---------COVID-19 DATA ANALYSIS USING MY SQL--------

Creating the database

---create database pandas_db;

---use pandas_db;

FOR, YEARLY, MONTHLY, WEEKLY ANALYSIS OF COVID DATA ACROSS STATES

Creating table for Andhra Pradesh

---create table AN(dates date ,confirmed float, recovered float, tested float, other float, deceased float, vaccinated1 float, vaccinated2 float);

---select * from AN;

dates	confirmed	recovered	tested	other	deceased	vaccinated1	vaccinated2
2021-10-21	7646	7510	584772	NULL	129	292802	183786
2021-10-22	7646	7511	586240	NULL	129	292988	186655
2021-10-23	7648	7511	587939	NULL	129	293112	189393
2021-10-24	7648	7513	589097	NULL	129	293117	189517
2021-10-25	7648	7514	590432	NULL	129	293347	191929
2021-10-26	7648	7515	591576	NULL	129	293512	194041
2021-10-27	7650	7516	592748	NULL	129	293644	195689
2021-10-28	7651	7516	594049	NULL	129	293776	197310
2021-10-29	7651	7517	595397	NULL	129	293904	198731
2021-10-30	7651	7518	596657	NULL	129	293998	200144
2021-10-31	7651	7518	598033	NULL	129	294001	200157

Extracting year, month, from dates column from above table and one more column based on if the day in above date is between 1 and 7 then week number is one and so on…

Also grouping the above result by year, month and weeknum to see the total types of cases in Andhra Pradesh

---create table table1 as(

with cte as(

```
select year(dates) as Year_ ,month(dates) as monthnum ,
case when day(dates)>=1 and day(dates)<8 then 1
when day(dates)>=8 and day(dates)<15 then 2
when day(dates)>=15 and day(dates)<22 then 3
else 4 end as weeknum,
 sum(confirmed) as sc ,
 sum(recovered) as sr, sum(tested) as st, sum(other) as so ,
 sum(deceased) as sd , sum(vaccinated1) as sv1,
 sum(vaccinated2) as sv2 from AN group by 1,2,3 order by 1,2,3)
 select Year_,monthnum,weeknum, coalesce(sc, 0) as total_confirmed,
 coalesce(sr, 0) as total_recovered, coalesce(st, 0) as total_tested,
 coalesce(so, 0) as total_others, coalesce(sd, 0) as total_deceased,
 coalesce(sv1, 0) as total_vaccinated1, coalesce(sv2, 0) as total_vaccinated2
 from cte order by 1,2,3);
 select * from table1;
 alter table table1 add column state_names varchar(5);
 update table1 set state_names='AN';
 select * from table1;
```

The final result of Andhra Pradesh Table

Year_	monthnum	weeknum	total_confirmed	total_recovered	total_tested	total_others	total_deceased	total_vaccinated1	total_vaccinated2	state_names
2020	3	4	45	0	0	0	0	0	0	AN
2020	4	1	70	0	0	0	0	0	0	AN
2020	4	2	77	60	0	0	0	0	0	AN
2020	4	3	96	76	7015	0	0	0	0	AN
2020	4	4	267	112	22235	0	0	0	0	AN
2020	5	1	231	203	26278	0	0	0	0	AN
2020	5	2	231	231	26278	0	0	0	0	AN
2020	5	3	231	231	44999	0	0	0	0	AN
2020	5	4	330	330	74586	0	0	0	0	AN
2020	6	1	231	231	59572	0	0	0	0	AN
2020	6	2	256	231	72444	0	0	0	0	AN
2020	6	3	316	241	87872	0	0	0	0	AN
2020	6	4	626	380	131152	0	0	0	0	AN

I similarly made few fore tables for some more states like Delhi, Chandigarh, Bihar, Arunachal Pradesh by the same procedure…

FOR DELTA 7 CONFIRMED CASES VS VACCINATED CASES ACROSS STATES

---create table DELTA(state_codes varchar(5) ,confirmed float, recovered float,

tested float, vaccinated1 float, vaccinated2 float, deceased float, other float);

---select * from DELTA;

state_codes	confirmed	recovered	tested	vaccinated1	vaccinated2	deceased	other
AN	3	5	8936	884	10640	NULL	NULL
AP	2873	3590	254532	1223010	1887000	30	NULL
AR	66	97	4788	3312	23647	NULL	NULL
AS	2056	2215	269097	274869	849889	24	NULL
BR	40	31	1378540	1286710	2144970	NULL	NULL
CH	28	20	10726	3680	21641	NULL	NULL
CT	205	103	147451	379374	604260	5	NULL
DL	267	239	395086	160323	269146	NULL	NULL
DN	NULL	2	NULL	2802	14244	NULL	NULL
GA	222	409	19026	8418	46494	6	NULL
GJ	159	116	328489	335172	1660380	2	NULL
HP	1537	1154	64352	13244	234011	20	-1
HR	95	87	148110	160777	368141	NULL	NULL

