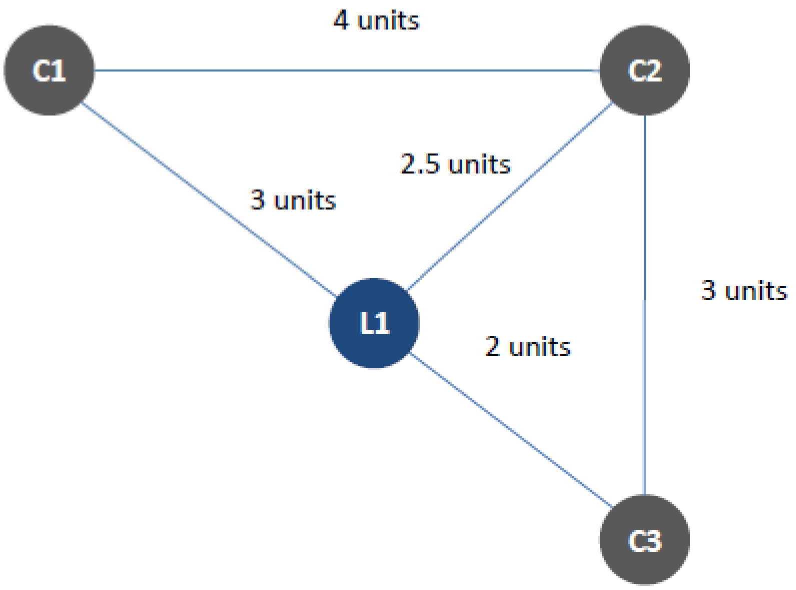
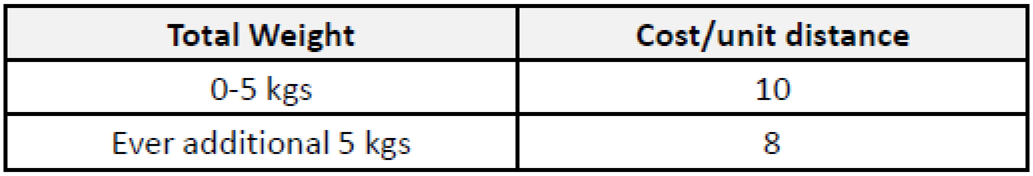
TWF ASSESMENT TASK

Submitted by,
Vedansh Chandra Dwivedi
(vedanshdwivedi0@gmail.com)
5/6/2020 22:37



There are 3 centers C1, C2 and C3 which act as warehouses and stock products in the following way.

Center	Stock Product	Weight
C1	A	3kg
C1	B	2kg
C1	C	8kg
C2	D	12kg
C2	E	25kg
C2	F	15kg
C3	G	500g
C3	H	1kg
C3	I	2kg



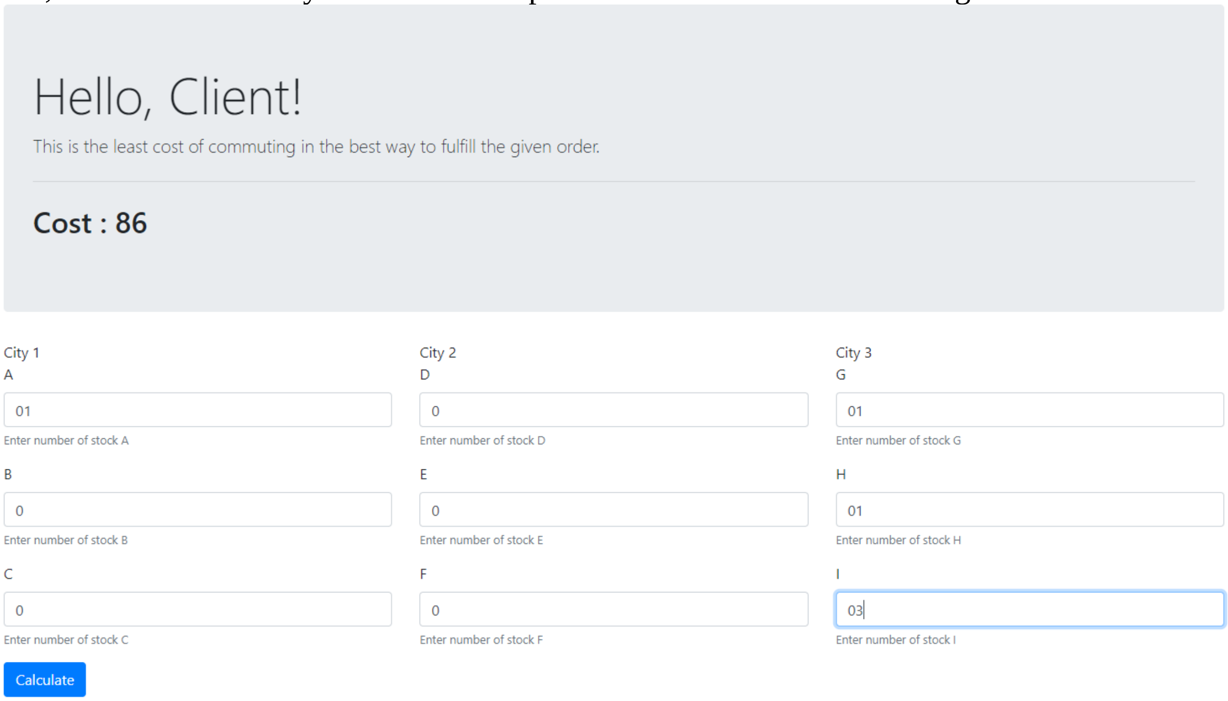
Total Weight	Cost/unit distance
0-5 kgs	10
Ever additional 5 kgs	8

