***Input Format Documentation***

Getting Minimum Cost <Url:-> [http://127.0.0.1:8000/getMinimumCost](http://127.0.0.1:8000/getMinimumCost?A=23.32&B=24&D=45&I=89)

The parameters would be quantities A,B,C,D,E,F,G and corresponding to each quantity will be the number of units to be ordered of that quantity:-

**Example:-** http://127.0.0.1:8000/getMinimumCost?A=23&B=24&D=45&I=89

**Answer:-** Rs. 1354.00

**Sample Output Format:-**

{

"MinimumCostRequired": "Rs. 1354.00",

"Total Quantity Ordered From Center C1": "117.00 units",

"Total Quantity Ordered From Center C2": "540.00 units",

"Total Quantity Ordered From Center C3": "178.00 units"

}

***Assumptions Made:-***

If no error encountered then calculation for the minimum cost

taking in consideration that **one vehicle can carry 100 kg at max at one time**

and the *cost of running vehicle is 0-5 kg Rs. 10 and for every additional 5kg Rs 8*

Distance of centers as given from L(customer) :-

*C1 -> L = 3 units*

*C2 -> L = 2.5 units*

*C3 -> L = 2 units*

*C1 -> C2 = 4 units*

*C2 -> C3 = 3 units*

***Order can be given in whole quantity or in multiples of the weight of the product taking in consideration that the centers are all wholesalers and give products in open quantity also(not in multiples of weights of the product).***

***One can’t order quantity in decimal values like 23.32 units***

**RealWorld Scenario**

I have stored the quantity in Json field in Postgres as querying in Json field is much more efficient than querying in whole database.

In real world a there may be many centers with many products and we may not hardcode every product and center in the dictionary as we have done in this problem(as it was given) so storing in database and calculating the minimum cost from database querying is an efficient approach which I have also applied.

We have to consider the cost of fuel so as to see if we have to drop the product or pick up product from another station which every gives the minimum cost in case the distance between the roads(centers and L1 were dynamic).