In [94]:
```
import pandas as pd
from pulp import *
```

In [95]:
```
rissh = pd.read_excel("diet - medium.xls")
rissh
```

Out[95]:

	Foods	Price/Serving	Serving Size	Calories	Cholesterol (mg)	Total_Fat (g)	Sodium (mg)	Carbohydrates (g)	Dietary_Fiber (g)	Protein (g)	Vit_A (IU)	Vit_C (IU)
0	Frozen Broccoli	0.48	10 Oz Pkg	73.8	0.0	0.8	68.2	13.6	8.5	8.0	5867.4	160.2
1	Frozen Corn	0.54	1/2 Cup	72.2	0.0	0.6	2.5	17.1	2.0	2.5	106.6	5.2
2	Raw Lettuce Iceberg	0.06	1 Leaf	2.6	0.0	0.0	1.8	0.4	0.3	0.2	66.0	0.8
3	Baked Potatoes	0.18	1/2 Cup	171.5	0.0	0.2	15.2	39.9	3.2	3.7	0.0	15.6
4	Tofu	0.93	1/4 block	88.2	0.0	5.5	8.1	2.2	1.4	9.4	98.6	0.1
5	Roasted Chicken	2.52	1 lb chicken	277.4	129.9	10.8	125.6	0.0	0.0	42.2	77.4	0.0
6	Spaghetti W/ Sauce	2.34	1 1/2 Cup	358.2	0.0	12.3	1237.1	58.3	11.6	8.2	3055.2	27.9
7	Raw Apple	0.72	1 Fruit,3/Lb,Wo/Rf	81.4	0.0	0.5	0.0	21.0	3.7	0.3	73.1	7.9
8	Banana	0.45	1 Fruit,Wo/Skn&Seeds	104.9	0.0	0.5	1.1	26.7	2.7	1.2	92.3	10.4
9	Wheat Bread	0.15	1 Sl	65.0	0.0	1.0	134.5	12.4	1.3	2.2	0.0	0.0
10	White Bread	0.18	1 Sl	65.0	0.0	1.0	132.5	11.8	1.1	2.3	0.0	0.0
11	Oatmeal Cookies	0.27	1 Cookie	81.0	0.0	3.3	68.9	12.4	0.6	1.1	2.9	0.1
12	Apple Pie	0.48	1 Oz	67.2	0.0	3.1	75.4	9.6	0.5	0.5	35.2	0.9

	Foods	Price/Serving	Serving Size	Calories	Cholesterol (mg)	Total_Fat (g)	Sodium (mg)	Carbohydrates (g)	Dietary_Fiber (g)	Protein (g)	Vit_A (IU)	Vit_C (IU)
13	Scrambled Eggs	0.33	1 Egg	99.6	211.2	7.3	168.0	1.3	0.0	6.7	409.2	0.1
14	Turkey Bologna	0.45	1 Oz	56.4	28.1	4.3	248.9	0.3	0.0	3.9	0.0	0.0
15	Beef Frankfurter	0.81	1 Frankfurter	141.8	27.4	12.8	461.7	0.8	0.0	5.4	0.0	10.8
16	Chocolate Chip Cookies	0.09	1 Cookie	78.1	5.1	4.5	57.8	9.3	0.0	0.9	101.8	0.0
17	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
18	NaN	NaN	Minimum daily intake	800.0	30.0	20.0	800.0	130.0	60.0	100.0	1000.0	400.0
19	NaN	NaN	Maximum daily intake	1300.0	240.0	50.0	2000.0	200.0	125.0	150.0	10000.0	5000.0
20	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
21	NaN	NaN	NaN	NaN	811.6	NaN	NaN	NaN	NaN	NaN	NaN	NaN
22	NaN	NaN	NaN	NaN	27.4	NaN	NaN	NaN	NaN	NaN	NaN	NaN

In [96]:
```
df = rissh.dropna()
df
```

Out[96]:

	Foods	Price/Serving	Serving Size	Calories	Cholesterol (mg)	Total_Fat (g)	Sodium (mg)	Carbohydrates (g)	Dietary_Fiber (g)	Protein (g)	Vit_A (IU)	Vit_C (IU)	Ca
0	Frozen Broccoli	0.48	10 Oz Pkg	73.8	0.0	0.8	68.2	13.6	8.5	8.0	5867.4	160.2	
1	Frozen Corn	0.54	1/2 Cup	72.2	0.0	0.6	2.5	17.1	2.0	2.5	106.6	5.2	
2	Raw Lettuce Iceberg	0.06	1 Leaf	2.6	0.0	0.0	1.8	0.4	0.3	0.2	66.0	0.8	

