

COVID-19 Healthy Diet



By Neha Thota

Project Type: Database Project

Date: 08/06/2021

Course Name: Managing Information Resources ISM4300

Project Report

Overview

COVID-19 or coronavirus disease has spread worldwide leading to an ongoing pandemic. It is spreading at an exponential rate. It is not only contagious but is also deadly. In order to protect themselves, people are doing multiple things like wearing masks, staying at least 6 feet away from others, and cleaning hands often with soap and water or sanitizer. The other major thing that we must do in order to protect ourselves from this deadly virus is to monitor our health daily.

Introduction

“In the past couple of months, we’ve witnessed doctors, nurses, paramedics and thousands of medical workers putting their lives on the frontline to save patients who are infected” (Ren). We must also be responsible, take a step forward, and protect ourselves and our families by following a healthy diet with proper nutrition and macros. “The USDA Center for Nutrition Policy and Promotion recommends a very simple daily diet intake guideline: 30% grains, 40% vegetables, 10% fruits, and 20% protein” (Ren).

The project I will be choosing is a database project. I will be using a five real-world datasets from www.kaggle.com. These datasets contain information about different categories of foods in different countries, including energy intake (kcal), fat quantity, protein quantity, food supply quantity (kg), and descriptions (types of healthy foods in those categories). The datasets also contain information about obesity, undernourished rates, and the number of COVID-19 cases of the population around the world. These datasets let us compare our country’s obesity, undernourished, and COVID rates to a lower rated country’s and also let us gain knowledge about the macros and nutrients in their food items.

Graphic User Interface

I made the below form using Visual Studio. The end-user types in his/her personal information, which include first and last name, email address, age, sex, country he/she is living in, food category, allergies he/she has, and other preferences.

The diet group box will give details depending on the food category and country of the user. These details include the average food energy intake, food quantity, fat quantity, and recommended foods in the category. The details also include COVID deaths, obese and undernourished rates of the country to alert the user.



Personal Information

First Name:

Last Name:

Email Address:

Age:

Sex:

Country:

Food Category:

Allergies:

Preferences:

☒ Non-vegetarian ☐ Vegetarian

☐ Eggetarian ☐ Vegan

Diet

Food Energy Intake:
(in kcal)

Food Quantity:
(in kg)

Fat Quantity:

Protein Quantity:

Recommended Foods:

COVID deaths:

Obese rate:

Undemourished rate:

Save

Clear

Exit

Queries

The queries that I will be using for this database will help a user plan his/her diet depending on his/her food preferences, the USDA recommended macros, and the given macros of foods in each country (as macros vary from one country to another). This will let an individual follow a healthy, nutritious diet making our society a better ("COVID-free") place to live in.

1. Create new table

Since this is an open source data, there are inconsistencies among the tables. One such issue is redundant columns. The tables- FatQuantity, FoodEnergyIntake, FoodQuantity, ProteinQuantity- have repetitive columns- Obesity, Undernourished, Confirmed Deaths, Recovered, Active, and Population. In order to optimize, I created a new table, OU_COVID_rate containing the above mentioned columns. I deleted these particular columns in the rest of the tables.

```
SQLQuery26.sql - D...3\I8V5G\nehat (54)  SQLQuery
USE [COVID Healthy Diet];
Create table dbo.OU_COVID_rate
( Country nvarchar(225),
  Obesity float,
  Undernourished float,
  Confirmed float,
  Deaths float,
  Recovered float,
  Active float,
  Population float
);
```

2. Insert column values and delete columns from the other tables

I inserted the above particular columns in the table that I had created and deleted them in the rest of the tables.

```
Insert into dbo.OU_COVID_rate
Select Country, Obesity, Undernourished, Confirmed, Deaths, Recovered, Active, Population
From FatQuantity
order by Country;

alter table dbo.ProteinQuantity
drop column Obesity, Undernourished, Confirmed, Deaths, Recovered, Active, Population
```

3. Update Statement

I updated Afghanistan record in ProteinQuantity table by rounding off Animal fats and Cereals columns.

SQLQuery24.sql - D:\3118V5G\nehat (55))

```

update ProteinQuantity
set [Animal fats]= '0.028',
[Cereals - Excluding Beer]= '36'
where Country= 'Afghanistan'

```

SQLQuery26.sql - D:\3118V5G\nehat (52))

