



Sentiment Analysis

Manimanya Reddy (NUID: 002763582)

About:

In today's world economy, people are leading a successful life in all parts of the world. Amidst covid wrecking out normal lives, A lot of families have lost members, countries' economies have hit bottom, people are still bouncing back to reach normality. But the real question arises, how many people are actually happy with the lives they are leading?

This project mainly revolves around different categories that could influence a person's success and happiness by analyzing data of internet usage, world happiness report, world's crime report, world's literacy report, world's gdp, covid statistics, universities in a country, olympics success rate and not the least world's alcohol consumption.

By analyzing, scrapping, munging, cleaning and normalizing huge amounts of data, this database provides a person a clear understanding of which country they could settle with respect to all satisfying permutations and combinations of categories mentioned above.

- Ultimately, by refining the data, a person can now visualize data by looking at graphs that could affect a person's opinion to expand business, or to even settle in!

Assumptions-

If a country has the highest GDP does that mean all people are successful and happy. But is that really true? Let's find out through this database

Data Sources:

- Kaggle
- Data repositories
- Web_scraping

Tables:

- Country
- Happiness Report
- Crime Report
- Gdp Report
- Literacy Report
- Internet Report
- Alchappines
- Covid status
- Olympic medals
- Universities
- Covidstatus_log (view)
- Country_medals_log (view)

SQL Create Statements-

```
create table crime(
    crime_id INT primary key auto_increment,
    country_code CHAR(5),
    crime_index DECIMAL(10,6),
    unemployment DECIMAL(10,6),
    foreign key (country_code) references country(country_code)
);
```

```
create table alchappiness(
    alchappiness_id INT primary key auto_increment,
    country_code CHAR(5),
    human_development DECIMAL(10,6),
    beer_capita DECIMAL(10,6),
```

```
wine_capita DECIMAL(10,6),
foreign key (country_code) references country(country_code)
);
```

```
create table gdp(
gdp_id INT primary key auto_increment,
country_code CHAR(5),
year_2020 BIGINT(50),
year_2021 BIGINT(50),
foreign key (country_code) references country(country_code)
);
```

```
create table happiness(
happiness_id INT primary key auto_increment,
country_code CHAR(5),
freedom DECIMAL(10,6),
happiness_score DECIMAL(10,6),
foreign key (country_code) references country(country_code)
);
```

```
create table internet(
internet_id INT primary key auto_increment,
country_code CHAR(5),
income_person DECIMAL(20,10),
internet_rate DECIMAL(20,10),
foreign key (country_code) references country(country_code)
);
```

```
create table literacy(
literacy_id INT primary key auto_increment,
country_code CHAR(5),
data_year INT,
literacy_rate DECIMAL(10,6),
population BIGINT(55),
foreign key (country_code) references country(country_code)
);
```

```
create table country(
country_code CHAR(5) primary key,
country_name VARCHAR(25));
```

```
create table covidstatus(
country_code CHAR(5) primary key,
total_cases INT,
total_deaths INT,
total_recovered INT,
active_cases INT,
death_permillion INT,
foreign key(country_code) references country(country_code)
);
```

```
Create table medals(
country_code CHAR(5) primary key,
Year INT
Gold INT,
Silver INT,
Bronze INT,
foreign key(country_code) references country(country_code)
);
```

```
Create table universities(
University_id INT primary key auto_increment
Country_code CHAR(5),
Uni_name VARCHAR(100)
Uni_url VARCHAR(100)
foreign key(country_code) references country(country_code)
);
```

model.mwb - MySQL Workbench

Administration Schemas active_covid - Routine active_cases - Routine active_cases - Routine active_cases - Routine querieswithoutviews country_medals_log > Context Help Snippets

SCHEMAS

Filter objects

```

13  #2. Which are the countries with top 3 internet usage?
14  • SELECT t2.country_name, t1.country_code, t1.internet_rate
15  FROM internet t1
16  INNER JOIN country t2 ON t2.country_code = t1.country_code
17  ORDER BY t1.internet_rate DESC
18  LIMIT 3;
19
20  #3. Average internet usage of all countries with 200 > gallons of beer capita
21  • select avg(t2.internet_rate) as "Average Internet Usage"
22  from alchappiness t1
23  join internet t2 on t1.country_code=t2.country_code
24  where t1.beer_capita>200;
25
26  #4. Average crime rate of highest 3 gdp countries ranking and 3
27  • (select avg(crime_index) from
28  (select cr.crime_index
29  from country c
30  join gdp g on c.country_code=g.country_code
31  join crime cr on cr.country_code=c.country_code
32  ORDER BY year 2020 DESC
33  LIMIT 3 ) AS t1)
34  UNION ALL
35  (select avg(crime_index) from

```

Object Info Session

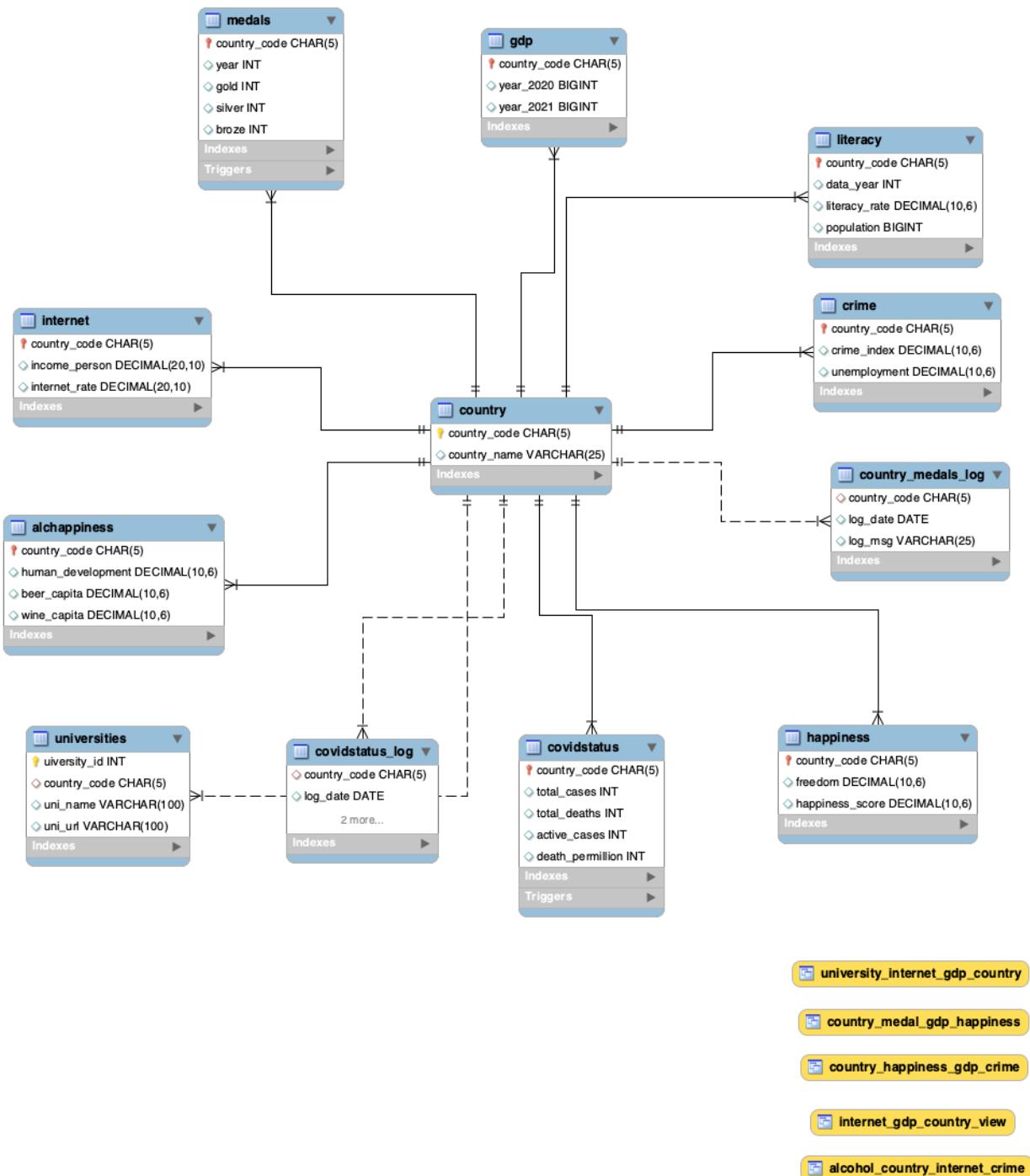
Schema: assignment_3

Action Output

Time	Action	Response	Duration / Fetch Time
139 18:27:08	SELECT * FROM assignment_3.covidstatus_log LIMIT 0,1...	3 row(s) returned	0.0011 sec / 0.00001...
140 18:27:16	update covidstatus set total_cases=total_cases+1 where...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.0071 sec
141 18:27:25	update covidstatus set total_cases=total_cases+1 where...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.0038 sec
142 18:27:34	SELECT * FROM assignment_3.covidstatus_log LIMIT 0,1...	5 row(s) returned	0.00054 sec / 0.000...