	Foods	Price/Serving	Serving Size	Calories	Cholesterol (mg)	Total_Fat (g)	Sodium (mg)	Carbohydrates (g)	Dietary_Fiber (g)	Protein (g)	Vit_A (IU)	Vit_C (IU)	Ca
3	Baked Potatoes	0.18	1/2 Cup	171.5	0.0	0.2	15.2	39.9	3.2	3.7	0.0	15.6	
4	Tofu	0.93	1/4 block	88.2	0.0	5.5	8.1	2.2	1.4	9.4	98.6	0.1	
5	Roasted Chicken	2.52	1 lb chicken	277.4	129.9	10.8	125.6	0.0	0.0	42.2	77.4	0.0	
6	Spaghetti W/ Sauce	2.34	1 1/2 Cup	358.2	0.0	12.3	1237.1	58.3	11.6	8.2	3055.2	27.9	
7	Raw Apple	0.72	1 Fruit,3/Lb,Wo/Rf	81.4	0.0	0.5	0.0	21.0	3.7	0.3	73.1	7.9	
8	Banana	0.45	1 Fruit,Wo/Skn&Seeds	104.9	0.0	0.5	1.1	26.7	2.7	1.2	92.3	10.4	
9	Wheat Bread	0.15	1 Sl	65.0	0.0	1.0	134.5	12.4	1.3	2.2	0.0	0.0	
10	White Bread	0.18	1 Sl	65.0	0.0	1.0	132.5	11.8	1.1	2.3	0.0	0.0	
11	Oatmeal Cookies	0.27	1 Cookie	81.0	0.0	3.3	68.9	12.4	0.6	1.1	2.9	0.1	
12	Apple Pie	0.48	1 Oz	67.2	0.0	3.1	75.4	9.6	0.5	0.5	35.2	0.9	
13	Scrambled Eggs	0.33	1 Egg	99.6	211.2	7.3	168.0	1.3	0.0	6.7	409.2	0.1	
14	Turkey Bologna	0.45	1 Oz	56.4	28.1	4.3	248.9	0.3	0.0	3.9	0.0	0.0	
15	Beef Frankfurter	0.81	1 Frankfurter	141.8	27.4	12.8	461.7	0.8	0.0	5.4	0.0	10.8	
16	Chocolate Chip Cookies	0.09	1 Cookie	78.1	5.1	4.5	57.8	9.3	0.0	0.9	101.8	0.0	

In [97]:
```
# Create the 'prob' variable to contain the problem data
diet_model = LpProblem("Simple Diet Problem",LpMinimize)
```

```
C:\Users\U.R Computer\anaconda\lib\site-packages\pulp\pulp.py:1199: UserWarning: Spaces are not permitted in the nam
```

```
e. Converted to '_'
  warnings.warn("Spaces are not permitted in the name. Converted to '_'")
```

In [98]:
```
# Creates a list of the Ingredients
food_items = list(df['Foods'])
food_items
```

Out[98]:
```
['Frozen Broccoli',
 'Frozen Corn',
 'Raw Lettuce Iceberg',
 ' Baked Potatoes',
 'Tofu',
 'Roasted Chicken',
 'Spaghetti W/ Sauce',
 'Raw Apple',
 'Banana',
 'Wheat Bread',
 'White Bread',
 'Oatmeal Cookies',
 'Apple Pie',
 'Scrambled Eggs',
 'Turkey Bologna',
 'Beef Frankfurter',
 'Chocolate Chip Cookies']
```

In [99]:
```
# Create a dictinary of costs for all food items
costs = dict(zip(food_items,df['Price/Serving']))

# Create a dictionary of calories for all food items
calories = dict(zip(food_items,df['Calories']))

# Create a dictionary of total fat for all food items
fat = dict(zip(food_items,df['Total_Fat (g)']))

# Create a dictionary of carbohydrates for all food items
carbs = dict(zip(food_items,df['Carbohydrates (g)']))

# Create a dictionary of fibers for all food items
fiber = dict(zip(food_items,df['Dietary_Fiber (g)']))

# Create a dictionary of proteins for all food items
protein = dict(zip(food_items,df['Protein (g)']))
```