There are three files available:

1. test.py
2. main.py
3. api.py

test.py

The test.py file is hosted online and is available on the link : (https://bit.ly/vedtwfapi). It can be opened on any browser on a smartphone, laptop or desktop. It has a very simple user interface.
It has got text fields for all the stocks available and by default, each stock is assumed to be zero, so that the user only needs to feed input for the stocks whose value is greater than 0.



Hello, Client!

This is the least cost of commuting in the best way to fulfill the given order.

Cost : 86

City 1

A

01

Enter number of stock A

B

0

Enter number of stock B

C

0

Enter number of stock C

City 2

D

0

Enter number of stock D

E

0

Enter number of stock E

F

0

Enter number of stock F

City 3

G

01

Enter number of stock G

H

01

Enter number of stock H

I

03

Enter number of stock I

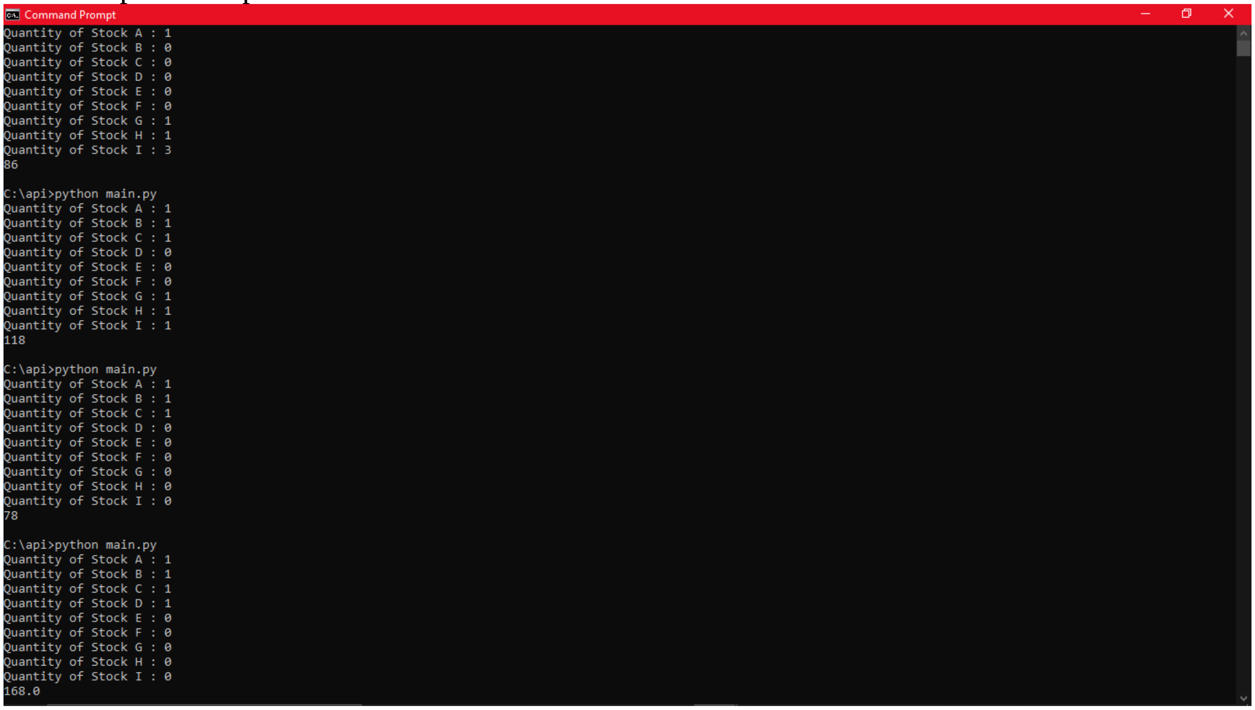
main.py

main.py is simple python program which can be run on any python shell or command line. The program offers a simple UI. This program can be run by going into the command prompt and executing the following command (Ensure that the directory of command prompt is the same where you have the available file)

Command: python main.py

This will execute the program which will then ask about number of each type of stocks required. You can enter the values one by one and upon getting the quantity of all the stocks, it would return the minimum cost of transportation.

An example of output screen is shown below :



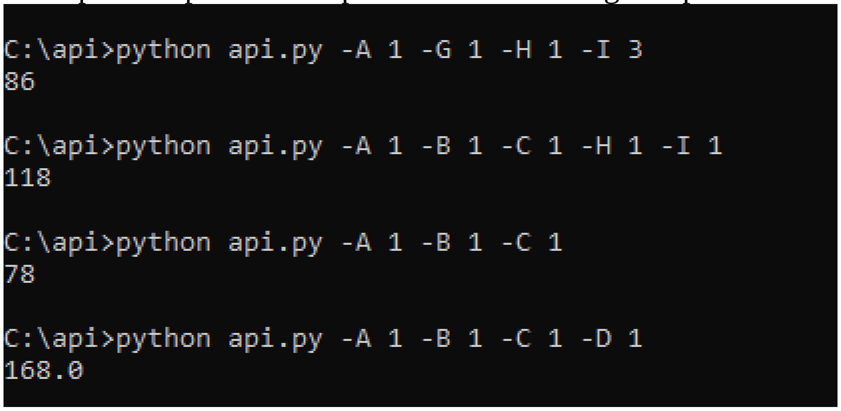
```
Quantity of Stock A : 1