Query Completed

ER Diagram:



Script to scrape and insert covid_status for all countries.

```
In [22]: import requests
page = requests.get("https://www.worldometers.info/coronavirus")

In [23]: from bs4 import BeautifulSoup
soup = BeautifulSoup(page.content, 'lxml')
# print(soup.prettify())

In [24]: table = soup.find('table', attrs={'id': 'main_table_countries_today'})
rows = table.find_all("tr", attrs={"style": ""})

In [26]: data = []
for i,item in enumerate(rows):
    if i == 0:
        data.append(item.text.strip().split("\n")[:13])
    else:
        data.append(item.text.strip().split("\n")[:12])

In [28]: import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)

import pandas as pd
import dask.dataframe as dd

dt = pd.DataFrame(data)
dt = pd.DataFrame(data[1:], columns=data[0][:12]) #Formatting the header
df = dd.from_pandas(dt,npartitions=1)

In [29]: dt.to_csv('./covidnew.csv')
```

1.insert into country(country_name, country_code)

VALUES

('belgium-luxembrog', 'blx');

2.insert into country(country_name, country_code)

VALUES

('br. antr. terr' , 'bat');

3.insert into country(country_name, country_code)

VALUES

('european union', 'eu');

4. insert into country(country_name, country_code)

VALUES

('free zones', 'fre');

5. insert into country(country_name, country_code)

VALUES

('jhonston island', 'jtn');

Python Scripts to import tables-

```
import mysql.connector as msq
from mysql.connector import Error
try:
    conn = msq.connect(host='localhost', user='root',
                        password='Naruto@99', database='assignment_3')
    if conn.is_connected():
        cursor = conn.cursor()
except Error as e:
    print("Error while connecting to MySQL", e)

import pandas as pd
gdp = pd.read_csv("/Users/manyareddy/Desktop/all/csv/gdpfinal.csv", header=None,
skiprows=1)
for i, row in gdp.iterrows():
    if (len(row[0]))<4:
        sql = "INSERT IGNORE INTO assignment_3.gdp VALUES (%s,%s,%s)"
        cursor.execute(sql, tuple(row))
        conn.commit()

import pandas as pd
alc = pd.read_csv("/Users/manyareddy/Desktop/all/csv/alchappinessfinal.csv", header=None,
skiprows=1)
# alc
for i, row in alc.iterrows():
    if (len(row[0]))<5:
        sql = "INSERT IGNORE INTO assignment_3.alchappiness VALUES (%s,%s,%s,%s)"
        cursor.execute(sql, tuple(row))
        conn.commit()

import pandas as pd
crime = pd.read_csv("/Users/manyareddy/Desktop/all/csv/crimefinal.csv", header=None,
skiprows=1)
for i, row in crime.iterrows():
    if (len(row[0]))<5:
        sql = "INSERT IGNORE INTO assignment_3.crime VALUES (%s,%s,%s)"
        cursor.execute(sql, tuple(row))
        conn.commit()
```

```

import pandas as pd
covidstatus = pd.read_csv("/Users/manyareddy/Desktop/all/csv/covidstatus.csv", header=None,
skiprows=1)
for i,row in covidstatus.iterrows():
    if (len(row[0]))<5:
        sql = "INSERT IGNORE INTO assignment_3.covidstatus VALUES (%s,%s,%s,%s,%s)"
        cursor.execute(sql, tuple(row))
        conn.commit()

import pandas as pd
literacy = pd.read_csv("/Users/manyareddy/Desktop/all/csv/literacyfinal.csv", header=None,
skiprows=1)
for i,row in literacy.iterrows():
    if (len(row[0]))<5:
        sql = "INSERT IGNORE INTO assignment_3.literacy VALUES (%s,%s,%s,%s)"
        cursor.execute(sql, tuple(row))
        conn.commit()

import pandas as pd
internet = pd.read_csv("/Users/manyareddy/Desktop/all/csv/internetfinal.csv", header=None,
skiprows=1)
for i,row in internet.iterrows():
    if (len(row[0]))<5:
        sql = "INSERT IGNORE INTO assignment_3.internet VALUES (%s,%s,%s)"
        cursor.execute(sql, tuple(row))
        conn.commit()

import pandas as pd
co = pd.read_csv("/Users/manyareddy/Desktop/all/csv/countryfinal.csv", header=None,
skiprows=1)
for i,row in co.iterrows():
    if (len(row[0]))<5:
        sql = "INSERT IGNORE INTO assignment_3.country VALUES (%s,%s)"
        cursor.execute(sql, tuple(row))
        conn.commit()

import pandas as pd
uni = pd.read_csv("/Users/manyareddy/Desktop/all/csv/unisfinal.csv", header=None,
skiprows=1)

```

```
for i,row in uni.iterrows():
    sql = "INSERT IGNORE INTO assignment_3.universities VALUES (%s,%s,%s,%s)"
    cursor.execute(sql, tuple(row))
    conn.commit()

import pandas as pd
medals = pd.read_csv("/Users/manyareddy/Desktop/all/csv/medalsfinal.csv", header=None,
skiprows=1)
medals
for i,row in medals.iterrows():
    sql = "INSERT IGNORE INTO assignment_3.medals VALUES (%s,%s,%s,%s,%s)"
    cursor.execute(sql, tuple(row))
    conn.commit()

import pandas as pd
happiness = pd.read_csv("/Users/manyareddy/Desktop/all/csv/happinessfinal.csv",
header=None, skiprows=1)

for i,row in happiness.iterrows():
    sql = "INSERT IGNORE INTO assignment_3.happiness VALUES (%s,%s,%s)"
    cursor.execute(sql, tuple(row))
    conn.commit()
```

Use cases-

1. How is the happiness index in the top 3 countries of Highest GDP?

```
SELECT  
t1.happiness_score, t2.country_name, t3.year_2021  
FROM happiness t1  
INNER JOIN  
country t2 ON t1.country_code = t2.country_code  
INNER JOIN  
gdp t3 ON t3.country_code = t1.country_code  
order by t3.year_2021 desc  
limit 3;
```

2. Which are the countries with highest internet usage? (limit by top3 countries)

```
SELECT t2.country_name, t1.country_code, t1.internet_rate  
FROM internet t1  
INNER JOIN country t2 ON t2.country_code = t1.country_code  
order by t1.internet_rate desc  
limit 3;
```

3. What is the Average internet usage of all countries with 200 > gallons of beer capita?

```
select avg(t2.internet_rate) as "Average Internet Usage"  
from alchappiness t1  
join internet t2 on t1.country_code=t2.country_code  
where t1.beer_capita>200;
```

4. What is the Average crime rate of the highest and lowest 3 GDP countries' ranking?

```
(select avg(cr.crime_index) from  
(select cr.crime_index  
from country c  
join gdp g on c.country_code=g.country_code  
join crime cr on cr.country_code=c.country_code  
order by year_2020 desc  
limit 3 ) as t1)  
union all  
(select avg(cr.crime_index) from  
(select cr.crime_index  
from country c
```

```
join gdp g on c.country_code=g.country_code
join crime cr on cr.country_code=c.country_code
order by year_2020 asc
limit 3) as t2);
```