In [100…
```
costs
```

Out[100…
```
{'Frozen Broccoli': 0.48,
 'Frozen Corn': 0.54,
 'Raw Lettuce Iceberg': 0.06,
 ' Baked Potatoes': 0.18,
 'Tofu': 0.9299999999999999,
 'Roasted Chicken': 2.52,
 'Spaghetti W/ Sauce': 2.34,
 'Raw Apple': 0.72,
 'Banana': 0.44999999999999996,
 'Wheat Bread': 0.15000000000000002,
 'White Bread': 0.18,
 'Oatmeal Cookies': 0.27,
 'Apple Pie': 0.48,
 'Scrambled Eggs': 0.33,
 'Turkey Bologna': 0.44999999999999996,
 'Beef Frankfurter': 0.81,
 'Chocolate Chip Cookies': 0.09}
```

In [101…
```
food_vars = LpVariable.dicts("Food",food_items,lowBound=0,cat='Continuous')
```

In [102…
```
diet_model += lpSum([costs[i]*food_vars[i] for i in food_items])
```

In [103…
```
min_max = rissh.iloc[18:20,:]
min_max
```

Out[103…

	Foods	Price/Serving	Serving Size	Calories	Cholesterol (mg)	Total_Fat (g)	Sodium (mg)	Carbohydrates (g)	Dietary_Fiber (g)	Protein (g)	Vit_A (IU)	Vit_C (IU)	Calcium (mg)	Iron (mg)
18	NaN	NaN	Minimum daily intake	800.0	30.0	20.0	800.0	130.0	60.0	100.0	1000.0	400.0	700.0	10.0
19	NaN	NaN	Maximum daily intake	1300.0	240.0	50.0	2000.0	200.0	125.0	150.0	10000.0	5000.0	1500.0	40.0

In [104…
```
diet_model += lpSum([calories[f] * food_vars[f] for f in food_items]) >= 800.0
diet_model += lpSum([calories[f] * food_vars[f] for f in food_items]) <= 1300.0
```

In [105…

```
# Fat
diet_model += lpSum([fat[f] * food_vars[f] for f in food_items]) >= 20.0, "FatMinimum"
diet_model += lpSum([fat[f] * food_vars[f] for f in food_items]) <= 50.0, "FatMaximum"

# Carbs
diet_model += lpSum([carbs[f] * food_vars[f] for f in food_items]) >= 130.0, "CarbsMinimum"
diet_model += lpSum([carbs[f] * food_vars[f] for f in food_items]) <= 200.0, "CarbsMaximum"

# Fiber
diet_model += lpSum([fiber[f] * food_vars[f] for f in food_items]) >= 60.0, "FiberMinimum"
diet_model += lpSum([fiber[f] * food_vars[f] for f in food_items]) <= 125.0, "FiberMaximum"

# Protein
diet_model += lpSum([protein[f] * food_vars[f] for f in food_items]) >= 100.0, "ProteinMinimum"
diet_model += lpSum([protein[f] * food_vars[f] for f in food_items]) <= 150.0, "ProteinMaximum"
```

