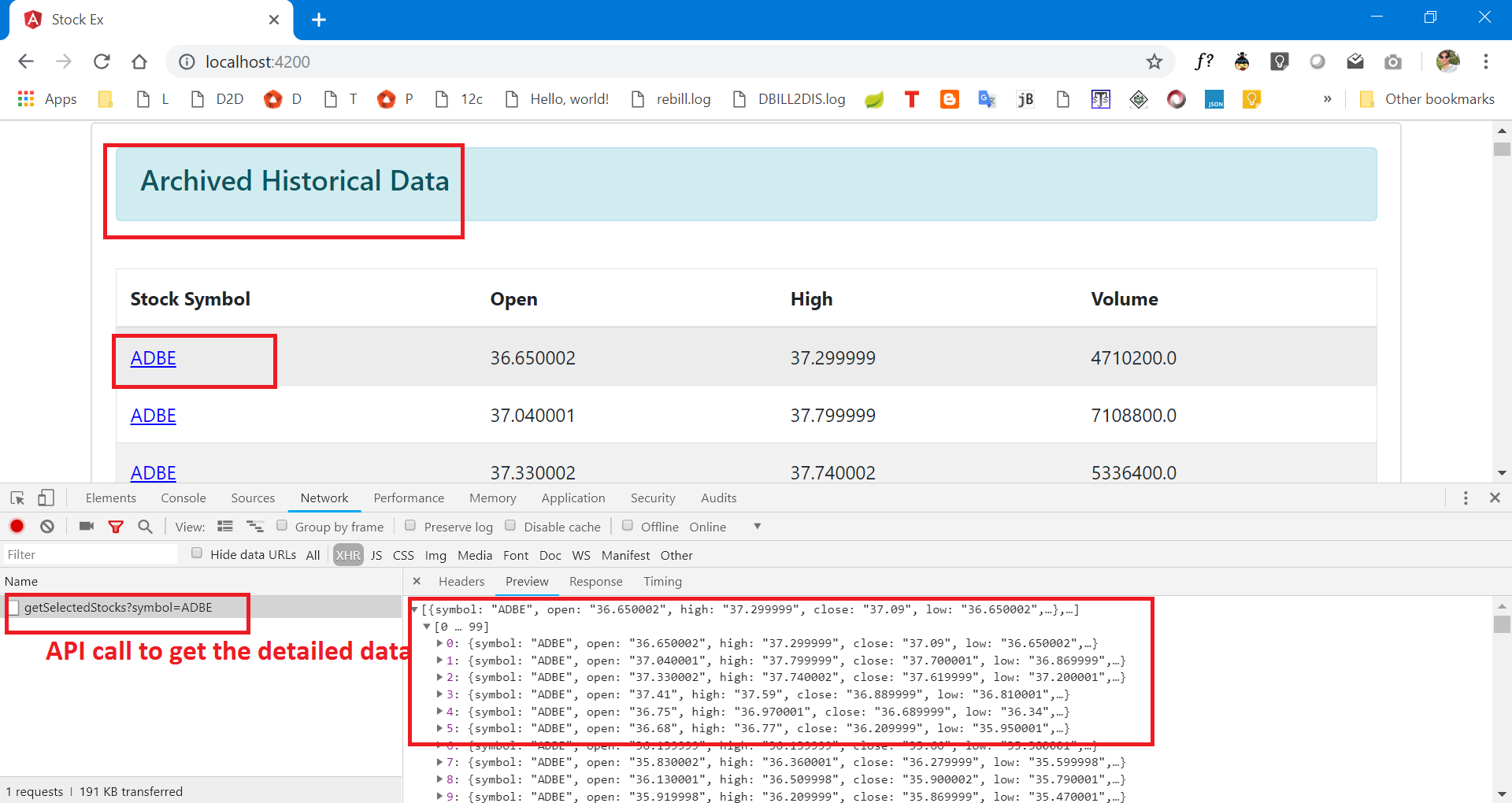
Backend Spring Boot Rest service calls used to access the data from DB

— Get all companies and their historical stock price related data from the provided dump file and store it in the database.

All the the data provided in the excel stored in MySql DB, Application is dealing with almost 851264 records (8.5 milling records)