Quantity of Stock B : 0
Quantity of Stock C : 0
Quantity of Stock D : 0
Quantity of Stock E : 0
Quantity of Stock F : 0
Quantity of Stock G : 1
Quantity of Stock H : 1
Quantity of Stock I : 3
86

C:\api>python main.py
Quantity of Stock A : 1
Quantity of Stock B : 1
Quantity of Stock C : 1
Quantity of Stock D : 0
Quantity of Stock E : 0
Quantity of Stock F : 0
Quantity of Stock G : 1
Quantity of Stock H : 1
Quantity of Stock I : 1
118

C:\api>python main.py
Quantity of Stock A : 1
Quantity of Stock B : 1
Quantity of Stock C : 1
Quantity of Stock D : 0
Quantity of Stock E : 0
Quantity of Stock F : 0
Quantity of Stock G : 0
Quantity of Stock H : 0
Quantity of Stock I : 0
78

C:\api>python main.py
Quantity of Stock A : 1
Quantity of Stock B : 1
Quantity of Stock C : 1
Quantity of Stock D : 1
Quantity of Stock E : 0
Quantity of Stock F : 0
Quantity of Stock G : 0
Quantity of Stock H : 0
Quantity of Stock I : 0
168.0
```

api.py

api.py is the API that is useful when the program needs to be run for large number of cases. The api.py file can be run in the same way as we ran man.py but for ‘api.py’, we need to provide positional arguments to give inputs. **By default, the program assumes that all the stocks are required in 0 quantity and thus no commutation is required**. The syntax for running api.py is:

Command: python api.py –[Stock Name] [quantity]

Example of inputs and outputs is shown in the given picture



```
C:\api>python api.py -A 1 -G 1 -H 1 -I 3
86

C:\api>python api.py -A 1 -B 1 -C 1 -H 1 -I 1
118

C:\api>python api.py -A 1 -B 1 -C 1
78

C:\api>python api.py -A 1 -B 1 -C 1 -D 1
168.0
```