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [Country]
,[Alcoholic Beverages]
,[Animal Products]
,[Animal fats]
,[Aquatic Products, Other]
,[Cereals - Excluding Beer]
,[Eggs]
,[Fish, Seafood]
,[Fruits - Excluding Wine]
,[Meat]
,[Milk - Excluding Butter]
,[Offals]
,[Oilcrops]
,[Pulses]
,[Spices]

```

Messages

(1 row affected)

Completion time: 2021-07-22T23:20:52.4669738-0

Results

Country	Alcoholic Beverages	Animal Products	Animal fats	Aquatic Products, Other	Cereals - Excluding Beer
1 Afghanistan	0	9.7523	0.028	0	36
2 Albania	0.184	27.7469	0.0711	0	14.2331
3 Algeria	0.0323	13.836	0.0054	0	26.5633
4 Angola	0.6285	15.2311	0.0277	0	20.3882
5 Antigua and Barbuda	0.1535	33.1901	0.1289	0	10.5108
6 Argentina	0.1704	31.9799	0.0097	0	13.6702
7 Armenia	0.0411	22.8846	0.1438	0	18.2738
8 Australia	0.2914	32.9787	0.074	0.0046	10.9343
9 Austria	0.615	30.0009	0.3376	0	13.1508
10 Azerbaijan	0.3465	16.3039	0.0596	0	28.2218

TOP-3118V5G\nehat ... COVID Healthy Diet 00:00:00 0 rows

Query execu... DESKTOP-3118V5G\SQLEXPRESS ... DESKTOP-3118V5G\nehat ... COVID Healthy Diet 00:00:00 170 rows

4. Delete statements

I deleted Treenuts and Eggs records from FoodDescription table as Treenuts had only nuts and Eggs had Eggs as their items and providing a description for these categories seemed unnecessary.

SQLQuery27.sql - D:\3118V5G\nehat (52))

```

delete from FoodDescription
where Categories= 'Treenuts'

```

SQLQuery29.sql - D:\3118V5G\nehat (54))

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [Categories]
,[Items]
FROM [COVID Healthy Diet].[dbo].[FoodDescription]

```

Messages

(1 row affected)

Completion time: 2021-07-22T23:25:05.1413957-04:00

Results

Categories	Items
1 Alcoholic Beverages	Alcohol, Non-Food; Beer; Beverages, Alcoholic; Be...
2 Animal fats	Butter, Ghee; Cream; Fats, Animals, Raw; Fish, Bod...
3 Animal Products	Aquatic Animals, Others; Aquatic Plants; Bovine Me...
4 Aquatic Products, Other	Aquatic Animals, Others; Aquatic Plants; Meat, Aqu...
5 Cereals - Excluding Beer	Barley and products; Cereals, Other; Maize and prod...
6 Eggs	Eggs
7 Fish, Seafood	Cephalopods; Crustaceans; Demersal Fish; Freshwa...
8 Fruits - Excluding Wine	Apples and products; Bananas; Citrus, Other; Dates;...
9 Meat	Bovine Meat; Meat, Other; Mutton & Goat Meat; Pig...
10 Milk - Excluding Butter	Milk - Excluding Butter
11 Miscellaneous	Infant food; Miscellaneous

RESS ... DESKTOP-3118V5G\nehat ... COVID Healthy Diet 00:00:00 0 rows

C DESKTOP-3118V5G\SQLEXPRESS ... DESKTOP-3118V5G\nehat ... COVID Healthy Diet 00:00:00 22 rows

SQLQuery27.sql - D:\318V5G\nehat (52)*)

```
delete from FoodDescription
where Categories= 'Eggs'
```

99 %

Messages

(1 row affected)

Completion time: 2021-07-22T23:28:39.7125469-04:00

SQLQuery31.sql - D:\318V5G\nehat (56) *

```
/****** Script for SelectTopRows command from SSMS *****/
SELECT TOP (1000) [Categories]
,[Items]
FROM [COVID_Healthy_Diet].[dbo].[FoodDescription]
```

99 %

Results

Categories	Items
1	Alcoholic Beverages
2	Animal fats
3	Animal Products
4	Aquatic Products, Other
5	Cereals - Excluding Beer
6	Fish, Seafood
7	Fruits - Excluding Wine
8	Meat
9	Milk - Excluding Butter
10	Miscellaneous
11	Offals
12	Oilcrops
13	Pulses
14	Spices
15	Starchy Roots
16	Stimulants
17	Sugar & Sweeteners
18	Sugar Crops
19	Vegetable Oils
20	Vegetables
21	Vegetal Products

RESS ... | DESKTOP-318V5G\nehat ... | COVID Healthy Diet | 00:00:00 | 0 rows

DESKTOP-318V5G\SQLEXPRESS ... | DESKTOP-318V5G\nehat ... | COVID Healthy Diet | 00:00:00 | 21 rows

5. Join Statements

- To find out and display meat fat quantity and protein quantity by country, I used inner join to join both the tables (FatQuantity and ProteinQuantity) by country.

SQLQuery3.sql - DE\318V5G\nehat (58))

```
select FatQuantity.Country as 'Country',
FatQuantity.Meat as 'Meat Fat Quantity',
ProteinQuantity.Meat as 'Meat Protein Quantity'
from FatQuantity
Inner Join ProteinQuantity
on FatQuantity.Country= ProteinQuantity.Country
```

99 %

Results

	Country	Meat Fat Quantity	Meat Protein Quantity
1	Afghanistan	6.1244	3.1337
2	Albania	8.7428	7.6582
3	Algeria	3.8961	3.5088
4	Angola	11.0268	7.6248
5	Antigua and Barbuda	14.3202	16.067
6	Argentina	19.2693	21.625
7	Amenia	10.8165	8.5233
8	Australia	11.6002	18.8298
9	Austria	8.1099	13.6549
10	Azerbaijan	11.9993	6.8551
11	Bahamas	17.4941	20.4311
12	Bangladesh	1.8407	1.2773
13	Barbados	13.1382	13.9872
14	Belarus	11.5636	14.2402
15	Belgium	5.6817	10.0507
16	Belize	10.3435	10.748
17	Benin	3.2849	5.6948
18	Bolivia	21.1476	17.0794
19	Bosnia and Herzegovina	9.6957	6.7096