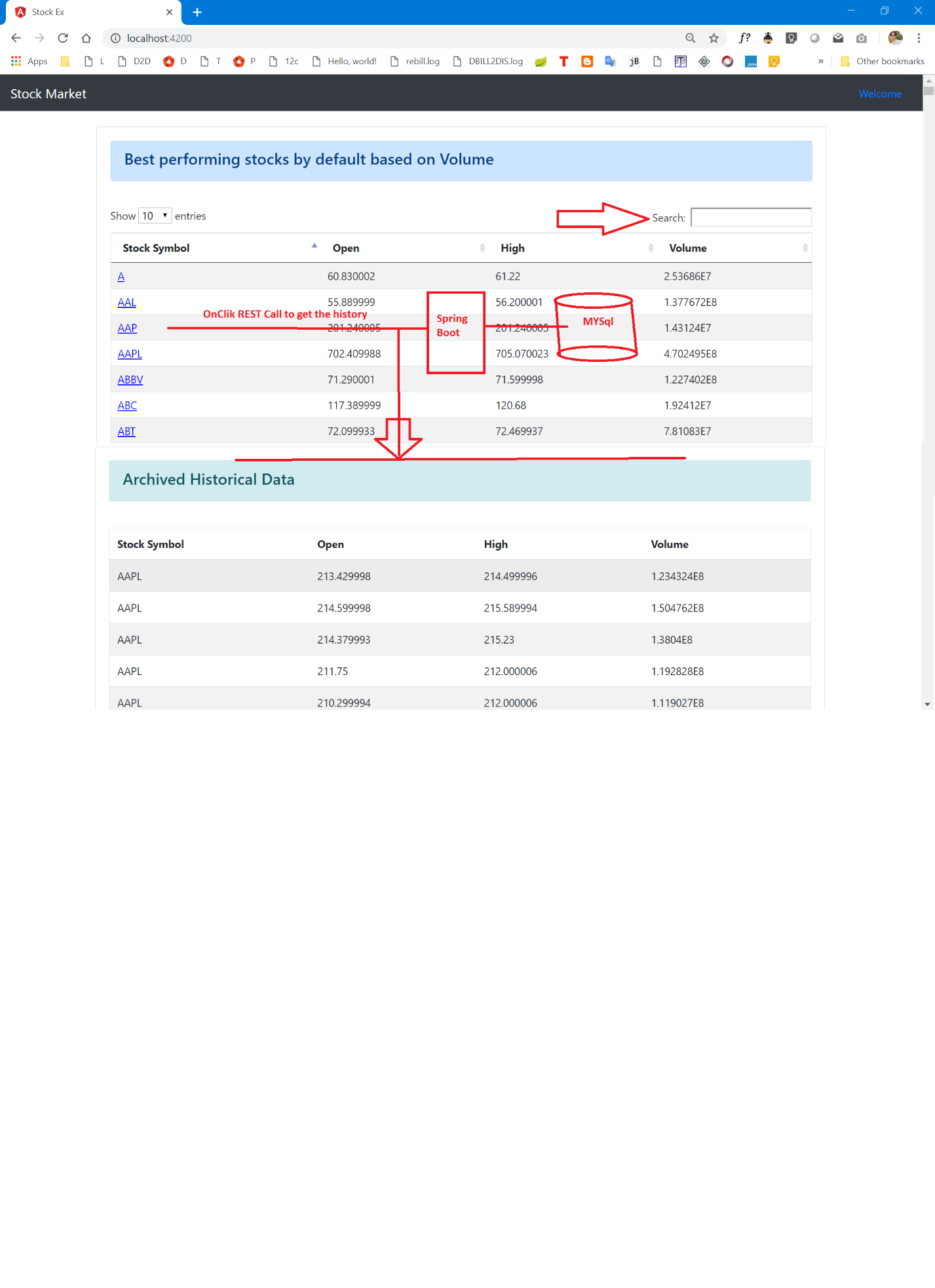


— You have to fetch this stored data based on the requests received from the frontend.



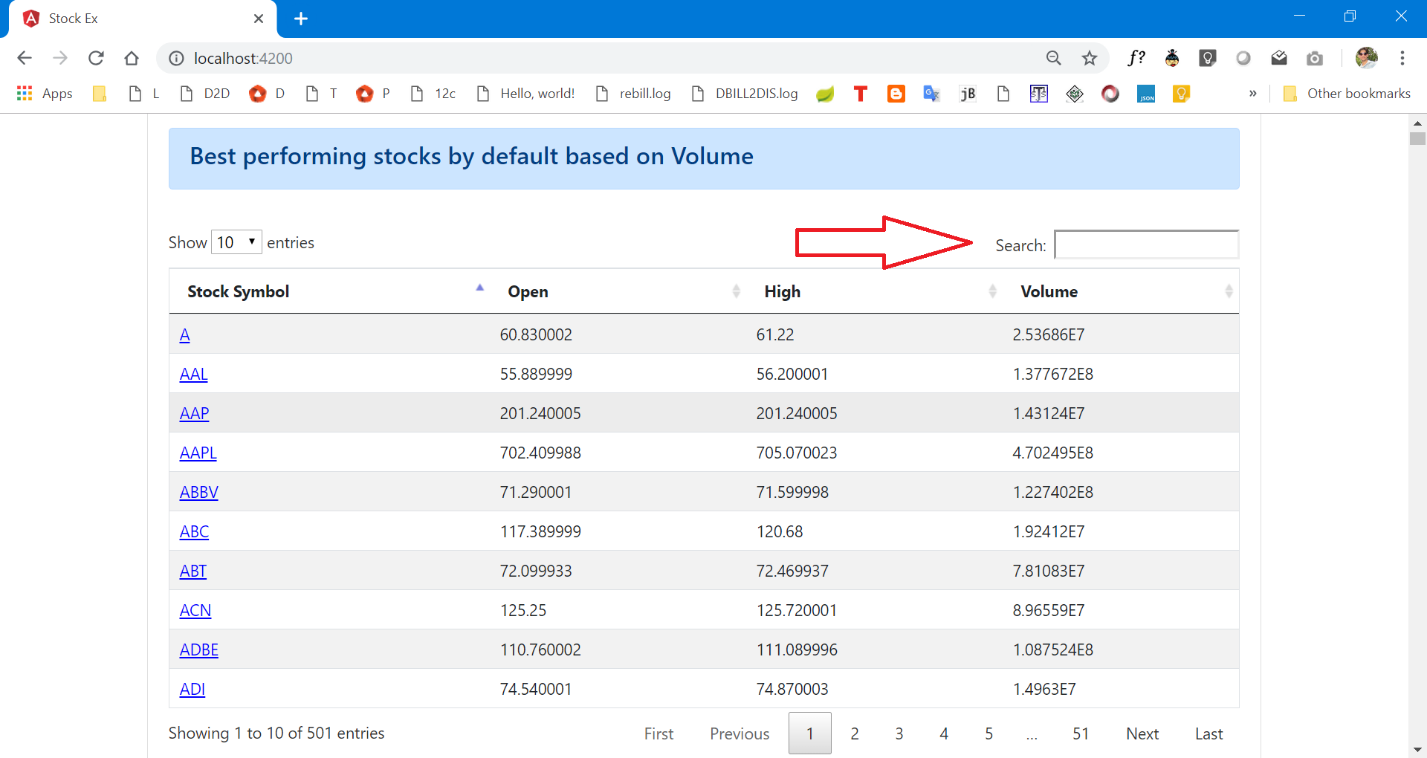
**On Front End**:

— Your frontend should be able to list all the archived data on the UI.