5. Which countries have the highest crime rate and what is their happiness index?

```
SELECT
t1.happiness_score, t2.country_name, t3.crime_index
FROM happiness t1
INNER JOIN
country t2 ON t1.country_code = t2.country_code
INNER JOIN
crime t3 ON t3.country_code = t1.country_code
order by t3.crime_index desc
limit 3;
```

6. What are the top 150 universities located in the highest GDP countries, limited by 3 countries?

```
select u.uni_name, c.country_name, g.year_2021
from gdp g
join country c on c.country_code=g.country_code
join universities u on c.country_code=u.country_code
order by year_2021 desc
limit 150;
```

7. Top 3 countries have the most gold medals in 1996 and hows the happiness index in those countries?

```
select c.country_name, m.year, m.gold
from medals m
join country c on c.country_code=m.country_code
where year=1996
order by gold desc
limit 3;
```

8. Top 3 countries have the highest crime rate with alcohol beer_capita consumption>200 gallons?

```
select com.country_name, c.crime_index, a.beer_capita
from crime c
join alchappiness a on a.country_code=c.country_code
join country com on com.country_code=c.country_code
where a.beer_capita>200
order by crime_index Desc
limit 3;
```

9. What countries have won gold medals since 1995 and how's their GDP in 2021?

```
select c.country_name, m.gold, g.year_2020 as "GDP 2020"
from medals m
join gdp g on g.country_code=m.country_code
join country c on c.country_code=m.country_code
where gold>0 AND year>1995;Internet usage of the highest GDP countries?
select c.country_name, g.year_2021 as gdp_2021, i.internet_rate
from gdp g
join internet i on i.country_code=g.country_code
join country c on c.country_code=g.country_code
order by year_2021
limit 3;
```

10. Internet usage of the highest GDP countries?

```
select c.country_name, g.year_2021 as gdp_2021, i.internet_rate
from gdp g
join internet i on i.country_code=g.country_code
join country c on c.country_code=g.country_code
order by year_2021
limit 3;
```

model.mwb - MySQL Workbench

Administration Schemas query_with_views triggers* country_medals_log index* SQL File 16* crime universities new_procedure - Routine > Context Help Snippets

Schemas

Filter objects

7
8 #2. Which are the countries with top 3 internet usage?
9 SELECT country_name, country_code, internet_rate
10 from alcohol_country_internet_crime
11 order by internet_rate desc
12 limit 3;
13
14 #3. Average internet usage of all countries with 200 > gallons of beer capita
15 select avg(internet_rate) as "Average Internet Usage"
16 from alcohol_country_internet_crime
17 where beer_capita>200;
18

Result Grid Filter Rows: Search Export: Fetch rows: Result Grid Form Editor

country_name	country_code	internet_rate
Iceland	ISL	95.6381132100
Norway	NOR	93.2775079300
Netherlands	NLD	90.703550900

alcohol_country_internet_crime 30

Action Output

Time	Action	Response	Duration / Fetch Time
151	19:16:14 select country_name, crime_index, beer_capita from alcohol_country_internet_crime	3 row(s) returned	0.0083 sec / 0.00002...
152	19:16:36 select country_name, year_2021 as gdp_2021, internet_rate from alcohol_country_internet_crime	3 row(s) returned	0.0055 sec / 0.00001...
153	19:28:43 select avg(internet_rate) as "Average Internet Usage" from alcohol_country_internet_crime	1 row(s) returned	0.014 sec / 0.000020...
154	19:28:47 SELECT country_name, country_code, internet_rate from alcohol_country_internet_crime	3 row(s) returned	0.0041 sec / 0.00001...

Query Completed

model.mwb - MySQL Workbench

Administration Schemas query_with_views triggers* country_medals_log index* SQL File 16* crime universities new_procedure - Routine > Context Help Snippets

Schemas

Filter objects

1 #1. How is the happiness index in the top 3 countries of Highest GDP?
2
3 SELECT happiness_score, country_name, year_2021
4 FROM country_happiness_gdp_crime
5 order by year_2021 desc
6 limit 3;
7
8 #2. Which are the countries with top 3 internet usage?
9 SELECT country_name, country_code, internet_rate
10 from alcohol_country_internet_crime
11 order by internet_rate desc
12 limit 3;
13
14 #3. Average internet usage of all countries with 200 > gallons of beer capita
15 select avg(internet_rate) as "Average Internet Usage"
16 from alcohol_country_internet_crime

Result Grid Filter Rows: Search Export: Fetch rows: Result Grid Form Editor

country_name	gdp_2021	internet_rate
Tuvalu	63100982	26.0000000000
Dominica	545616519	47.2804360300
Samoa	788389972	6.9650380430

internet_gdp_country_view 18

Action Output

Time	Action	Response	Duration / Fetch Time
139	18:27:08 SELECT * FROM assignment_3.covidstatus_log LIMIT 0,...	3 row(s) returned	0.0011 sec / 0.000011...
140	18:27:16 update covidstatus set total_cases=total_cases+1 where...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.0071 sec
141	18:27:25 update covidstatus set total_cases=total_cases+1 where...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.0038 sec
142	18:27:34 SELECT * FROM assignment_3.covidstatus_log LIMIT 0,...	5 row(s) returned	0.00054 sec / 0.000...

Query Completed

Views-

1. internet_gdp_country_view
2. country_happiness_gdp_crime
3. country_medal_gdp_happiness
4. alcohol_country_internet_crime
5. University_internet_gdp_country

Creating views-

```
create view country_happiness_gdp_crime as
select c.country_code, c.country_name, h.freedom, h.happiness_score, g.year_2020,
g.year_2021, cr.crime_index, cr.unemployment
from country c
join happiness h on c.country_code=h.country_code
join gdp g on c.country_code=g.country_code
join crime cr on c.country_code=cr.country_code;
```

```
create view alcohol_country_internet_crime as
select c.country_code, c.country_name, a.beer_capita, a.wine_capita, a.human_development,
i.income_person, i.internet_rate, cr.unemployment, cr.crime_index
from country c
join alchappiness a on c.country_code=a.country_code
join internet i on c.country_code=i.country_code
join crime cr on c.country_code=cr.country_code;
```

```
create view country_medal_gdp_happiness as
select c.country_code, c.country_name, m.year, m.gold, m.silver, m.bronze, g.year_2020,
g.year_2021, h.freedom, h.happiness_score
from country c
join medals m on c.country_code=m.country_code
join gdp g on c.country_code=g.country_code
join happiness h on c.country_code=h.country_code;
```