In [106…

```
diet_model
```

Out[106…

```
Simple_Diet_Problem:
MINIMIZE
0.48*Food_Apple_Pie + 0.44999999999999996*Food_Banana + 0.81*Food_Beef_Frankfurter + 0.09*Food_Chocolate_Chip_Cookies
+ 0.48*Food_Frozen_Broccoli + 0.54*Food_Frozen_Corn + 0.27*Food_Oatmeal_Cookies + 0.72*Food_Raw_Apple + 0.06*Food_Raw
_Lettuce_Iceberg + 2.52*Food_Roasted_Chicken + 0.33*Food_Scrambled_Eggs + 2.34*Food_Spaghetti_W__Sauce + 0.9299999999
999999*Food_Tofu + 0.44999999999999996*Food_Turkey_Bologna + 0.15000000000000002*Food_Wheat_Bread + 0.18*Food_White_B
read + 0.18*Food__Baked_Potatoes + 0.0
SUBJECT TO
_C1: 67.2 Food_Apple_Pie + 104.9 Food_Banana + 141.8 Food_Beef_Frankfurter
 + 78.1 Food_Chocolate_Chip_Cookies + 73.8 Food_Frozen_Broccoli
 + 72.2 Food_Frozen_Corn + 81 Food_Oatmeal_Cookies + 81.4 Food_Raw_Apple
 + 2.6 Food_Raw_Lettuce_Iceberg + 277.4 Food_Roasted_Chicken
 + 99.6 Food_Scrambled_Eggs + 358.2 Food_Spaghetti_W__Sauce + 88.2 Food_Tofu
 + 56.4 Food_Turkey_Bologna + 65 Food_Wheat_Bread + 65 Food_White_Bread
 + 171.5 Food__Baked_Potatoes >= 800

_C2: 67.2 Food_Apple_Pie + 104.9 Food_Banana + 141.8 Food_Beef_Frankfurter
 + 78.1 Food_Chocolate_Chip_Cookies + 73.8 Food_Frozen_Broccoli
 + 72.2 Food_Frozen_Corn + 81 Food_Oatmeal_Cookies + 81.4 Food_Raw_Apple
 + 2.6 Food_Raw_Lettuce_Iceberg + 277.4 Food_Roasted_Chicken
 + 99.6 Food_Scrambled_Eggs + 358.2 Food_Spaghetti_W__Sauce + 88.2 Food_Tofu
 + 56.4 Food_Turkey_Bologna + 65 Food_Wheat_Bread + 65 Food_White_Bread
 + 171.5 Food__Baked_Potatoes <= 1300

FatMinimum: 3.1 Food_Apple_Pie + 0.5 Food_Banana + 12.8 Food_Beef_Frankfurter
```

```
 + 4.5 Food_Chocolate_Chip_Cookies + 0.8 Food_Frozen_Broccoli
 + 0.6 Food_Frozen_Corn + 3.3 Food_Oatmeal_Cookies + 0.5 Food_Raw_Apple
 + 10.8 Food_Roasted_Chicken + 7.3 Food_Scrambled_Eggs
 + 12.3 Food_Spaghetti_W__Sauce + 5.5 Food_Tofu + 4.3 Food_Turkey_Bologna
 + Food_Wheat_Bread + Food_White_Bread + 0.2 Food__Baked_Potatoes >= 20

FatMaximum: 3.1 Food_Apple_Pie + 0.5 Food_Banana + 12.8 Food_Beef_Frankfurter
 + 4.5 Food_Chocolate_Chip_Cookies + 0.8 Food_Frozen_Broccoli
 + 0.6 Food_Frozen_Corn + 3.3 Food_Oatmeal_Cookies + 0.5 Food_Raw_Apple
 + 10.8 Food_Roasted_Chicken + 7.3 Food_Scrambled_Eggs
 + 12.3 Food_Spaghetti_W__Sauce + 5.5 Food_Tofu + 4.3 Food_Turkey_Bologna
 + Food_Wheat_Bread + Food_White_Bread + 0.2 Food__Baked_Potatoes <= 50

CarbsMinimum: 9.6 Food_Apple_Pie + 26.7 Food_Banana
 + 0.8 Food_Beef_Frankfurter + 9.3 Food_Chocolate_Chip_Cookies
 + 13.6 Food_Frozen_Broccoli + 17.1 Food_Frozen_Corn
 + 12.4 Food_Oatmeal_Cookies + 21 Food_Raw_Apple
 + 0.4 Food_Raw_Lettuce_Iceberg + 1.3 Food_Scrambled_Eggs
 + 58.3 Food_Spaghetti_W__Sauce + 2.2 Food_Tofu + 0.3 Food_Turkey_Bologna
 + 12.4 Food_Wheat_Bread + 11.8 Food_White_Bread + 39.9 Food__Baked_Potatoes
 >= 130

CarbsMaximum: 9.6 Food_Apple_Pie + 26.7 Food_Banana
 + 0.8 Food_Beef_Frankfurter + 9.3 Food_Chocolate_Chip_Cookies
 + 13.6 Food_Frozen_Broccoli + 17.1 Food_Frozen_Corn
 + 12.4 Food_Oatmeal_Cookies + 21 Food_Raw_Apple
 + 0.4 Food_Raw_Lettuce_Iceberg + 1.3 Food_Scrambled_Eggs