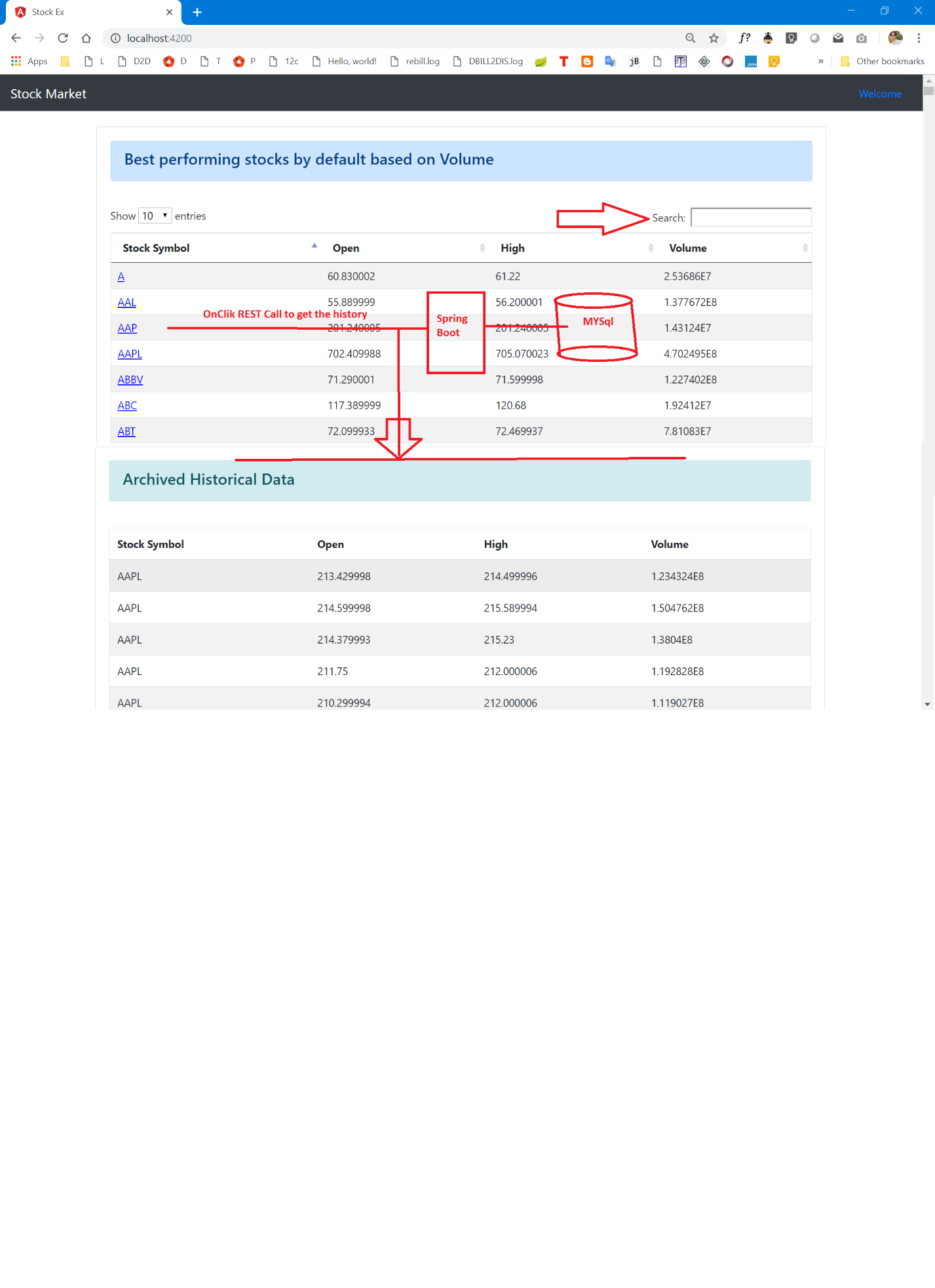


— Your frontend should be able to query historical data for desired stocks.

User can search and get the desired Stock



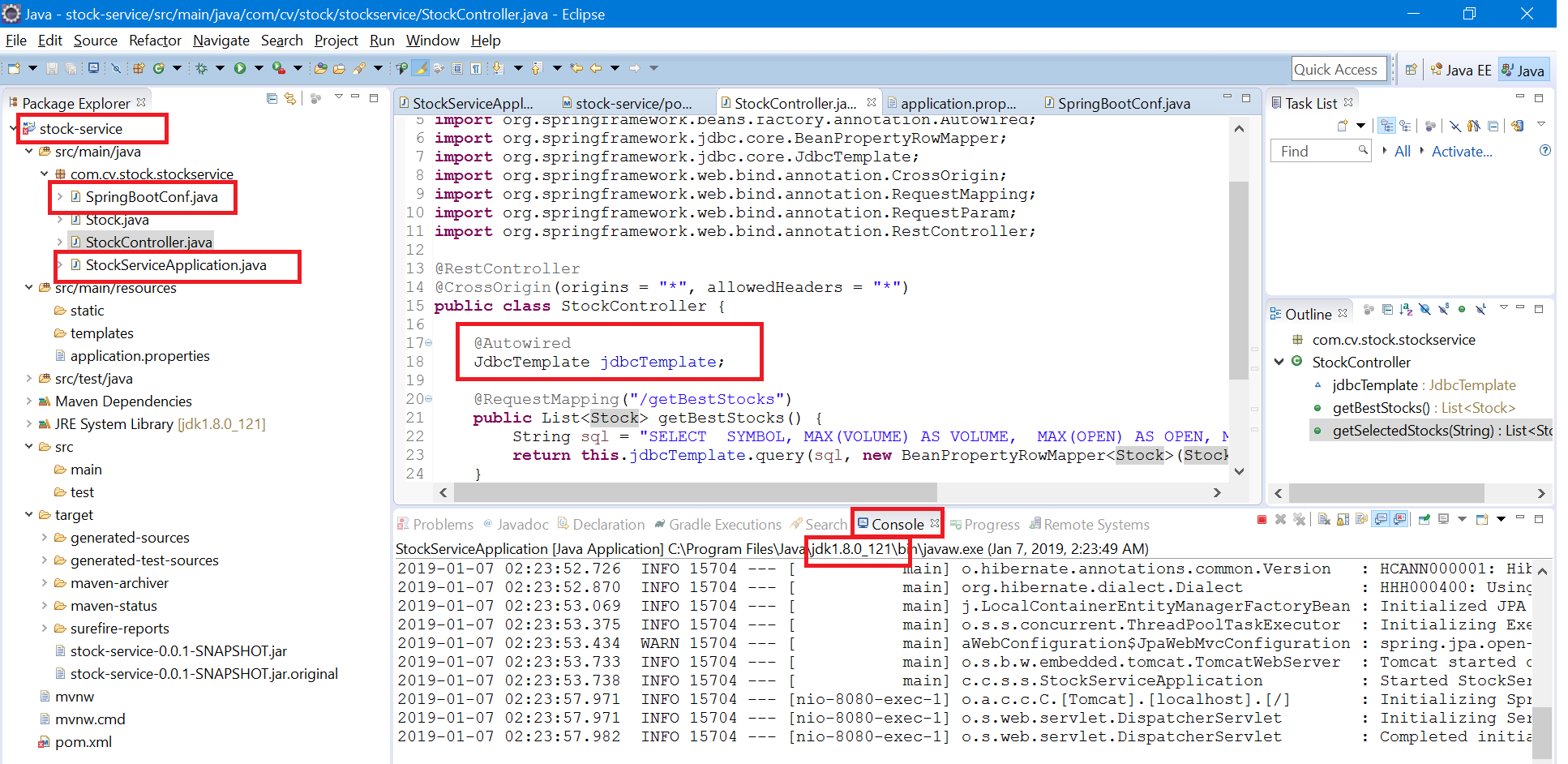
— Your frontend should display detailed company and stock info provided in the dump.



