Coding Challenge

The Task

When pharmacists supply medications, a medication string is created for every individual medication supplied. You are part of a development team that has been asked to implement a medication string recorder for pharmacists. You are required to provide the back-end API server that ingests medication strings and can produce some simple statistics on the medications that have been supplied.

Your task is the following:

- Create a simple REST API with two endpoints:
    - one endpoint that accepts the medication string input that's POSTed to it in two separate JSON formats (see end of document for the formats)
    - one endpoint that returns statistical data in a JSON format (you create the appropriate JSON format)
- The statistics must include the following:
    - the total number of medications that have been inputted
    - the total dosage count of all medications
    - the total number of medications by bottle size
    - a list of individual medication IDs and the number of times each individual medication has been supplied

Medication strings have the following format:

```
medicationID_bottleSize_dosageCount;
```

Where:

- `medicationId` is a variable-length string no greater than 20 characters in length that represents the ID of the medicine that is administered
- `bottleSize` is one of *S, M, L, XL, XXL* or *NA*
- `dosageCount` is a 4-digit number (eg. *0025* or *0106* or *3106*)
- `;` is the termination character that enables multiple termination strings to be inputted at the same time (not relevant for JSON format #2)

Notes

- No logins or security is required for the coding challenge.
- Any medication strings that are invalid can just be skipped over and ignored.
- Any JSON posted to the endpoint that doesn't match the required format should receive a 403 response
- No need for a database -- just use an in-memory data structure from which to obtain your statistics
- Feel free to use third-party libraries to help you with completing the challenge.
- You can use any build tools that you're comfortable with
- Provide simple instructions on how we can run up the application in the README
- Posting your code to GitHub or BitBucket just for ease of access.

Assessment Criteria

- The API must be written in a statically-typed object-oriented language (TypeScript counts)
- The API must be accompanied by at least one set of unit tests (for instance, could be unit tests around the medication string parsing)
- The solution should be developed to professional standards, as if the code were to be used and extended. Excessive commenting is not required, however.
- The solution should be deliverable within two to three hours - please do not spend excessive amounts of time on this. We value expediency over elaborate solutions as long as you can justify the approach you take.
- Readability and fluency of your code (including tests) will be highly valued

Example Medication String

```
186FASc73541_M_1058;18673cda541_S_0061;18673541_S_0146;18673cda541
_XL_0056,18896541_M_0055;18896541_XXL_0038;aa1867354cc1_S_0073;186
73541_L_0105;186735412333123121_NA_0073;18673543311_L_0105
```

JSON Format #1 for API Endpoint that Handles the Medication String

```
{ medicationStrings: “186FASc73541_M_1058;18673cda541_S_0061;” }
```

JSON Format #2 for API Endpoint that Handles the Medication String

```
{ medicationStrings: [“186FASc73541_M_1058”, “18673cda541_S_0061”] }
```