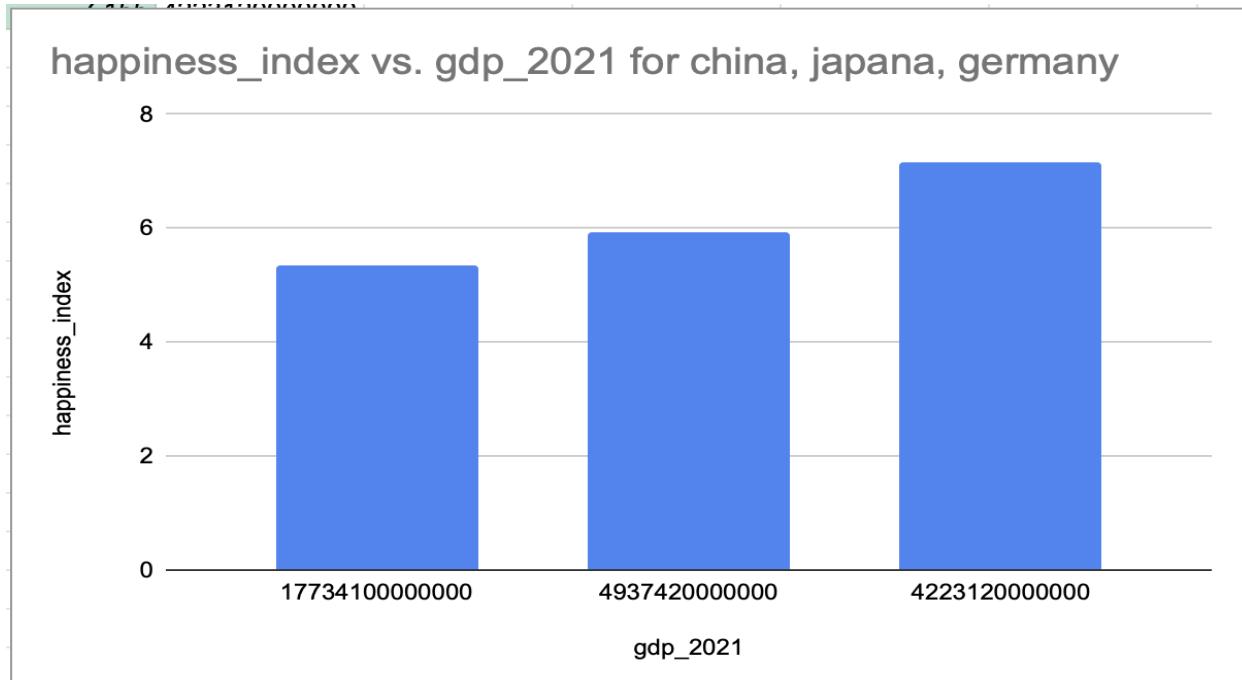
```
Create view university_internet_gdp_country as
select c.country_code, c.country_name, i.income_person, i.internet_rate, g.year_2020,
g.year_2021, u.uni_name, u.uni_url
from country c
join universities u on c.country_code=u.country_code
join internet i on c.country_code=i.country_code
```

```
join gdp g on c.country_code=g.country_code
```

Queries with views and visualization-

1. How is the happiness index in the top 3 countries of Highest GDP?

```
SELECT happiness_score, country_name, year_2021  
FROM country_happiness_gdp_crime  
order by year_2021 desc  
limit 3;
```



2. Which are the countries with highest internet usage? (limit by top3 countries)

```
SELECT country_name, country_code, internet_rate  
from alcohol_country_internet_crime  
order by internet_rate desc  
limit 3;
```

3. What is the Average internet usage of all countries with 200 > gallons of beer capita?

```
select avg(internet_rate) as "Average Internet Usage"  
from alcohol_country_internet_crime  
where beer_capita>200;
```

4. What is the Average crime rate of the highest and lowest 3 GDP countries' ranking?

```
(select avg(crime_index) from
```

```

(select crime_index
from country_happiness_gdp_crime
order by year_2020 desc
limit 3 ) as t1)
union all
(select avg(crime_index) from
(select crime_index
from country_happiness_gdp_crime
order by year_2020 asc
limit 3) as t2);

```

5. Which countries have the highest crime rate and what is their happiness index?

```

SELECT happiness_score, country_name, crime_index
from country_happiness_gdp_crime
order by crime_index desc
limit 3;

```

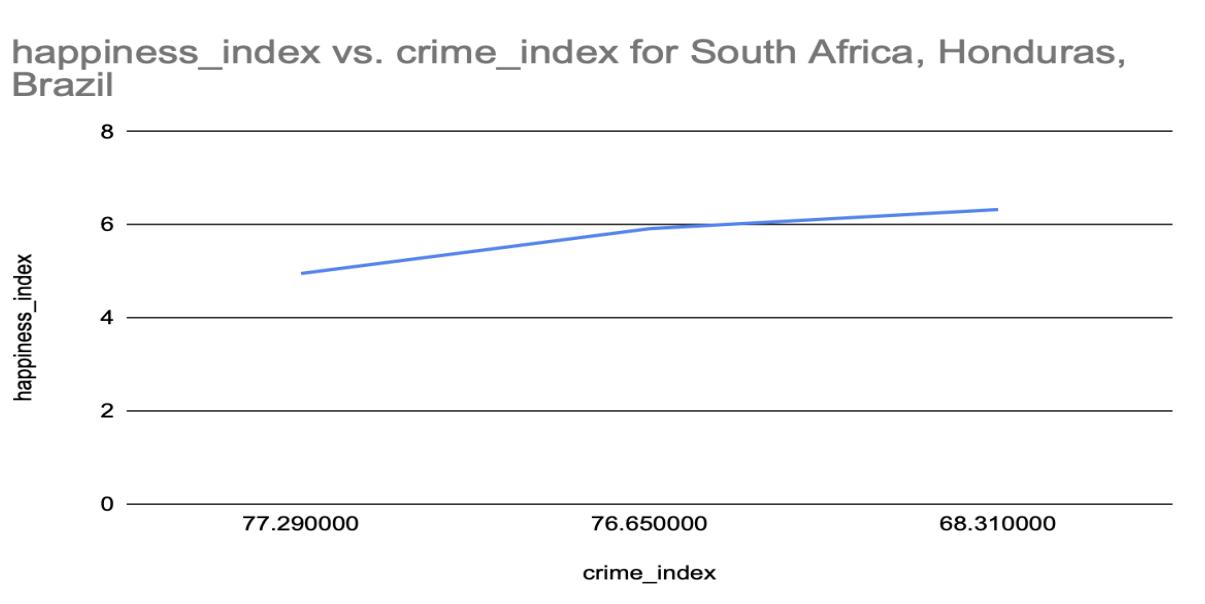
6. What are the top 150 universities located in the highest GDP countries, limited by 3 countries?

```

select uni_name, country_name, year_2021
from university_internet_gdp_country
order by year_2021 desc
limit 150;

```

happiness_index vs. crime_index for South Africa, Honduras, Brazil

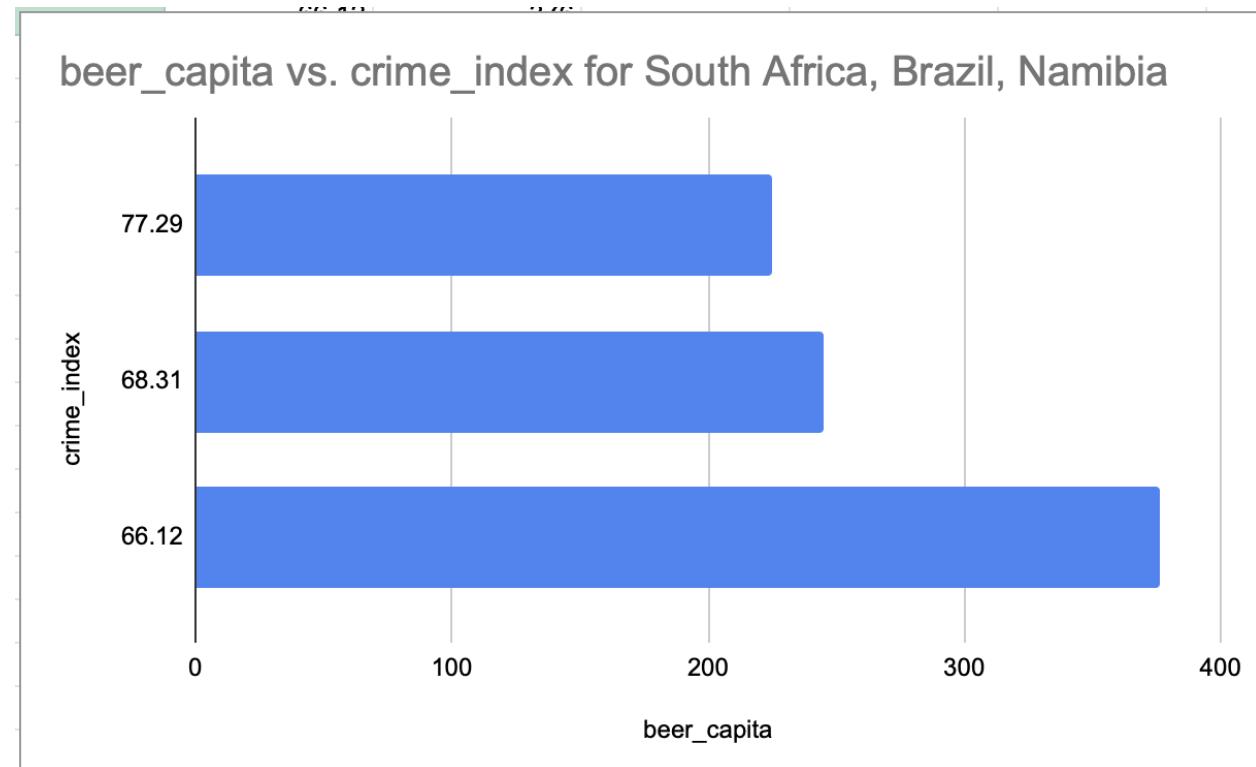


7. Top 3 countries have the most gold medals in 1996 and how's the happiness index in those countries?

```
select country_name, year, gold  
from country_medal_gdp_happiness  
where year=1996  
order by gold desc  
limit 3;
```