**[Minimum Requirement]**

— Write a backend in tech stack mentioned below which exposes some APIs to render data, which is requested from the frontend.

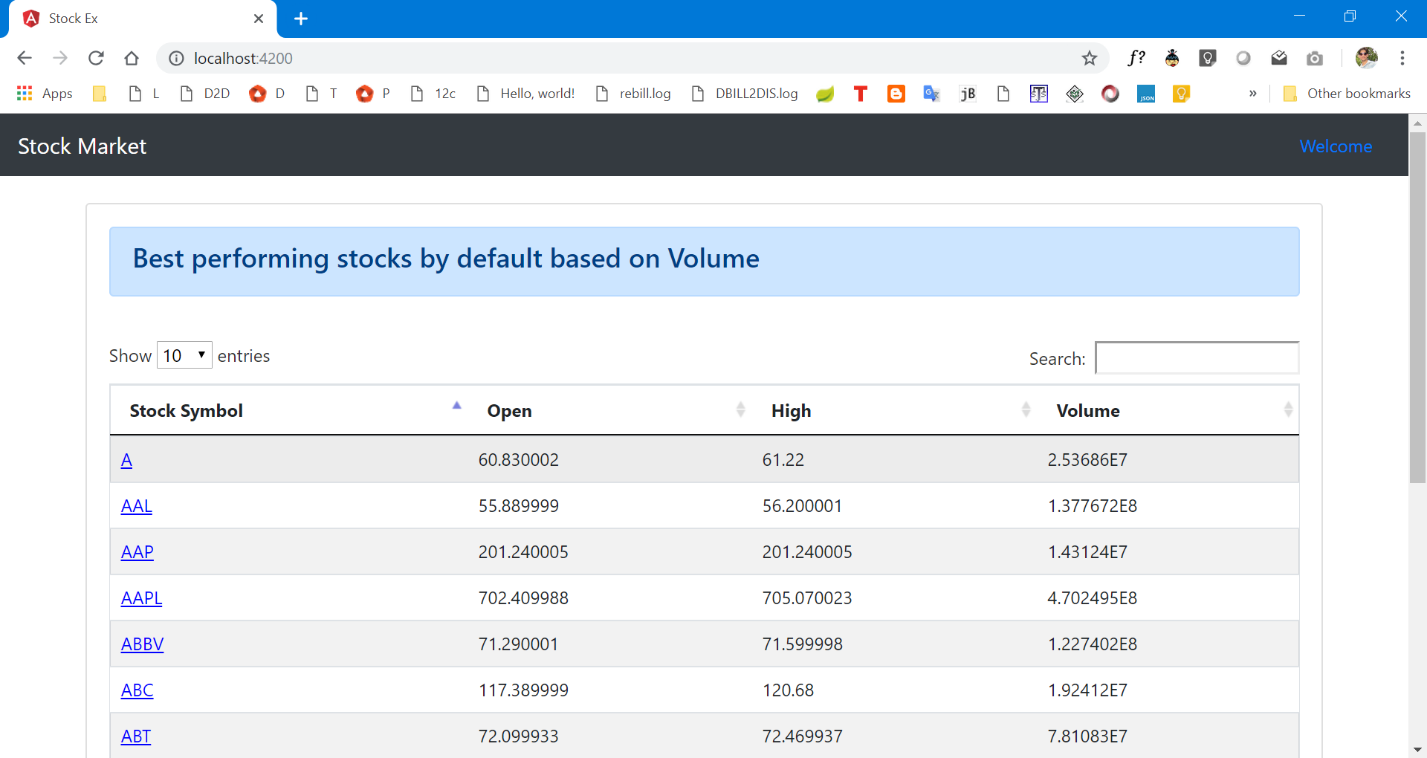
Backend Spring Boot code glimpse to get the data