DESKTOP-318V5G\SQLEXPRESS ... | DESKTOP-318V5G\nehat ... | COVID Healthy Diet | 00:00:00 | 21 rows

- To find out and display food quantity, energy intake, fat quantity, and protein quantity of vegetables, I used join statement to join the four tables (FoodQuantity, FoodEnergyIntake, FatQuantity, and ProteinQuantity) by Country.

```

select FoodQuantity.Country,
FoodQuantity.Vegetables as 'Veg Food Quantity',
FoodEnergyIntake.Vegetables as 'Veg Energy Intake',
FatQuantity.Vegetables as 'Veg Fat Quantity',
ProteinQuantity.Vegetables as 'Veg Protein Quantity'
from FatQuantity
Join FoodEnergyIntake
On FatQuantity.Country = FoodEnergyIntake.Country
Join FoodQuantity
On FoodEnergyIntake.Country = FoodQuantity.Country
Join ProteinQuantity
On FoodEnergyIntake.Country = ProteinQuantity.Country

```

	Country	Veg Food Quantity	Veg Energy Intake	Veg Fat Quantity	Veg Protein Quantity
1	Afghanistan	6.7642	0.7504	0.3593	1.137
2	Albania	11.7753	2.7508	0.6503	3.2456
3	Algeria	11.6484	2.0457	0.5145	3.1267
4	Angola	2.3041	0.3525	0.1231	0.8133
5	Antigua and Barbuda	5.4495	1.296	0.2469	1.6024
6	Argentina	4.3503	0.8643	0.1878	1.0516
7	Amenia	16.7019	3.2164	0.8717	4.6621
8	Australia	5.1406	1.1326	0.2144	1.4524
9	Austria	5.1098	1.029	0.2039	1.5259
10	Azerbaijan	10.0755	1.4661	0.6534	1.9114
11	Bahamas	7.7649	1.5185	0.3974	2.6759
12	Bangladesh	3.6405	0.6165	0.3092	1.2026
13	Barbados	5.4725	1.0534	0.2534	1.4844
14	Belarus	8.7859	1.8389	0.3394	2.8587

Query executed successfully. DESKTOP-3II8V5G\SQLEXPRES

Ln 2 Col 37 Ch 37

- To find out Food Energy Intake and Fat Quantity of Cereals, I used inner join statement to join both the tables by Country. The last two lines can be used if the user needs to have less fat quantity, but at least 2.0000.

```

Select FoodEnergyIntake.Country,
FoodEnergyIntake.[Cereals - Excluding Beer] as 'Cereals Energy Intake',
FatQuantity.[Cereals - Excluding Beer] as 'Cereals Fat Quantity'
from FoodEnergyIntake
Inner Join FatQuantity
On FoodEnergyIntake.Country= FatQuantity.Country
where FatQuantity.[Cereals - Excluding Beer]> 2.0000
order by FatQuantity.[Cereals - Excluding Beer]

```

	Country	Cereals Energy Intake	Cereals Fat Quantity
1	Slovakia	10.6717	2.0003
2	Turkey	19.2938	2.0428
3	Saint Kitts and Nevis	13.3969	2.0602
4	Barbados	15.3514	2.1187
5	Serbia	20.0428	2.1213
6	Portugal	13.9072	2.1808
7	Jamaica	16.1255	2.2375
8	Lebanon	21.3268	2.253
9	Malaysia	21.5194	2.2933
10	Amenia	19.2658	2.5068
11	Trinidad and Tobago	18.4224	2.5134
12	Thailand	23.1871	2.538
13	Venezuela (Bolivarian Republic of)	21.3126	2.5449
14	Romania	17.849	2.5519

- To find out and display Food Quantity, Energy Intake, Fat Quantity, and Protein Quantity of Fish/Seafood and Confirmed COVID rate (%), I used join statement to join all the tables by Country and ordered them by confirmed COVID cases.

SQLQuery12.sql - D:\3118V5G\nehat (66)* X DESKTOP-3118V5G\SQ...Diet - dbo.View_1* SQLQuery10.sql - D:\3118V5G\neh

```

Select FatQuantity.Country,
OU_COVID_rate.Confirmed,
FoodQuantity.[Fish, Seafood] as 'Fish Food Quantity',
FoodEnergyIntake.[Fish, Seafood] as 'Fish Energy Intake',
FatQuantity.[Fish, Seafood] as 'Fish Fat Quantity',
ProteinQuantity.[Fish, Seafood] as 'Fish Protein Quantity'
From FatQuantity
Join FoodEnergyIntake
On FatQuantity.Country = FoodEnergyIntake.Country
Join FoodQuantity
On FoodEnergyIntake.Country = FoodQuantity.Country
Join OU_COVID_rate
On FoodEnergyIntake.Country = OU_COVID_rate.Country
Join ProteinQuantity
On FoodEnergyIntake.Country = ProteinQuantity.Country
order by OU_COVID_rate.Confirmed