8. Top 3 countries have the highest crime rate with alcohol beer_capita consumption>200 gallons?

```
select country_name, crime_index, beer_capita  
from alcohol_country_internet_crime  
where beer_capita>200  
order by crime_index Desc  
limit 3;
```

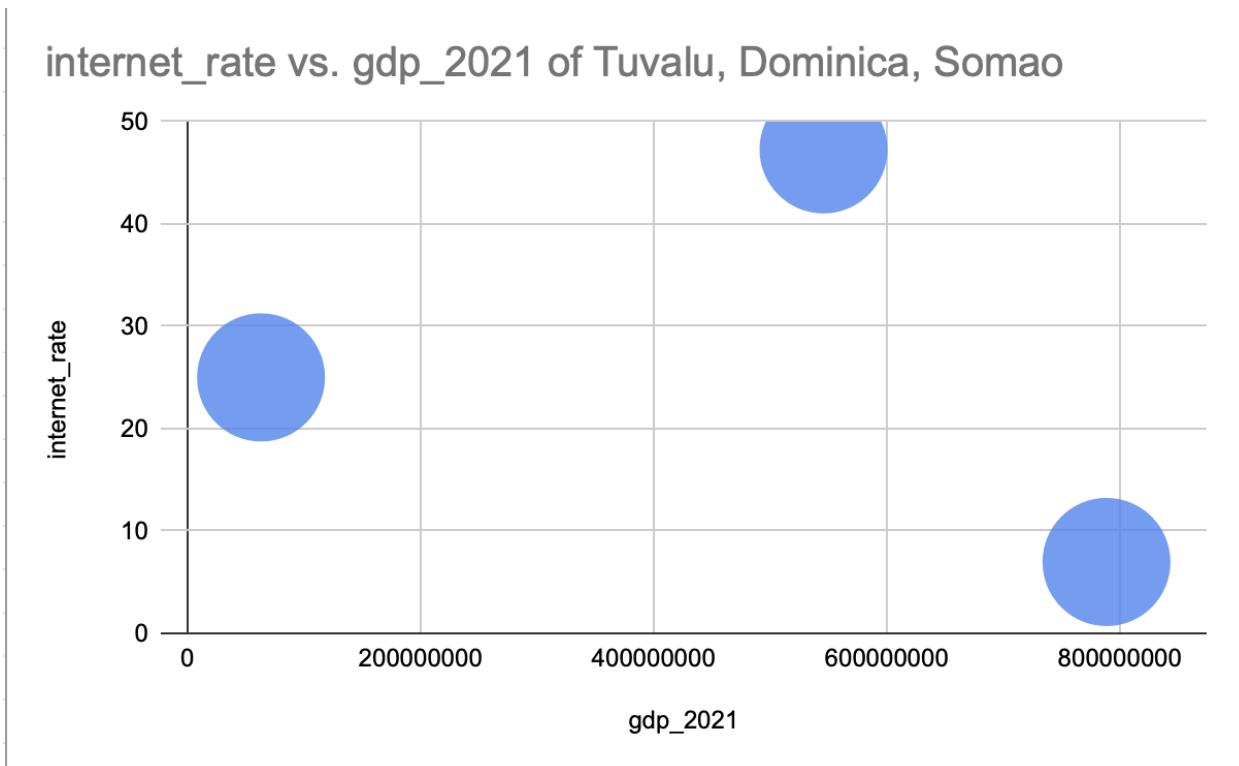


9. What countries have won gold medals since 1995 and how's their GDP in 2021?

```
select country_name, gold, year_2020 as "GDP 2020"  
from country_medal_gdp_happiness  
where gold>0 AND year>1995;
```

10. Internet usage of the highest GDP countries?

```
select country_name, year_2021 as gdp_2021, internet_rate  
from internet_gdp_country_view  
order by year_2021  
limit 3;
```



Indexes

-Country_name

-literacy_rate

Creating indexes-

1. Creating an index for country_name

```
CREATE INDEX country_name  
ON country (country_name);
```

-To see indexes on country-
show indexes from country;

-To see prominence of index-
explain analyze
select * from country
where country_name LIKE "in%";

-Without country Index-

Filter: (country.country_name like 'in%') (cost=25.05 rows=28) (actual time=0.413..0.595 rows=2 loops=1)

Table scan on country (cost=25.05 rows=248) (actual time=0.198..0.431 rows=248 loops=1)

```

26
27
28
29 • select literacy_rate from literacy
30   where literacy_rate > 80;
31
32
33
34
35
36 • create index literacy_rate
37   on literacy(literacy_rate);
38
39
40
41

```

Result Grid

literacy_rate
97.524200
95.000000
97.553900

Action Output

Time	Action	Response	Duration / Fetch Time
24 18:19:48	SELECT * FROM assignment_3.covidstatus	187 rows(s) returned	0.0030 sec / 0.00005...
25 18:21:19	update medals set gold:=gold+1 where ...	1 row(s) affected Rows matched: 1 Change...	0.0081 sec
26 18:21:39	SELECT * FROM assignment_3.literacy ...	187 row(s) returned	0.0017 sec / 0.00002...
27 18:44:48	select literacy_rate from literacy where ...	138 row(s) returned	0.053 sec / 0.00086...

-With country Index-

Filter: (country.country_name like 'in%') (cost=0.66 rows=2) (actual time=0.816..1.086 rows=2 loops=1)

Covering index range scan on country using country_name over ('in' <= country_name <= "in%") (cost=0.66 rows=2) (actual time=0.065..0.333 rows=2 loops=1)

The screenshot shows the MySQL Workbench interface with a query window open. The query is:

```

26
27
28
29 • select literacy_rate from literacy
30   where literacy_rate > 80;
31
32
33
34
35
36 • create index literacy_rate
37   on literacy(literacy_rate);
38
39
40
41

```

The 'Object Info' panel on the left shows the 'Table: literacy' selected. The 'Columns' section lists:

- country_code: char(5) PK
- data_year: int
- literacy_rate: decimal(10,6)
- population: bigint

The 'Action Output' panel at the bottom shows the execution log:

Time	Action	Response	Duration / Fetch Time
✓ 26 13:21:39	SELECT * FROM assignment_3.literacy L...	187 rows(s) returned	0.0017 sec / 0.00002...
✓ 27 13:44:46	select literacy_rate from literacy where ...	138 rows(s) returned	0.053 sec / 0.00086...
✓ 28 13:46:15	create index literacy_rate on literacy(lit...	0 row(s) affected Records: 0 Duplicates: 0...	0.089 sec
✓ 29 13:46:27	select literacy_rate from literacy where ...	138 row(s) returned	0.0032 sec / 0.00004...