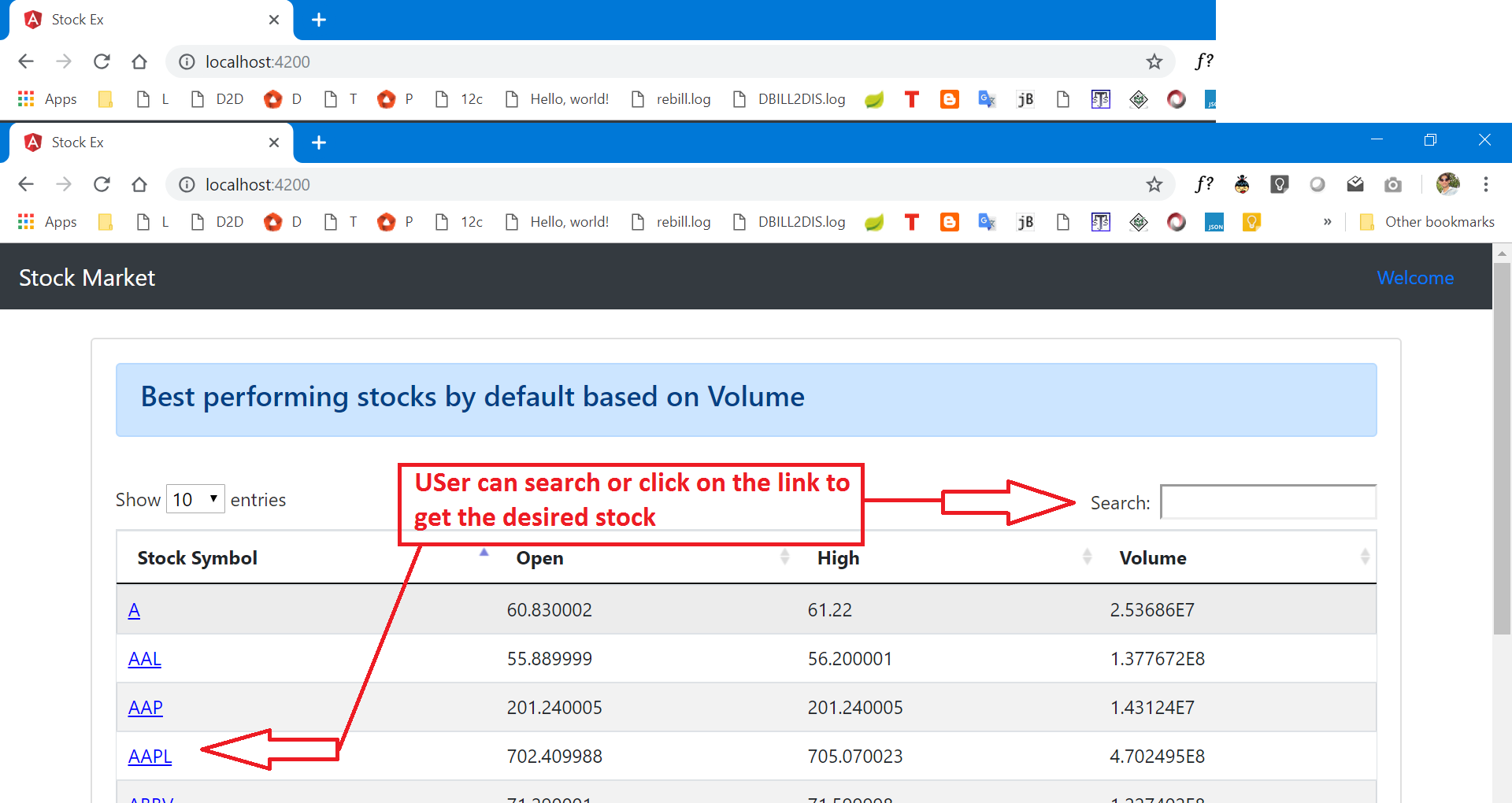
**On Front End**:

— Display best performing stocks by default.

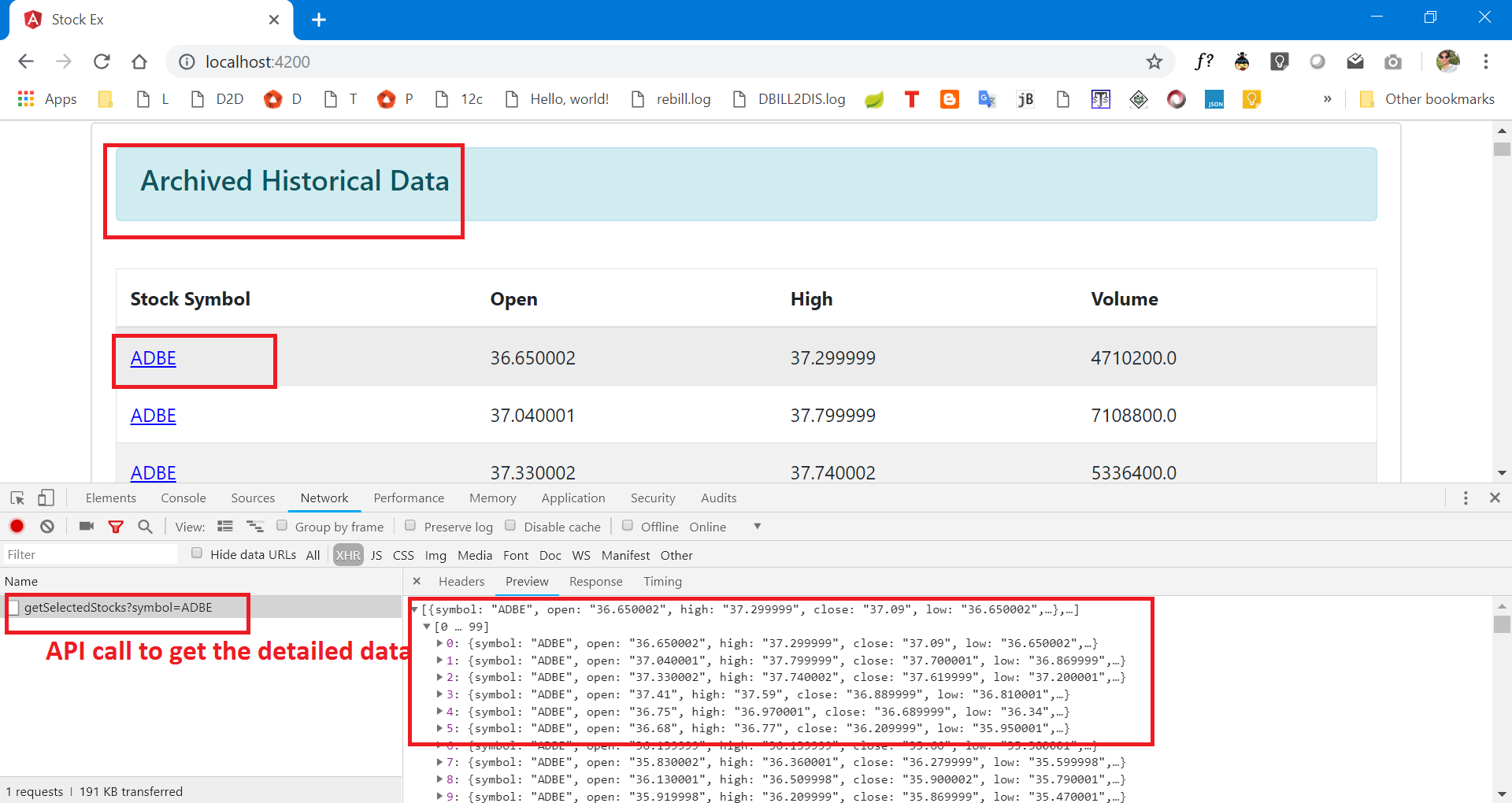
Best performing stock based on openness and Volume