Just replacing null values to 0 using coalesce from above table in the below query

select state_codes, coalesce(confirmed,0) as confirmed ,

coalesce(recovered,0) as recovered, coalesce(tested,0) as tested,

coalesce(vaccinated1,0) as vaccinated1, coalesce(vaccinated2,0) as

vaccinated2, coalesce(deceased,0) as deceased , coalesce(other,0) as other

from DELTA;

state_codes	confirmed	recovered	tested	vaccinated1	vaccinated2	deceased	other
AN	3	5	8936	884	10640	0	0
AP	2873	3590	254532	1223010	1887000	30	0
AR	66	97	4788	3312	23647	0	0
AS	2056	2215	269097	274869	849889	24	0
BR	40	31	1378540	1286710	2144970	0	0
CH	28	20	10726	3680	21641	0	0
CT	205	103	147451	379374	604260	5	0
DL	267	239	395086	160323	269146	0	0
DN	0	2	0	2802	14244	0	0
GA	222	409	19026	8418	46494	6	0
GJ	159	116	328489	335172	1660380	2	0
HP	1537	1154	64352	13244	234011	20	-1
HR	95	87	148110	160777	368141	0	0

FOR SEVERINITY OR CONDITION OF COVID-19 CASES ACROSS STATES

Replacing the null values with 0 using coalesce

---create table new(state_codes varchar(5) ,confirmed float, deceased float,recovered float, tested float, vaccinated1 float, vaccinated2 float, other float);

---select * from new; select state_codes, confirmed, deceased, recovered, tested, vaccinated1, vaccinated2, coalesce(other,0) AS other from new;

state_codes	confirmed	deceased	recovered	tested	vaccinated1	vaccinated2	other
AN	7651	129	7518	598033	294001	200157	0
AP	2066450	14373	2047720	29518800	32977000	20375200	0
AR	55155	280	54774	1185440	771875	534486	0
AS	610645	5997	600974	24712000	20172500	8068800	1347
BR	726098	9661	716390	50531800	49874800	18346800	1
CH	65351	820	64495	792851	926035	546981	0
CT	1006050	13577	992159	13709500	14851700	7343270	0
DL	1439870	25091	1414430	29427800	13055600	7425400	0
DN	10681	4	10644	72410	660753	370255	31
GA	178108	3364	174392	1468400	1262570	911114	0
GJ	826577	10089	816283	30928100	44735200	25972400	0
HP	224106	3738	218410	3685010	5713700	3443820	16
HR	771252	10049	761068	13032500	17772400	8115460	0

FOR MONTHWISE CONFIRMED COVID-19 CASES ACROSS STATES

The raw table

create table r4(state_dates varchar(30) ,confirmed float);

select * from r4;

state_dates	confirmed
AN.dates.2020-03-26.delta	1
AN.dates.2020-03-26.delta7	1
AN.dates.2020-03-26.total	1
AN.dates.2020-03-27.delta	5
AN.dates.2020-03-27.delta7	6
AN.dates.2020-03-27.total	6
AN.dates.2020-03-28.delta	3
AN.dates.2020-03-28.delta7	9
AN.dates.2020-03-28.total	9
AN.dates.2020-03-29.delta7	9
AN.dates.2020-03-29.total	9
AN.dates.2020-03-30.delta	1
AN.dates.2020-03-30.delta7	10

Extracting state names, month, monthname from above table and then grouping by month to find total confirmed cases in each month

with cte as (select left(state_dates,2) as state_code, mid(state_dates, 10,10) as dates, confirmed from r4), cte2 as(select state_code, month(dates) as month_number,monthname(dates) as monthname_, sum(confirmed) as total_confirmed from cte group by 1,2,3 order by 1,2)

SELECT monthname_ ,sum(total_confirmed) as total_confirmed from cte2 group by 1 order by 2 desc;

	monthname_	total_confirmed
▶	October	2608869272
	September	2362205125
	August	2214353230
	July	2027818293
	June	1842037776
	May	1683730731
	April	994012432
	March	730781950
	January	661768876
	December	629970154
	February	617503352
	November	552515274

FOR CATEGORIZING DISTRICTS INTO 5 CATEGORIES A,B,C,D,E BASED ON TESTING RATIO

Testing ratio = number of tested people/ population

The raw table

create table r5(districts varchar(255) ,population float,

tested1 varchar(50),tested float);

select * from r5;

	districts	population	tested
	AP.districts.Anantapur	4083320	787085
	AP.districts.Chittoor	4170470	780322
	AP.districts.East Godavari	5151550	944746
	AP.districts.Foreign Evac...	NULL	NULL
	AP.districts.Guntur	4889230	833823
	AP.districts.Krishna	4529010	782232
	AP.districts.Kurnool	4046600	867361
	AP.districts.Other State	NULL	NULL
	AP.districts.Prakasam	3392760	697340
	AP.districts.S.P.S. Nellore	2966080	682964
	AP.districts.Srikakulam	2699470	670899
	AP.districts.Visakhapatnam	4288110	744983
	AP.districts.Vizianagaram	2342870	531433

Selecting rows which are not null and selecting only district name from districts column using substring function

WITH CTE AS(select * from r5 where population is not null and

tested is not null)

select SUBSTRING(districts,14,20) as district, population,

tested from cte;

district	population	tested
Anantapur	4083320	787085
Chittoor	4170470	780322
East Godavari	5151550	944746
Guntur	4889230	833823
Krishna	4529010	782232
Kurnool	4046600	867361
Prakasam	3392760	697340
S.P.S. Nellore	2966080	682964
Srikakulam	2699470	670899
Visakhapatnam	4288110	744983
Vizianagaram	2342870	531433
West Godavari	3934780	793098
Y.S.R. Kadapa	2884520	671761

The testing ratio is calculated in excel workbook successfully and also the further analysis is done in excel workbook only.

----------THE END----------