```

99 %

Results Messages

	Country	Confirmed	Fish Food Quantity	Fish Energy Intake	Fish Fat Quantity	Fish Protein Quantity
1	French Polynesia	0	3.4809	1.6419	1.5842	7.0448
2	Kiribati	0	6.1065	2.6014	3.0833	16.1052
3	Korea, North	0	1.1568	0.4186	0.5375	2.9561
4	Myanmar	0	4.0424	2.0548	2.5441	8.7797
5	New Caledonia	0	2.0361	0.8544	0.8385	3.9207
6	Turkmenistan	0	0.2137	0.1045	0.1274	0.4685
7	Vanuatu	0.000311526	2.1649	1.2197	1.4343	6.8016
8	Lao People's Democratic Republic	0.000621375	1.4213	0.7877	1.6871	4.531
9	United Republic of Tanzania	0.000852111	0.5828	0.2922	0.4558	1.8567
10	Samoa	0.001	2.7788	1.7479	1.8968	7.3761

- To find out and display all the combinations of eggs in each country and fat quantity, I used cross join statements to join FoodEnergyIntake and FatQuantity tables.

SQLQuery10.sql - D:\3118V5G\nehat (51)* X

```

Select FoodEnergyIntake.Country,
FatQuantity.Eggs As 'Eggs Fat Quantity'
From FatQuantity
Cross Join FoodEnergyIntake

```

99 %

Results Messages

	Country	Eggs Fat Quantity
1	Afghanistan	0.6859
2	Albania	0.6859
3	Algeria	0.6859
4	Angola	0.6859
5	Antigua and Barbuda	0.6859
6	Argentina	0.6859
7	Amenia	0.6859
8	Australia	0.6859
9	Austria	0.6859
10	Azerbaijan	0.6859
11	Bahamas	0.6859
12	Bangladesh	0.6859
13	Barbados	0.6859
14	Belarus	0.6859
15	Belgium	0.6859
16	Belize	0.6859
17	Benin	0.6859
18	Bolivia	0.6859
19	Bosnia and Herzegovina	0.6859

6. View Statements

- To find out and display Food quantity, Energy Intake, Fat quantity, and Protein quantity of Milk, I used a view statement to join the four tables by country. This is an easier and a less time consuming way to display columns from multiple tables, instead of typing in the query code.

DESIGN: Desktop-3II8V5G\SQ...Diet - dbo.View_1

Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Or...	Or...	Or...
Country		FatQuantity	<input checked="" type="checkbox"/>						
[Milk - Exclud...		FatQuantity	<input checked="" type="checkbox"/>						
[Milk - Exclud...	Expr1	FoodEner...	<input checked="" type="checkbox"/>						
[Milk - Exclud...	Expr2	FoodQua...	<input checked="" type="checkbox"/>						

```

SELECT dbo.FatQuantity.Country, dbo.FatQuantity.[Milk - Excluding Butter], dbo.FoodEnergyIntake.[Milk - Excluding Butter] AS Expr1, dbo.FoodQuantity.[Milk - Excluding Butter] AS Expr2,
FROM dbo.FatQuantity INNER JOIN
dbo.FoodEnergyIntake ON dbo.FatQuantity.Country = dbo.FoodEnergyIntake.Country INNER JOIN
dbo.FoodQuantity ON dbo.FoodEnergyIntake.Country = dbo.FoodQuantity.Country INNER JOIN
dbo.ProteinQuantity ON dbo.FoodEnergyIntake.Country = dbo.ProteinQuantity.Country
  
```

SQLQuery5.sql - DE...3II8V5G\nehat (57)

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [Country]
, [Milk - Excluding Butter]
, [Expr1]
, [Expr2]
, [Expr3]
FROM [COVID_Healthy_Diet].[dbo].[Query_3]
  
```

99 %

	Country	Milk - Excluding Butter	Expr1	Expr2	Expr3
1	Afghanistan	8.2803	2.4512	7.5828	5.5278
2	Albania	17.7576	9.9441	15.7213	16.475
3	Algeria	8.0934	3.9869	7.6189	8.0616
4	Angola	1.2309	0.5067	0.8311	1.146
5	Antigua and Barbuda	6.6607	4.6904	6.3663	7.4349
6	Argentina	5.8512	3.1641	10.2328	5.8322
7	Armenia	10.4709	5.6368	9.9407	10.038
8	Australia	6.5196	4.8022	12.1018	8.432
9	Austria	5.1497	4.0482	12.3776	11.4954
10	Azerbaijan	9.9202	3.1738	8.3212	6.8497
11	Bahamas	5.1318	2.988	2.1757	4.8312
12	Bangladesh	3.2249	0.7706	1.8778	1.8578
13	Barbados	5.6176	3.1428	2.2041	5.3944
14	Belarus	2.0644	2.1884	6.2104	6.0202
15	Belgium	6.5057	4.6311	11.6344	11.9121
16	Belize	2.5053	1.8906	3.6317	4.5999
17	Benin	1.2469	0.5263	0.9458	1.094
18	Bolivia	3.2703	1.6355	4.2873	3.2143

Query executed successfully. DESKTOP-3II8V5G

- To find out and display Alcohol Energy Intake and Fat Quantity, I created a view with schemabinding and joined FoodEnergyIntake and FatQuantity tables.

SQLQuery20.sql - D:\3118V5G\nehat (57)*