— Provide functionality to call stock details by name or ticker(symbol).



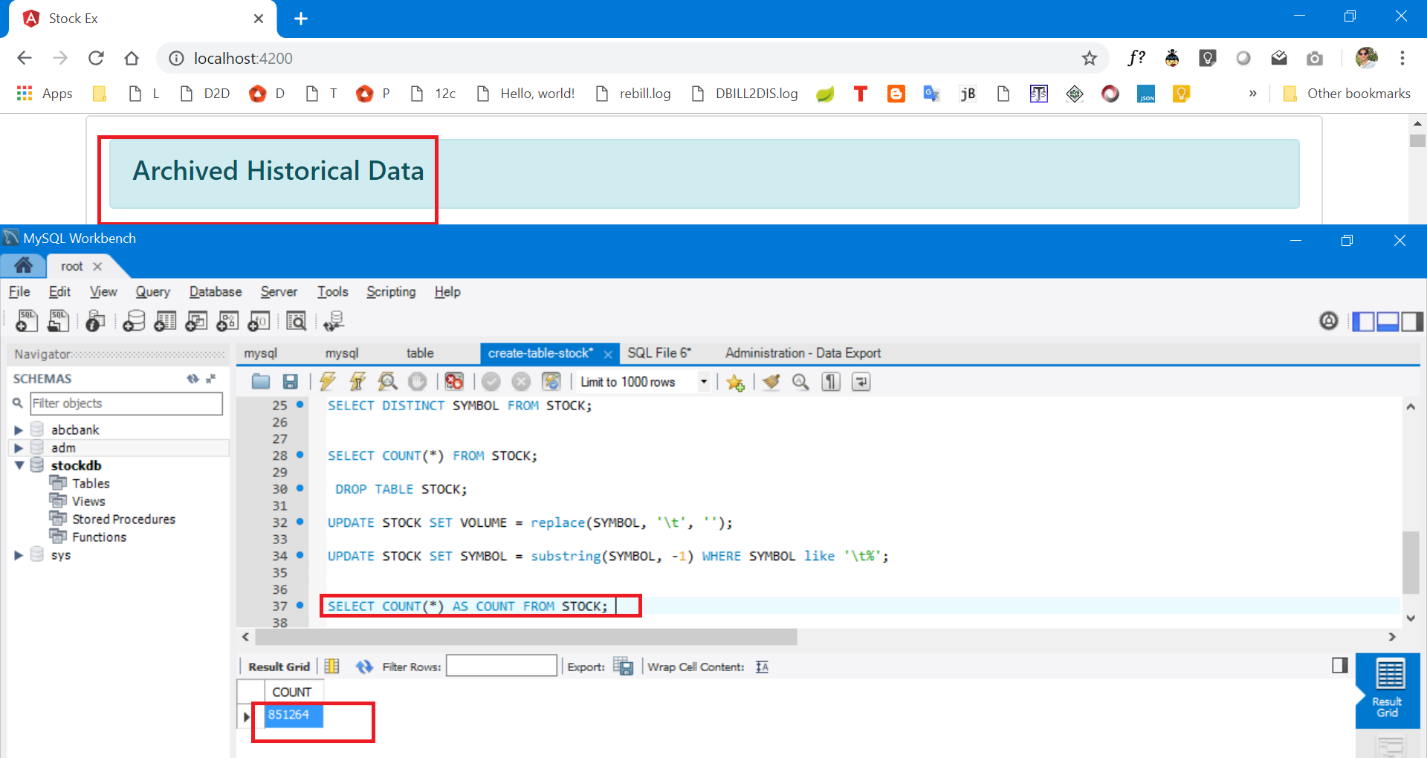
— Make an API call and demonstrate details of stocks.