2. Creating literacy_rate index

create index literacy_rate
on literacy(literacy_rate);

To see indexes-
show indexes from literacy;

To see prominence of index-
explain analyze
select * from literacy
where literacy_rate>90;

Without literacy_rate index-

Filter: (literacy.literacy_rate > 90.000000) (cost=18.95 rows=123) (actual time=0.170..0.345 rows=123 loops=1)
Table scan on literacy (cost=18.95 rows=187) (actual time=0.159..0.262 rows=187 loops=1)

With literacy_rate index-

Filter: (literacy.literacy_rate > 90.000000) (cost=18.95 rows=123) (actual time=0.092..0.235 rows=123 loops=1)\n

Table scan on literacy (cost=18.95 rows=187) (actual time=0.084..0.169 rows=187 loops=1)\n'

Triggers-

- Covidstatus_trigger
- medals_trigger

Creating Triggers-

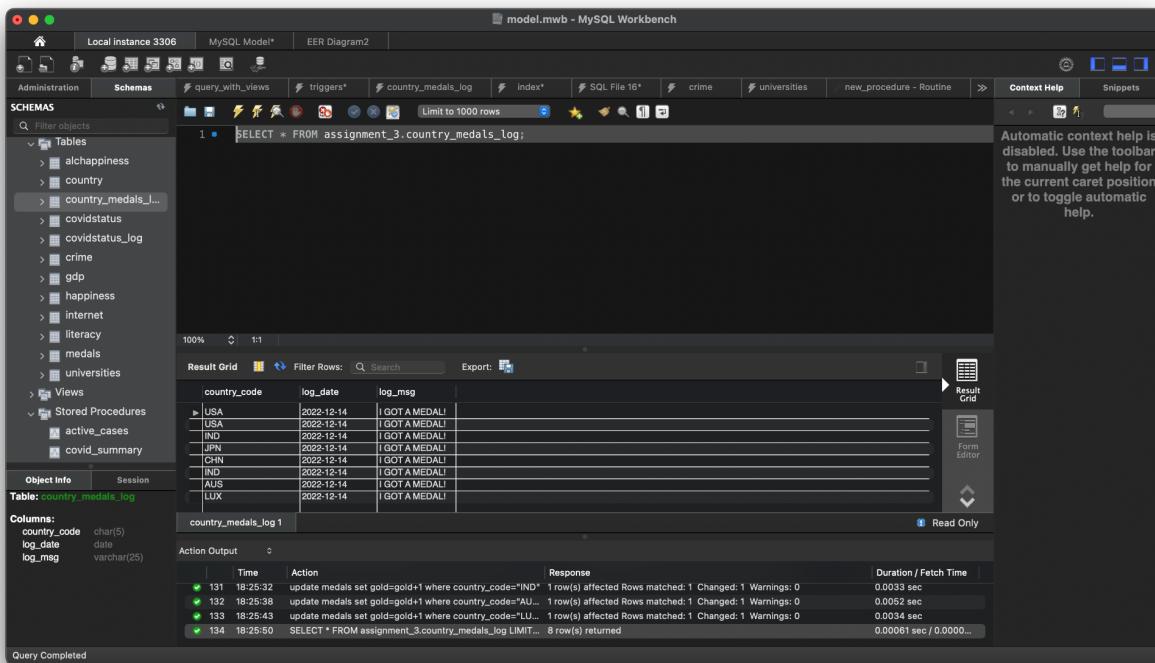
```
drop table if exists country_medals_log;  
drop trigger if exists medals_trigger;  
drop table if exists covidstatus_log;  
drop trigger if exists covidstatus_trigger;
```

1. Creating an Olympic Medals log and Trigger for “Medal updates” if any country has won again.

```
create table country_medals_log(  
country_code CHAR(5),  
log_date DATE,  
log_msg VARCHAR(25)  
);
```

```
delimiter //  
create trigger medals_trigger after update on medals  
for each row  
begin  
insert into country_medals_log(country_code, log_date, log_msg)  
values(NEW.country_code, NOW(), "I GOT A MEDAL!");  
end;  
//  
delimiter ;
```

```
update medals set gold=gold+1 where country_code="USA";
```



2. Creating a Covid log table and Trigger for “Total Cases updates” in any country.

```
create table covidstatus_log(
    country_code CHAR(5),
    log_date DATE,
    log_msg VARCHAR(25),
    case_count INT
);
```

```
delimiter //
```

```
create trigger covidstatus_trigger after update on covidstatus
for each row
```

```
begin
```

```
insert into covidstatus_log(country_code, log_date, log_msg, case_count)
values(NEW.country_code, NOW(), "covid cases updated", NEW.total_cases);
end;
```

```
//
```

delimiter ;

```
update covidstatus set total_cases=total_cases+1 where country_code="ind";
```

The screenshot shows the MySQL Workbench interface with the following details:

- Schemas:** The current schema is "model.mwb".
- Tables:** The "country_medals_log" table is selected.
- Query Editor:** The query "SELECT * FROM assignment_3.covidstatus_log;" is run.
- Result Grid:** The results show the following data:

country_code	log_date	log_msg	case_count
IND	2022-12-14	covid cases updated	44676370
CHN	2022-12-14	covid cases updated	369919
JPN	2022-12-14	covid cases updated	2630154
AUS	2022-12-14	covid cases updated	10870760
DNK	2022-12-14	covid cases updated	3155298

- Action Output:** The log shows the following actions:

Action	Time	Response	Duration / Fetch Time
139	18:27:08	SELECT * FROM assignment_3.covidstatus_log LIMIT 0,...	0.0011 sec / 0.00001...
140	18:27:16	update covidstatus set total_cases=total_cases+1 where...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0 0.0071 sec
141	18:27:25	update covidstatus set total_cases=total_cases+1 where...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0 0.0038 sec
142	18:27:34	SELECT * FROM assignment_3.covidstatus_log LIMIT 0,...	5 row(s) returned 0.00054 sec / 0.000...

Stored procedures-

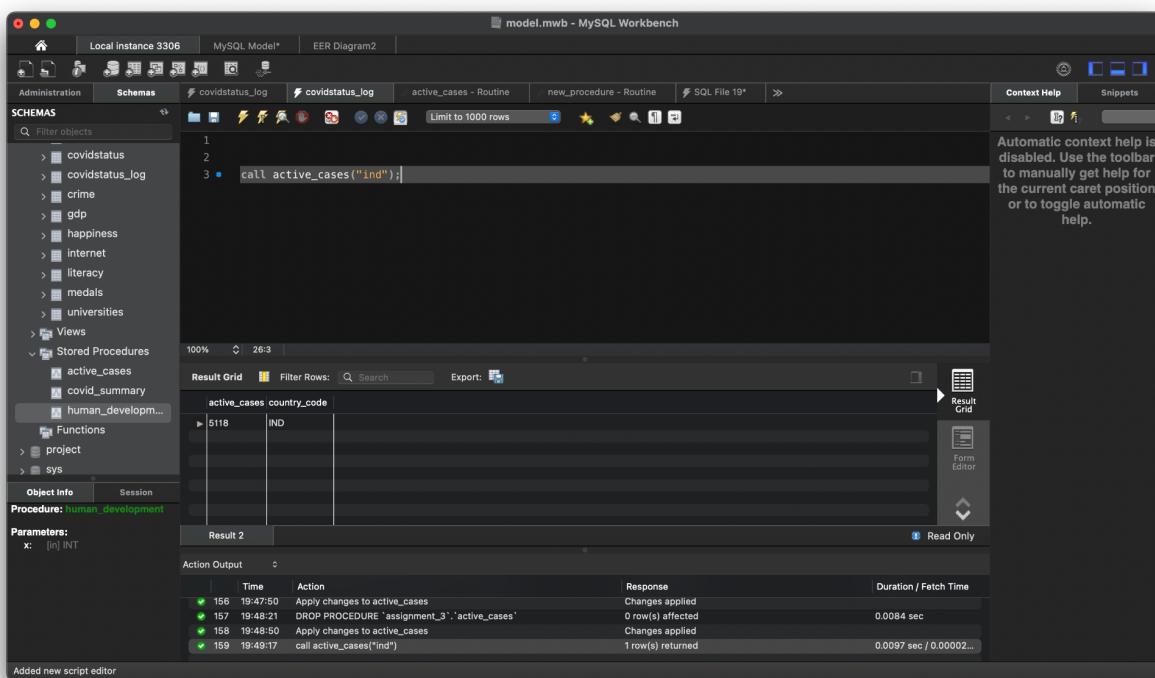
- active_cases**
- human_developmenent**
- covid_summary**

Creating stored procedures

1. Creating a stored procedure to view covid active_cases in any given country-

Create-

```
CREATE PROCEDURE `active_cases`(In cc CHAR(10))
BEGIN
select active_cases, country_code from covidstatus where country_code=cc;
END
```



2. Creating Covid summary to view total cases and total deaths of all countries-

```
CREATE PROCEDURE `covid_summary`()
BEGIN
```

```
select country_code, total_cases, total_deaths from covidstatus;
```

```
END
```

3. Creating human development to view humans enhancement of all countries-

```
CREATE PROCEDURE
```

```
PROCEDURE 'covid_summary'()
```

```
BEGIN
```

```
select country_code, total_cases, total_deaths from covidstatus;
```

```
END
```

-To view stored procedures

```
Call procedure_name;
```

Normalisation

1st Normalization-

- Atomic values
- No duplicate values

Example 1NF-

- Initially, due to a merge of tables, duplicate country values were inserted but through 1st normalization, atomic and unique data was retrieved.
 - Data atomicity was done as columns came in-merged

The screenshot shows the MySQL Workbench interface with a query editor and a results grid.