```

Create view [dbo].[Alcohol_vw]
with schemabinding
as
Select dbo.FoodEnergyIntake.Country,
dbo.FoodEnergyIntake.[Alcoholic Beverages] as 'Alcohol Energy Intake',
dbo.FatQuantity.[Alcoholic Beverages] as 'Alcohol Fat Quantity'
From dbo.FoodEnergyIntake
Join dbo.FatQuantity
On dbo.FoodEnergyIntake.Country = dbo.FatQuantity.Country
Go

```

Messages
Commands completed successfully.
Completion time: 2021-07-22T23:10:28.3190319-04:00

SQLQuery23.sql - D:\3118V5G\nehat (60)*

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [Country]
,[Alcohol Energy Intake]
,[Alcohol Fat Quantity]
FROM [COVID Healthy Diet].[dbo].[Alcohol_vw]

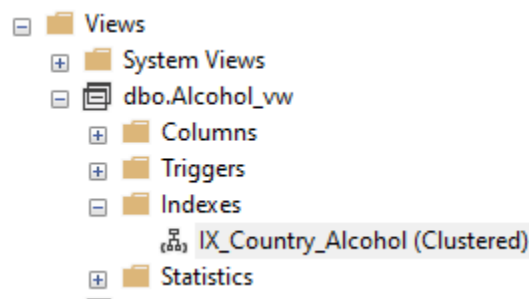
```

Results

	Country	Alcohol Energy Intake	Alcohol Fat Quantity
1	Afghanistan	0	0
2	Albania	0.912	0
3	Algeria	0.0896	0
4	Angola	1.9388	0
5	Antigua and Barbuda	2.3041	0
6	Argentina	1.4354	0
7	Ameria	0.2274	0
8	Australia	1.9783	0
9	Austria	2.8161	0
10	Azerbaijan	2.2555	0
11	Bahamas	2.9145	0

DESKTOP-3118V5G\SQLEXPRESS ... | DESKTOP-3118V5G\nehat ... | COVID Healthy Diet | 00:00:00 | 0 rows

SQLEXPRESS ... | DESKTOP-3118V5G\nehat ... | COVID Healthy Diet | 00:00:00 | 170 r



7. Aggregators

- To find out and display obesity rate by country in descending order, I used max aggregator and join statement to join FatQuantity and OU_COVID_rate tables.

SQLQuery15.sql - D...3I18V5G\nehat (74))* - X

```

select FatQuantity.Country,
max(OU_COVID_rate.Obesity) as 'Max Obesity'
from FatQuantity
Join OU_COVID_rate
On FatQuantity.Country=OU_COVID_rate.Country
Group by FatQuantity.Country
order by 'Max Obesity' desc

```

99 %

Results Messages

	Country	Max Obesity
1	Kiribati	45.6
2	Samoa	45.5
3	United States of America	37.3
4	Kuwait	37
5	Saudi Arabia	35
6	Jordan	33.4
7	Turkey	32.2
8	Bahamas	32.1
9	New Zealand	32
10	Lebanon	31.3
11	Canada	31.3

- To find out and display confirmed COVID cases % by country in descending order, I used min aggregator and join statement to join FatQuantity and OU_COVID_rate tables.

SQLQuery16.sql - D...3I18V5G\nehat (51))* - X

```

select FatQuantity.Country,
min(OU_COVID_rate.Confirmed) as 'Min Confirmed Cases'
from FatQuantity
Join OU_COVID_rate
On FatQuantity.Country=OU_COVID_rate.Country
Group by FatQuantity.Country
order by 'Min Confirmed Cases' desc

```

99 %

Results Messages

	Country	Min Confirmed Cases
1	Montenegro	10.40819936
2	Czechia	9.612840612
3	Slovenia	8.235901094
4	United States of America	8.15992185
5	Luxembourg	8.150632911
6	Panama	7.622320803
7	Israel	7.439051649
8	Portugal	7.429605071
9	Georgia	7.042261104
10	Lithuania	6.667072298
11	Belgium	6.286322188
12	Spain	6.176110003
13	Switzerland	6.160215427

- To find out the total population of the world, I used sum aggregator to add the populations of all the countries.

SQLQuery18.sql - D...3I18V5G\nehat (51))*

```

select sum (Population) as 'Total Population'
from OU_COVID_rate

```

99 %

Results Messages

	Total Population
1	7569019000

- To find out and display the number of confirmed COVID cases, I used multiplication to multiply confirmed % column to population column in OU_COVID_rate table and divided the values by 100.