**On Back end**:

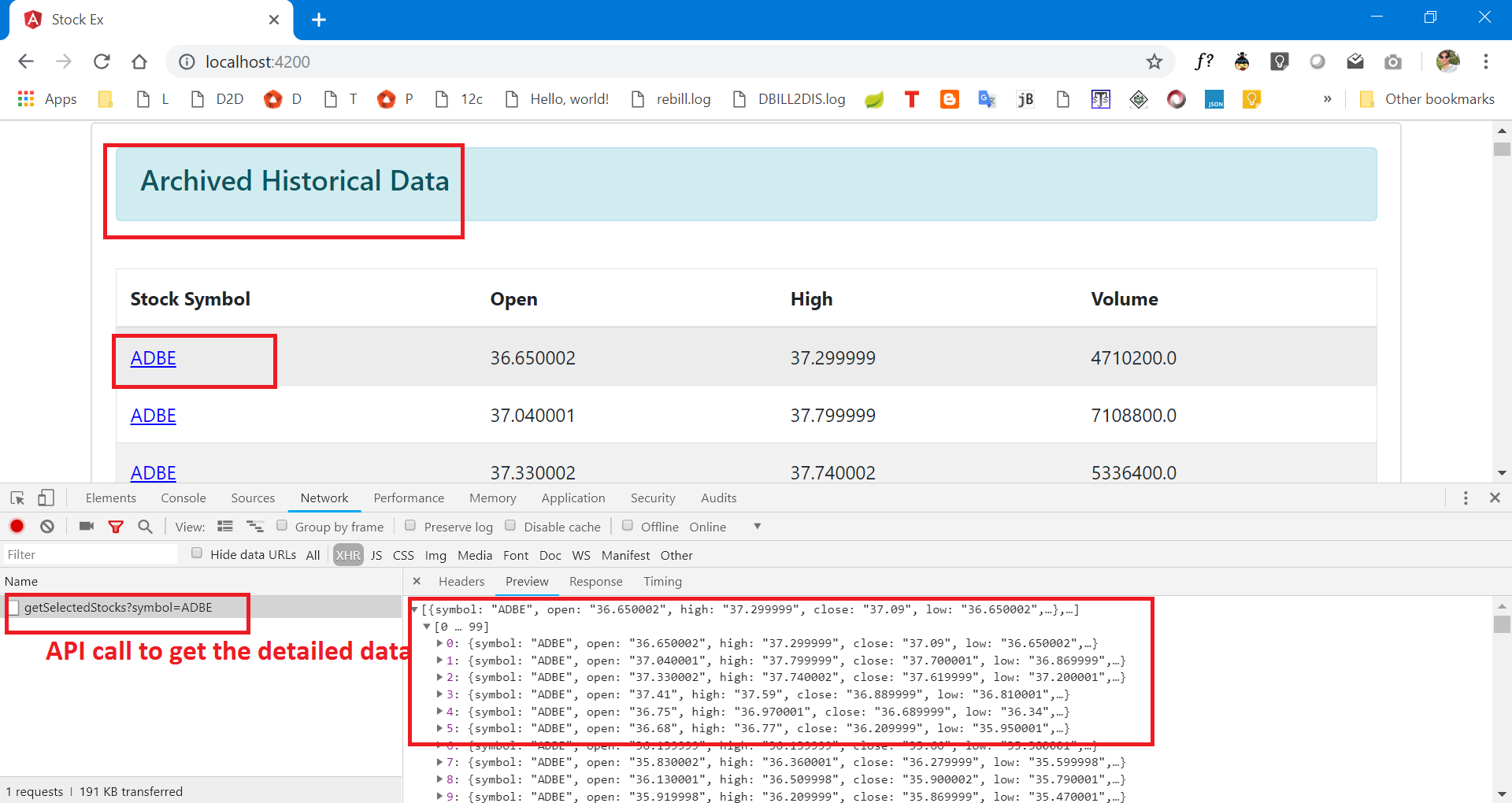
— Store all the data provided in the dump file in your database.

Inserted all the records in the MYSql Database. Also given the dump



— Render all the requested data over network via REST call.