Query Editor:

```
model.mwb - MySQL Workbench
Local Instance 3306 MySQL Model* EER Diagram1
Administration Schemas medals universities universities alchappiness medals gdp >
Schemas
assignment_3
Tables
alchappiness
country
crime
gdp
happiness
internet
literacy
medals
universities
Views
Stored Procedures
Functions
project
sys
Object Info Session
Table: gdp
Columns:
country_code char(5) PK
year_2020 bigint
year_2021 bigint
SELECT * FROM assignment_3.gdp;
2
3 Select country_code, year_2021, year_2020, COUNT(*) from gdp
4 group by country_code, year_2021, year_2020
5 Having COUNT(*)>1;
6
```

Result Grid:

country_code	year_2021	year_2020	COUNT(*)
AGO	72546985709	53619071176	1
ALB	18260043500	15131866271	1
AND	3329910724	2891001149	1
ARG	491493000000	389591000000	1
ARM	13861183874	12641209802	1
ATG	1471125926	1370281481	1
AUS	154266000000	132784000000	1
AVL	43322100000	43265000000	1
AZE	54622176471	42852000000	1
BDI	2902039366	2760510265	1
BEL	599573000000	521677000000	1
BEN	17785640079	15651545332	1
BFA	19737615114	17933606353	1
BGD	416265000000	373902000000	1

Action Output:

Time	Action	Response	Duration / Fetch Time
59	18:29:48	SELECT *...	84 row(s) returned 0.0033 sec / 0.00002...
60	18:40:16	SELECT *...	175 row(s) returned 0.093 sec / 0.00086...
61	18:43:01	Select cou...	0 row(s) returned 0.060 sec / 0.00082...
62	18:43:16	Select cou...	175 row(s) returned 0.0043 sec / 0.0000...

- Example INF - Happiness table was refined and mapped to country_code that wasn't provided in the data sources..
- Later, country_code was made as a primary key.

Country name	Regional indicator	Ladder score	Standard error of ladder score	upperwhisker	lowerwhisker	Logged GDP per capita	Social support	Healthy life expectancy	Freedom to make life choices	Generosity	Perceptions of corruption	happy	Ladder score in D
Finland	Western Europe	7.842	0.032	7.904	7.780	10.775	0.954	72.000	0.949	-0.098	0.186		
Denmark	Western Europe	7.620	0.035	7.687	7.552	10.933	0.954	72.700	0.946	0.030	0.179		
Switzerland	Western Europe	7.571	0.036	7.643	7.500	11.117	0.942	74.400	0.919	0.025	0.292		
Iceland	Western Europe	7.554	0.059	7.670	7.438	10.878	0.983	73.000	0.955	0.160	0.673		
Netherlands	Western Europe	7.464	0.027	7.518	7.410	10.932	0.942	72.400	0.913	0.175	0.338		
Norway	Western Europe	7.392	0.035	7.462	7.323	11.053	0.954	73.300	0.960	0.093	0.270		
Sweden	Western Europe	7.363	0.036	7.433	7.293	10.867	0.934	72.700	0.945	0.086	0.237		
Luxembourg	Western Europe	7.324	0.037	7.396	7.252	11.647	0.908	72.600	0.907	-0.034	0.386		
New Zealand	North America and ANZ	7.277	0.040	7.355	7.198	10.643	0.948	73.400	0.929	0.134	0.242		
Austria	Western Europe	7.268	0.036	7.337	7.198	10.906	0.934	73.300	0.908	0.042	0.481		
Australia	North America and ANZ	7.183	0.041	7.265	7.102	10.796	0.940	73.900	0.914	0.159	0.442		
Israel	Middle East and North Africa	7.157	0.034	7.224	7.090	10.575	0.939	73.503	0.800	0.031	0.753		
Germany	Western Europe	7.155	0.040	7.232	7.077	10.873	0.903	72.500	0.875	0.011	0.460		
Canada	North America and ANZ	7.103	0.042	7.185	7.021	10.776	0.926	73.800	0.915	0.089	0.415		
Ireland	Western Europe	7.085	0.040	7.164	7.006	11.342	0.947	72.400	0.879	0.077	0.363		
Costa Rica	Latin America and Caribbean	7.069	0.056	7.179	6.960	9.880	0.891	71.400	0.934	-0.126	0.809		
United Kingdom	Western Europe	7.064	0.038	7.138	6.990	10.707	0.934	72.500	0.859	0.233	0.459		
Czech Republic	Central and Eastern Europe	6.965	0.049	7.065	6.668	10.556	0.947	70.807	0.858	-0.208	0.868		
United States	North America and ANZ	6.951	0.049	7.047	6.856	11.023	0.920	68.200	0.837	0.098	0.698		
Belgium	Western Europe	6.834	0.034	6.901	6.767	10.823	0.906	72.199	0.783	-0.153	0.646		
France	Western Europe	6.690	0.037	6.762	6.618	10.704	0.942	74.000	0.822	-0.147	0.571		
Bahrain	Middle East and North Africa	6.647	0.068	6.779	6.514	10.669	0.862	69.495	0.925	0.089	0.722		
Malta	Western Europe	6.602	0.044	6.688	6.516	10.674	0.931	72.200	0.927	0.133	0.653		
Taiwan Province of China	East Asia	6.584	0.038	6.659	6.510	10.871	0.898	69.600	0.784	-0.070	0.721		
United Arab Emirates	Middle East and North Africa	6.561	0.039	6.637	6.484	11.085	0.844	67.333	0.932	0.074	0.589		
Saudi Arabia	Middle East and North Africa	6.494	0.056	6.604	6.384	10.743	0.891	66.603	0.877	-0.149	0.684		
Spain	Western Europe	6.491	0.042	6.574	6.408	10.571	0.932	74.700	0.761	-0.081	0.745		
Italy	Western Europe	6.483	0.045	6.572	6.395	10.623	0.880	73.800	0.693	-0.084	0.866		
Slovenia	Central and Eastern Europe	6.461	0.043	6.546	6.376	10.529	0.948	71.400	0.949	-0.101	0.806		
Guatemala	Latin America and Caribbean	6.435	0.073	6.577	6.292	9.053	0.813	64.958	0.906	-0.038	0.775		
Uruguay	Latin America and Caribbean	6.431	0.046	6.521	6.341	9.966	0.925	69.100	0.896	-0.092	0.590		
Singapore	Southeast Asia	6.377	0.043	6.460	6.293	11.488	0.915	76.953	0.927	-0.018	0.882		
Kosovo	Central and Eastern Europe	6.372	0.059	6.487	6.257	9.318	0.821	63.813	0.869	0.257	0.917		
Slovakia	Central and Eastern Europe	6.351	0.041	6.411	6.251	10.369	0.936	69.201	0.766	-0.124	0.911		
Brazil	Latin America and Caribbean	6.330	0.043	6.415	6.245	9.577	0.882	66.601	0.804	-0.071	0.756		
Mexico	Latin America and Caribbean	6.317	0.053	6.420	6.213	9.859	0.831	68.597	0.862	-0.147	0.799		
Jamaica	Latin America and Caribbean	6.309	0.156	6.615	6.004	9.186	0.877	67.500	0.890	-0.137	0.884		
Lithuania	Central and Eastern Europe	6.255	0.045	6.344	6.167	10.499	0.935	67.906	0.773	-0.203	0.826		
Cyprus	Western Europe	6.223	0.049	6.319	6.128	10.576	0.802	73.898	0.763	-0.015	0.844		
Estonia	Central and Eastern Europe	6.189	0.038	6.263	6.115	10.481	0.941	68.800	0.909	-0.108	0.527		
Panama	Latin America and Caribbean	6.180	0.073	6.323	6.036	10.350	0.896	69.652	0.872	-0.166	0.856		
Uzbekistan	Commonwealth of Independent States	6.179	0.068	6.312	6.045	8.836	0.918	65.255	0.970	0.311	0.515		
Chile	Latin America and Caribbean	6.172	0.046	6.262	6.081	10.071	0.882	70.000	0.742	-0.044	0.830		
Poland	Central and Eastern Europe	6.166	0.040	6.245	6.087	10.382	0.898	69.702	0.841	-0.165	0.735		
Kazakhstan	Commonwealth of Independent States	6.152	0.047	6.243	6.060	10.155	0.952	65.200	0.853	-0.069	0.733		
Romania	Central and Eastern Europe	6.140	0.057	6.253	6.027	10.284	0.832	67.355	0.845	-0.219	0.938		
Kuwait	Middle East and North Africa	6.106	0.066	6.235	5.977	10.817	0.843	66.900	0.867	-0.104	0.736		
Serbia	Central and Eastern Europe	6.078	0.053	6.181	5.974	9.787	0.873	68.600	0.778	0.002	0.835		
El Salvador	Latin America and Caribbean	6.061	0.065	6.188	5.933	9.054	0.762	66.402	0.888	-0.110	0.688		
Mauritius	Sub-Saharan Africa	6.049	0.059	6.165	5.933	10.008	0.905	66.701	0.867	-0.054	0.789		