 + 58.3 Food_Spaghetti_W__Sauce + 2.2 Food_Tofu + 0.3 Food_Turkey_Bologna
 + 12.4 Food_Wheat_Bread + 11.8 Food_White_Bread + 39.9 Food__Baked_Potatoes
 <= 200

FiberMinimum: 0.5 Food_Apple_Pie + 2.7 Food_Banana + 8.5 Food_Frozen_Broccoli
 + 2 Food_Frozen_Corn + 0.6 Food_Oatmeal_Cookies + 3.7 Food_Raw_Apple
 + 0.3 Food_Raw_Lettuce_Iceberg + 11.6 Food_Spaghetti_W__Sauce + 1.4 Food_Tofu
 + 1.3 Food_Wheat_Bread + 1.1 Food_White_Bread + 3.2 Food__Baked_Potatoes
 >= 60

FiberMaximum: 0.5 Food_Apple_Pie + 2.7 Food_Banana + 8.5 Food_Frozen_Broccoli
 + 2 Food_Frozen_Corn + 0.6 Food_Oatmeal_Cookies + 3.7 Food_Raw_Apple
 + 0.3 Food_Raw_Lettuce_Iceberg + 11.6 Food_Spaghetti_W__Sauce + 1.4 Food_Tofu
 + 1.3 Food_Wheat_Bread + 1.1 Food_White_Bread + 3.2 Food__Baked_Potatoes
 <= 125

ProteinMinimum: 0.5 Food_Apple_Pie + 1.2 Food_Banana
 + 5.4 Food_Beef_Frankfurter + 0.9 Food_Chocolate_Chip_Cookies
```

```
 + 8 Food_Frozen_Broccoli + 2.5 Food_Frozen_Corn + 1.1 Food_Oatmeal_Cookies
 + 0.3 Food_Raw_Apple + 0.2 Food_Raw_Lettuce_Iceberg
 + 42.2 Food_Roasted_Chicken + 6.7 Food_Scrambled_Eggs
 + 8.2 Food_Spaghetti_W__Sauce + 9.4 Food_Tofu + 3.9 Food_Turkey_Bologna
 + 2.2 Food_Wheat_Bread + 2.3 Food_White_Bread + 3.7 Food__Baked_Potatoes
 >= 100

ProteinMaximum: 0.5 Food_Apple_Pie + 1.2 Food_Banana
 + 5.4 Food_Beef_Frankfurter + 0.9 Food_Chocolate_Chip_Cookies
 + 8 Food_Frozen_Broccoli + 2.5 Food_Frozen_Corn + 1.1 Food_Oatmeal_Cookies
 + 0.3 Food_Raw_Apple + 0.2 Food_Raw_Lettuce_Iceberg
 + 42.2 Food_Roasted_Chicken + 6.7 Food_Scrambled_Eggs
 + 8.2 Food_Spaghetti_W__Sauce + 9.4 Food_Tofu + 3.9 Food_Turkey_Bologna
 + 2.2 Food_Wheat_Bread + 2.3 Food_White_Bread + 3.7 Food__Baked_Potatoes
 <= 150

VARIABLES
Food_Apple_Pie Continuous
Food_Banana Continuous
Food_Beef_Frankfurter Continuous
Food_Chocolate_Chip_Cookies Continuous
Food_Frozen_Broccoli Continuous
Food_Frozen_Corn Continuous
Food_Oatmeal_Cookies Continuous
Food_Raw_Apple Continuous
Food_Raw_Lettuce_Iceberg Continuous
Food_Roasted_Chicken Continuous
Food_Scrambled_Eggs Continuous
Food_Spaghetti_W__Sauce Continuous
Food_Tofu Continuous
Food_Turkey_Bologna Continuous
Food_Wheat_Bread Continuous
Food_White_Bread Continuous
Food__Baked_Potatoes Continuous
```

In [107…
```
# The problem is solved using PuLP's choice of Solver
diet_model.solve()
```

Out[107…
```
1
```

In [108…
```
# The status of the solution is printed to the screen
print("Status:", LpStatus[diet_model.status])
```

```
Status: Optimal
```

In [109…
```
print("the optimal (least cost) balanced diet consists of\n"+"-"*100)
for v in diet_model.variables():
    if v.varValue>0:
        print(v.name, "=", v.varValue)
```

```
the optimal (least cost) balanced diet consists of
----------------------------------------------------------------------------------------------------
Food_Frozen_Broccoli = 6.9242113
Food_Scrambled_Eggs = 6.060891
Food__Baked_Potatoes = 1.0806324
```

In [110…
```
print("The total cost of this balanced diet is: ${}".format(round(value(diet_model.objective),2)))
```

```
The total cost of this balanced diet is: $5.52
```

In [ ]:
```
```