SQLQuery18.sql - D...3I18V5G\nehat (51))*

```

select Country,
Confirmed*Population/100 as 'Confirmed COVID Cases'
from OU_COVID_rate
order by 'Confirmed COVID Cases' desc

```

99 %

Results Messages

	Country	Confirmed COVID Cases
1	United States of America	26917787.000343
2	India	10826362.99641
3	Brazil	9447164.99904552
4	United Kingdom	3941273.0001248
5	Russian Federation	3907652.99979462
6	France	3376266.0000928
7	Spain	2941989.99992905
8	Italy	2625097.99985288
9	Turkey	2524785.99987664
10	Germany	2285002.99972
11	Colombia	2151207.00019516
12	Argentina	1976688.99979899
13	Mexico	1926080.00038608
14	Poland	1545529.9998336
15	South Africa	1473699.99995712
16	Iran (Islamic Republic of)	1459369.999989
17	Ukraine	1285058.99981052
18	Peru	1173044.99989776
19	Indonesia	1147010.00106537
20	Czechia	1030111.99998192

SQLQuery19.sql - D...3I18V5G\nehat (58))*

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [Country]
,[Obesity]
,[Undernourished]
,[Confirmed]
,[Deaths]
,[Recovered]
,[Active]
,[Population]
FROM [COVID Healthy Diet].[dbo].[OU_COVID_rate]

```

99 %

Results Messages

	Country	Obesity	Undernourished	Confirmed	Deaths	Reco
1	Afghanistan	4.5	29.8	0.142134196	0.006185779	0.12
2	Albania	22.3	6.2	2.967300916	0.050951374	1.79
3	Algeria	26.6	3.9	0.244897085	0.006558153	0.16
4	Angola	6.8	25	0.061687473	0.00146055	0.05
5	Antigua and Barbuda	19.1	0	0.293877551	0.007142857	0.19
6	Argentina	28.5	4.6	4.356147387	0.108226635	3.90
7	Armenia	20.9	4.3	5.681224628	0.105345061	5.39
8	Australia	30.4	2.5	0.112025316	0.003529549	0.10
9	Austria	21.9	2.5	4.739982051	0.089679156	4.49
10	Azerbaijan	19.9	2.5	2.285536209	0.031222794	2.22

RESS ... DESKTOP-3I18V5G\nehat ... COVID Healthy Diet 00:00:00 170 rows

DESKTOP-3I18V5G\SQLEXPRESS ... DESKTOP-3I18V5G\nehat ... COVID Healthy Diet 0

- To find out and display the number of death due to COVID, I used multiplication to multiply death % column to population column in OU_COVID_rate table and divided the values by 100.

SQLQuery18.sql - D:\3118V5G\nehat (51))

```

select Country,
Deaths*Population/100 as 'COVID Deaths'
from OU_COVID_rate
order by 'COVID Deaths' desc

```

Results

	Country	COVID Deaths
1	United States of America	462169.00073024
2	Brazil	230033.99996544
3	Mexico	165786.00053792
4	India	154995.998352
5	United Kingdom	112304.9996836
6	Italy	91002.99999528
7	France	78940.0002828
8	Russian Federation	75010.00057446
9	Germany	61550.999748
10	Spain	61385.99980415
11	Iran (Islamic Republic of)	58411.9996515
12	Colombia	55693.00021204
13	Argentina	49110.00016395
14	South Africa	46179.99996636
15	Peru	41933.00005664
16	Poland	38994.00001024
17	Indonesia	31393.0012357
18	Turkey	26684.99984521
19	Ukraine	24734.99995956
20	Belgium	21352.00000045

SQLQuery19.sql - D:\3118V5G\nehat (58))

```

/***** Script for SelectTopNRows command from SSMS *****/
SELECT TOP (1000) [Country]
,[Obesity]
,[Undernourished]
,[Confirmed]
,[Deaths]
,[Recovered]
,[Active]
,[Population]
FROM [COVID Healthy Diet].[dbo].[OU_COVID_rate]

```

Results

	Country	Obesity	Undernourished	Confirmed	Deaths	Recovered
1	Afghanistan	4.5	29.8	0.142134196	0.006185779	0.1233
2	Albania	22.3	6.2	2.967300916	0.050951374	1.7926
3	Algeria	26.6	3.9	0.244897085	0.006558153	0.1675
4	Angola	6.8	25	0.061687473	0.00146055	0.0568
5	Antigua and Barbuda	19.1	0	0.293877551	0.007142857	0.1908
6	Argentina	28.5	4.6	4.356147387	0.108226635	3.9051
7	Armenia	20.9	4.3	5.681224628	0.105345061	5.3984
8	Australia	30.4	2.5	0.112025316	0.003529549	0.1012
9	Austria	21.9	2.5	4.739982051	0.089679156	4.4968
10	Azerbaijan	19.9	2.5	2.285536209	0.031222794	2.2255

8. Stored Procedures (SPROCs)

- I also used stored procedure statements in order to save and reuse the code. I use them to get other values from different tables by inputting the values that I want from the records. In order to get only vegan meals in one particular country, I used stored procedure to filter out only the details of vegan categories by inputting the name of the Country.

SQLQuery13.sql - D:\3118V5G\nehat (59))