All the requests are made REST call to get the data

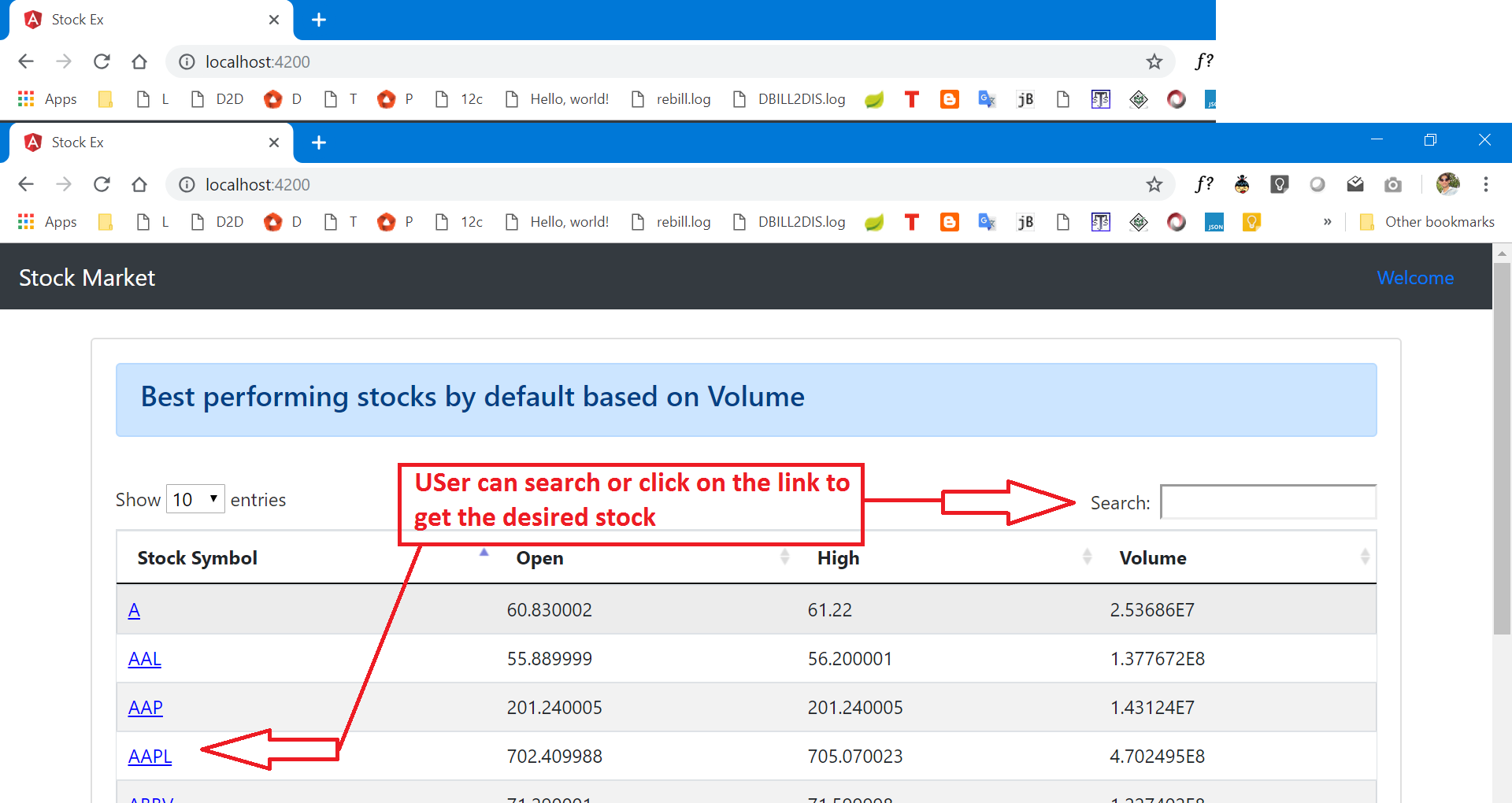


— Zip all your source code, deployment instructions, screenshots and upload them.

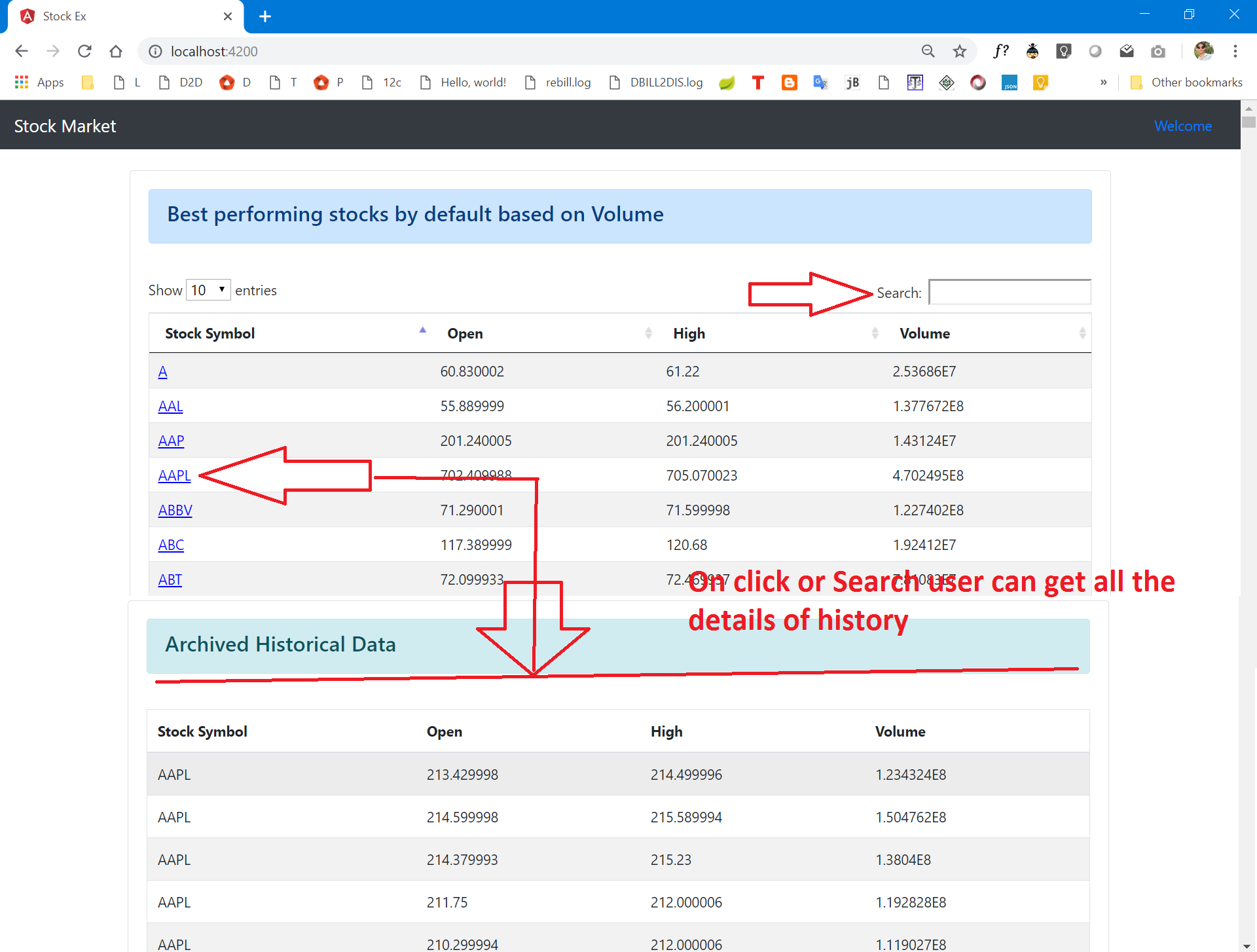
Ziped and uploaded.

**[Extra Work]**

Along with everything from the above level:

— Implement a functionality to fetch stock details by name/ticker.  
  


— When clicked on a listed stock, show detailed information about it. With the detailed history of its value fluctuations over time.



— Use your imagination and add features which would make things easier for end users.

Datatable feature used to make the application userfriendly

— Zip all your source code, deployment instruction, db dump file, screenshots and upload them.

Zipped and uploaded

**[Guide]**

**Stock Archives Data dump**: http://hck.re/CPKVPp

**[Ideal Stack]**

* **Frontend**: Angular, React, VueJS  
  + **Used Angular 4 , Bootstrap and Datatables**
* **Backend**: Java Enterprise, Python - Django, Node - Express, .Net  
  + **Used Spring Boot as service to make REST calls and returns Json data**
* **Db** : Mysql, MongoDB, PostgreSQL
  + **MySql for data storage and retrieval**

Thank You!

Channaveer M Patil