GREEDYGAME

BACKEND ASSIGNMENT

Data Collection Tree

One of the most common data structures used around is a tree. You are tasked with building the same which can collect data and aggregate on some dimensions and metrics.

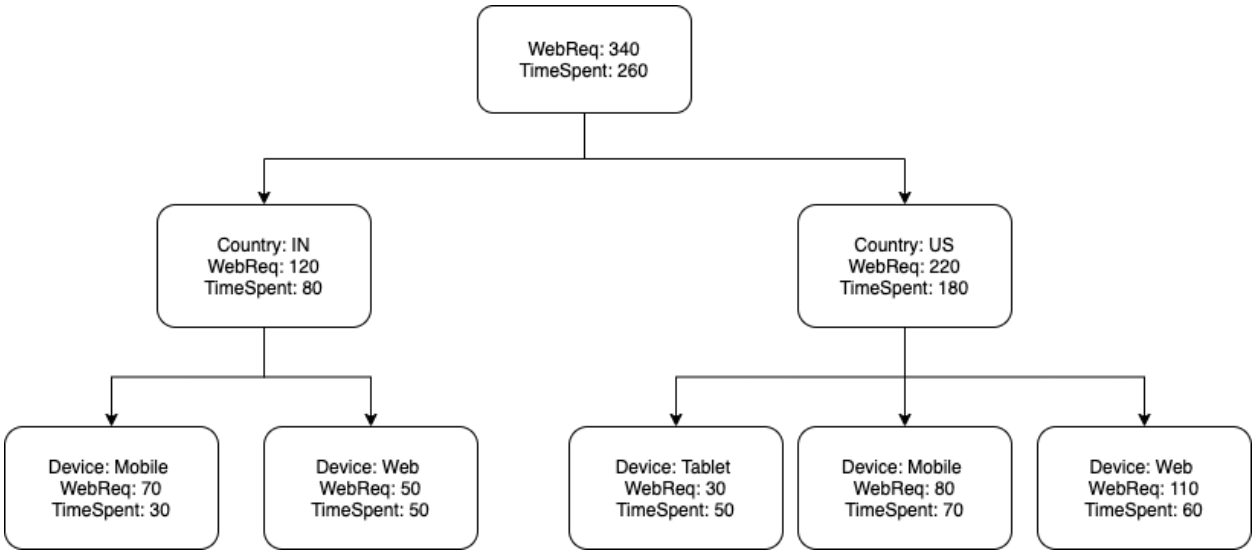
Consider yourself as an administrator for a website which has traffic globally from different devices. In this case, your dimensions are:

1. Country: Represented by a string like IN,US,KR etc
2. Device: Represented by a string like Mobile, Desktop, Tablet

You have a couple of metrics as well on these dimensions. Metrics are the actionable values that a user generates. Some examples of the same are:

1. WebRequests: Represented by an integer denoting the number of times a request has been made to a website
2. TimeSpent: Represented by an integer which denotes the time spent in seconds by users on the website

In order to summarize this data, you have to create a tree in memory which has data split based on different dimensions. For example, consider the following diagram:



If you observe this diagram, you will find that all the nodes contain all metrics. The parent node contains a sum or aggregation of the metrics from the child nodes.

Your task is to create a tree which adds data to this particular structure in a scalable architecture. Every time there is an insertion in the tree, it should not block and update the parent and child nodes to maintain the correct values.

For inserting and querying the data from the tree, you need to expose REST endpoints which take in the following data set:

```
/v1/insert

{
      “dim”: [{
            “key”: “device”,
            “val”: “mobile”
      },
      {
            “key”: “country”,
            “val”: “IN”
      }],
      “metrics”: [{
            “key”: “webreq”,
            “val”: 70
      },
      {
```

```
            “key”: “timespent”,
            “val”: 30
      }]
}
RES: 200 OK



/v1/query

{
      “dim”: [{
            “key”: “country”
            “val”: “US”
      }]
}

RES: 200 OK
Output:
{
      “dim”: [{
            “key”: “country”,
            “val”: “US”
      }],
      “metrics”: [{
            “key”: “webreq”,
            “val”: 220
      },
      {
            “key”: “timespent”,
            “val”: 180
      }]
}
```

You are requested to create a repo which contains the source code along with a set of test cases to run. This service should be containerized with relevant docker files.

FAQs:

1. What happens when an insertion is made with the same dimensions?
   In case the dimensions are the same, the metrics should be added. The idea behind this tree is to aggregate values

2. What if there are not all metrics in a particular insert request?
   If there are not enough metrics, then only the metrics which are present should be updated.