```

Create proc VeganFatProtein @Country nvarchar(225)
as
Begin
Select FatQuantity.[Country as 'Country',
FatQuantity.[Cereals - Excluding Beer] as 'Cereals Fat', FatQuantity.[Fruits - Excluding Wine] 'Fruits Fat',
FatQuantity.[Alcoholic Beverages] as 'Alcohol Fat', FatQuantity.Pulses as 'Pulses Fat', FatQuantity.Spices as 'Spices Fat',
FatQuantity.[Starchy Roots] as 'Starchy Roots Fat', FatQuantity.Oilcrops as 'Oilcrops Fat',
FatQuantity.Stimulants as 'Stimulants Fat', FatQuantity.[Sugar Crops] as 'Sugar Crops Fat',
FatQuantity.[Sugar & Sweeteners] as 'Sugar & Sweeteners Fat', FatQuantity.Treenuts as 'Treenuts Fat',
FatQuantity.[Vegetal Products] as 'Vegetal Products Fat', FatQuantity.[Vegetable Oils] as 'Vegetable Oil Fat',
FatQuantity.Vegetables as 'Vegetables Fats', ProteinQuantity.[Alcoholic Beverages] as 'Alcohol Protein',
ProteinQuantity.[Cereals - Excluding Beer] as 'Cereals Protein', ProteinQuantity.[Fruits - Excluding Wine] as 'Fruits Protein',
ProteinQuantity.Oilcrops as 'Oil Crops Protein', ProteinQuantity.Pulses as 'Pulses Protein', ProteinQuantity.Spices as 'Spices Protein',
ProteinQuantity.[Starchy Roots] as 'Starchy Roots Protein', ProteinQuantity.Stimulants as 'Stimulants Protein',
ProteinQuantity.[Sugar Crops] as 'Sugar Crops Protein', ProteinQuantity.[Sugar & Sweeteners] as 'Sugar & Sweeteners Protein',
ProteinQuantity.Treenuts as 'Treenuts Protein', ProteinQuantity.[Vegetable Oils] as 'Vegetable Oils Protein',
ProteinQuantity.[Vegetal Products] as 'Vegetal Products', ProteinQuantity.Vegetables as 'Vegetables Protein'
From FatQuantity
Join ProteinQuantity
On FatQuantity.Country= ProteinQuantity.Country
where FatQuantity.Country= @Country
end

```

Messages

Commands completed successfully.

Completion time: 2021-08-01T19:20:05.0030252-04:00

Query executed successfully.

Execute Procedure - [dbo].[VeganFatProtein]

Select a page

General

Parameter	Data Type	Output Param...	Pass Null Value	Value
@Country	nvarchar(225)	No	<input type="checkbox"/>	

Ex- I filtered out Country to 'United States of America' in order to get its vegan diet details.

The screenshot shows a SQL query window with the following code:

```
USE [COVID Healthy Diet]
GO

DECLARE @return_value int

EXEC @return_value = [dbo].[VeganFatProtein]
    @Country = N'United States of America'

SELECT 'Return Value' = @return_value
GO
```

The Results pane shows a single row for the 'United States of America' with the following values:

Country	Cereals Fat	Fruits Fat	Alcohol Fat	Pulses Fat	Spices Fat	Starchy Roots Fat	Oilcrops Fat	Stimulants Fat	Sugar Crops Fat	Sugar & Sweeteners Fat	Treenuts Fat
United States of America	1.2829	0.4426	0	0.0449	0.0867	0.0478	1.7464	0.3977	0	0	0.942

The Messages pane shows: "Return Value: 0". The status bar indicates "Query executed successfully." and "2 rows".

- To find out and display Animal Fats details, I used SPROC and joined the four tables by Country.

The screenshot shows the creation of a stored procedure named 'AnimalFats' with the following code:

```
Create proc AnimalFats @Country nvarchar(225)
as
Begin
Select FatQuantity. Country as 'Country', FatQuantity.[Animal fats] 'Animal fats Fat',
FoodEnergyIntake.[Animal fats] as 'Animal fats Energy Intake',
FoodQuantity.[Animal fats] as 'Animal fats Quantity', ProteinQuantity.[Animal fats] as 'Animal fats Protein'
From FatQuantity
Inner Join FoodEnergyIntake
On FatQuantity.Country = dbo.FoodEnergyIntake.Country
Inner Join FoodQuantity
On FoodEnergyIntake.Country = dbo.FoodQuantity.Country
Inner Join ProteinQuantity
On FoodEnergyIntake.Country = dbo.ProteinQuantity.Country
where FatQuantity.Country= @Country
end
```

The Messages pane shows: "Commands completed successfully." and "Completion time: 2021-08-01T19:33:40.7487174-04:00".

An "Execute Procedure - [dbo].[AnimalFats]" dialog box is open, showing the parameter '@Country' with a data type of 'nvarchar(225)' and a value of 'India'.

Ex- I filtered out Country to 'India' to find out its Animal Fats details.

```

USE [COVID Healthy Diet]
GO

DECLARE @return_value int

EXEC @return_value = [dbo].[AnimalFats]
    @Country = N'India'

SELECT 'Return Value' = @return_value
GO

```

99 %

Results Messages

	Country	Animal fats Fat	Animal fats Energy Intake	Animal fats Quantity	Animal fats Protein
1	India	7.4859	1.4899	0.0028	0.0306

Return Value

1	0
---	---

Query executed successfully. DESKTOP-3

- To find out and display Alcoholic Beverages details, I used SPROC and joined the four tables by Country.