Example2 INF- Although the count shows more than 1 in medals, it's valid as multiple countries won medals during that year.

The screenshot shows the MySQL Workbench interface with a query editor and a results grid. The message bar at the top right says: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help." The results grid displays data from a query that counts medals by year, showing values for 1968, 1984, 2012, 1990, 1996, and 2000.

year	gold	silver	bronze	COUNT(*)
1968	1	0	0	3
1984	0	1	0	2
2012	1	0	3	4
1990	1	0	0	2
1996	0	1	0	2
2000	0	0	1	2

Result 15

Action Output

Time	Action	Response	Duration / Fetch Time
73 18:58:34	SELECT *	175 row(s) returned	0.029 sec / 0.00048...
74 18:58:34	Select year...	7 row(s) returned	0.0029 sec / 0.00001...
75 18:58:34	Select cou...	1 row(s) returned	0.0041 sec / 0.00002...
76 18:58:41	Select year...	7 row(s) returned	0.0024 sec / 0.00002...

Query Completed

2nd Normalization

- No partial dependencies
- columns must fully depend on primary key, else be moved to new table

2NF

Example1 -

Here in the crime table,

Crime_index and unemployment is fully dependent
on country_code (PK,FK)
(primary key= country_code)

MySQL Workbench Screenshot showing the 'happiness' table:

country_code	freedom	happiness_score
AFG	0.382000	2.523000
ALB	0.785000	5.117000
ARE	0.932000	6.581000
ARG	0.828000	5.929000
ARM	0.825000	5.283000
AUS	0.914000	7.183000
AUT	0.908000	7.268000
AZE	0.814000	5.171000
BDI	0.626000	3.775000
BEL	0.783000	6.834000
BEN	0.757000	5.045000
BFA	0.695000	4.834000
BGR	0.778000	5.266000

2NF

Example 2-

In the happiness table,
changing freedom

Or

Happiness_score has no impact on each other as they fully depend on
country_code (PK, FK.)
(primary key= country_code)

MySQL Workbench Screenshot showing the 'happiness' table:

country_code	freedom	happiness_score
AFG	0.382000	2.523000
ALB	0.785000	5.117000
ARE	0.932000	6.581000
ARG	0.828000	5.929000
ARM	0.825000	5.283000
AUS	0.914000	7.183000
AUT	0.908000	7.268000
AZE	0.814000	5.171000
BDI	0.626000	3.775000
BEL	0.783000	6.834000
BEN	0.757000	5.045000
BFA	0.695000	4.834000
BGR	0.778000	5.266000

3rd normalization

- being in 1NF, 2NF,
- no transitive dependencies

3NF Example1-

Here we have country_code, data_year, literacy_rate, population.

Literacy rate = Divide the number of literates of a given age range by the corresponding age group population and multiply the result by 100

Initially, I thought changing population can affect literacy_rate and transitive dependency could occur.

But the data that was retrieved, has a different year for each country which makes it a unique set overall.

The screenshot shows the MySQL Workbench interface with a query results grid. The query is:

```
1 • SELECT * FROM assignment_3.literacy;
```

The results grid displays the following data:

	country_code	data_year	literacy_rate	population
1	ABW	2015	97.524200	106445
2	AFG	2015	38.168000	41128771
3	AGO	2015	71.164200	35588987
4	AIA	1984	95.000000	15857
5	ALB	2015	97.553900	2842321
6	AND	2011	100.000000	79824
7	ARE	2015	92.986400	9441129
8	ARG	2015	98.990000	45510318
9	ARM	2015	69.344200	2705069
10	ASM	1980	97.344200	44273
11	ATO	2014	98.950000	93763
12	AUS	2003	99.000000	26177415
13	AUT	2011	98.000000	8839617
14	AZE	2015	99.805300	10358074

The Action Output section shows the following log:

Action	Time	Response	Duration / Fetch Time
Select year...	19:05:32	75 row(s) returned	0.0047 sec / 0.00007...
SELECT ...	19:11:19	97 row(s) returned	0.056 sec / 0.00086...
SELECT ...	19:21:09	129 row(s) returned	0.058 sec / 0.00064...
SELECT ...	19:28:34	187 row(s) returned	0.048 sec / 0.00082...

3NF Example 2-

Here in the table Alchappiness,

Country_code PK FK

Human_development

Beer_capita

wine_capita

Are fully dependent on Primary key and no transitive dependencies are established.

The screenshot shows the MySQL Workbench interface with the 'model.mwb' database open. The left sidebar displays the schema 'assignment_3' with its tables: alchappiness, country, crime, gdp, happiness, internet, literacy, medals, universities, and views. The 'alchappiness' table is selected. The main pane shows the results of the query: 'SELECT * FROM assignment_3.alchappiness;'. The results grid contains the following data:

country_code	human_development	beer_capita	wine_capita
ALB	782.000000	85.000000	54.000000
ARE	892.000000	16.000000	5.000000
ARG	822.000000	193.000000	221.000000
ARM	748.000000	21.000000	11.000000
AUS	938.000000	261.000000	212.000000
AUT	906.000000	279.000000	191.000000
AZE	757.000000	21.000000	5.000000
BEL	915.000000	295.000000	212.000000
BGR	810.000000	281.000000	94.000000
BHR	846.000000	42.000000	7.000000
BIH	766.000000	76.000000	8.000000
BLR	805.000000	142.000000	42.000000
BLZ	709.000000	283.000000	8.000000
BOL	689.000000	167.000000	8.000000

The bottom pane shows the 'Action Output' log with the following entries:

Time	Action	Response	Duration / Fetch Time
19:49:05	SELECT ...	156 row(s) returned	0.054 sec / 0.0008...
19:49:13	SELECT ...	156 row(s) returned	0.0035 sec / 0.00002...
19:49:15	SELECT ...	187 row(s) returned	0.0015 sec / 0.00002...
19:49:19	SELECT ...	105 row(s) returned	0.0048 sec / 0.00001...