```

Create proc Alcohol @AlcoholProtein float
as
Begin
Select ProteinQuantity.Country as 'Country',
FatQuantity.[Alcoholic Beverages] as 'Alcohol Fat',
FoodEnergyIntake.[Alcoholic Beverages] as 'Alcohol Energy Intake',
FoodQuantity.[Alcoholic Beverages] as 'Alcohol Food Quantity'
From FatQuantity
Join FoodEnergyIntake
On dbo.FatQuantity.Country = FoodEnergyIntake.Country
Join FoodQuantity
On FoodEnergyIntake.Country = FoodQuantity.Country
Join ProteinQuantity
On FoodEnergyIntake.Country = ProteinQuantity.Country
where ProteinQuantity.[Alcoholic Beverages]= @AlcoholProtein
end

```

99 %

Messages

Commands completed successfully.

Completion time: 2021-08-01T19:54:01.6003192-04:00

Execute Procedure - [dbo].[Alcohol]

Parameter Data Type Output Param... Pass Null Value Value

@AlcoholProtein	float	No	<input type="checkbox"/>	
-----------------	-------	----	--------------------------	--

Ex- To find out the details of countries and alcohol, I filtered out Alcohol Protein with the value of '0' only.

```

USE [COVID Healthy Diet]
GO

DECLARE @return_value int

EXEC     @return_value = [dbo].[Alcohol]
        @AlcoholProtein = 0

SELECT   'Return Value' = @return_value
GO

```

Country	Alcohol Fat	Alcohol Energy Intake	Alcohol Food Quantity
1 Afghanistan	0	0	0.0014
2 Bangladesh	0	0	0.0031
3 Ecuador	0	0.9275	3.2929
4 Haiti	0	2.0559	1.2733
5 Iran (Islamic Republic of)	0	0	0.0007
6 Kuwait	0	0	0.0044
7 Mauritania	0	0	0.0055
8 Pakistan	0	0	0.0085

Return Value
0

Query executed successfully.

- To find out and display country and meat details with COVID confirmed and death cases, I used SPROC and joined the five tables by Obesity and Undernourished rates of the countries.

```

Create proc Meat @Undernourished float, @Obesity float
as
Begin
Select FatQuantity.Country as 'Country',
FatQuantity.Meat as 'Meat Fats', FoodEnergyIntake.Meat as 'Meat Energy Intake',
FoodQuantity.Meat as 'Meat Food Quantity', ProteinQuantity.Meat as 'Meat Protein',
OU_COVID_rate.Confirmed as 'Confirmed cases', OU_COVID_rate.Deaths as 'Deaths'
From FatQuantity
Join FoodEnergyIntake
On FatQuantity.Country = FoodEnergyIntake.Country
Join FoodQuantity
On FoodEnergyIntake.Country = FoodQuantity.Country
Join OU_COVID_rate
On FoodEnergyIntake.Country = OU_COVID_rate.Country
Join ProteinQuantity
On FoodEnergyIntake.Country = ProteinQuantity.Country
where Undernourished=@Undernourished and Obesity=@Obesity
end

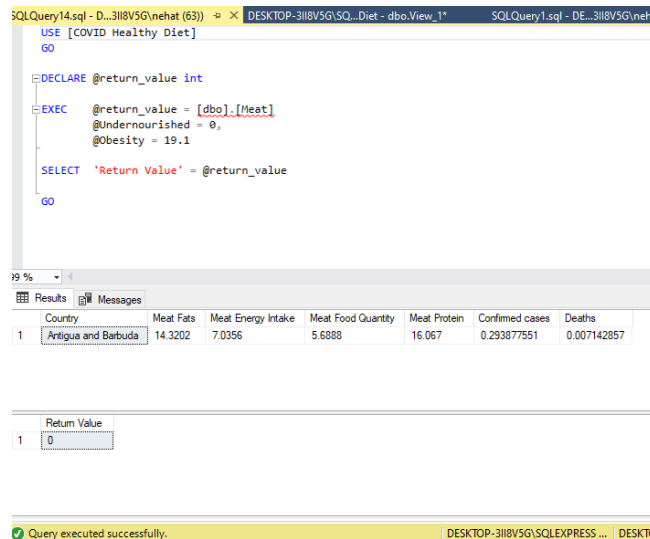
```

Commands completed successfully.

Completion time: 2021-08-01T20:16:47.7363853-04:00

Parameter	Data Type	Output Parame...	Pass Null Value	Value
@Undernourished	float	No	<input type="checkbox"/>	
@Obesity	float	No	<input type="checkbox"/>	

Ex- In order to find out meat details and COVID cases of a country, I filtered out obesity rate with '19.1' and undernourished rate with '0'.



```
USE [COVID Healthy Diet]
GO

DECLARE @return_value int

EXEC @return_value = [dbo].[Meat]
    @Undernourished = 0,
    @Obesity = 19.1

SELECT 'Return Value' = @return_value
GO
```

	Country	Meat Fats	Meat Energy Intake	Meat Food Quantity	Meat Protein	Confirmed cases	Deaths
1	Antigua and Barbuda	14.3202	7.0356	5.6888	16.067	0.293877551	0.007142857

	Return Value
1	0

Query executed successfully.

Conclusion

Humanity has faced deadly pandemics in the past such as Spanish flu, Bubonic plague, etc., where the death toll has reached 100s of millions. The world has gone through these and yet is still standing. My project serves more as an alert for people to be more careful of their health and take necessary actions when the time is dire and not as a mere indication of the end of the world.

Work Cited

Ren, Maria. Covid-19 Healthy Diet Dataset. Kaggle. Feb 07, 2021.