```sql
select * from time_series

--Vaccination 1 and Vaccination 2
select  State_Code,(total_vaccinated1*100.0/meta_population)
Vaccination1_percentage,
(total_vaccinated2*100.0/meta_population) Vaccination2_percentage
from State_covid19

select *,sum(Vaccinated1) over (order by date) as Total_vaccination1,
sum(Vaccinated2) over (order by date) as Total_vaccination2
from (
select Date,sum(delta_vaccinated1) as Vaccinated1,sum(delta_vaccinated2) as
Vaccinated2 from time_series
group by Date) as A
where Vaccinated1 is not null

------- Confirmed Cases wrt time
select CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date)) as Date,
SUM(delta_confirmed) As Confirmed
from time_series
group by DATENAME(MONTH,Date),DATENAME(YEAR,Date)
order by MIN(date)

--delta_deceased
select CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date)) as Date,
SUM(delta_deceased) As deceased
from time_series
group by DATENAME(MONTH,Date),DATENAME(YEAR,Date)
order by MIN(date)

--testing
select CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date)) as Date,
SUM(delta_tested) As Testing
from time_series
group by DATENAME(MONTH,Date),DATENAME(YEAR,Date)
order by MIN(date)
```

```
--Vaccination1

select CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date)) as Date,
SUM(delta_vaccinated1) As Vaccination1
from time_series
group by DATENAME(MONTH,Date),DATENAME(YEAR,Date)
order by min(date)

--Vaccination2

select CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date)) as Date,
SUM(delta_vaccinated2) As Vaccination2
from time_series
group by DATENAME(MONTH,Date),DATENAME(YEAR,Date)
order by min(date)


select * from #Vaccination1 as A inner join
#Vaccination2 As B on
A.Date=B.Date
```

```sql
---Recovered Cases wrt to time
select CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date)) as Date,
SUM(delta_recovered) As Recovered
from time_series
group by DATENAME(MONTH,Date),DATENAME(YEAR,Date)
order by MIN(date)



select * into #Categories from (
select District, (total_tested*100.0/population) as Percentage_tesing, case when
(total_tested*100.0/population) between 0 and 10 then 'Category_A'
when (total_tested*100.0/population) between 10 and 30 then 'Category_B'
when (total_tested*100.0/population) between 30 and 50 then 'Category_C'
when (total_tested*100.0/population) between 50 and 76 then 'Category_D'
when (total_tested*100.0/population) between 76 and 100 then 'Category_E'
else 'Category_F'
end As Category
from district1
where total_tested is not null
) as A

select Category,Count(District)as Number_of_states from #Categories
group by Category

Select Category,count(District) as Number_of_District,Avg(Percentage_tesing) as
Percentage_tesing,
Avg(Percentage_decesed) as Percentage_decesed from
(select
A.District,A.Category,A.Percentage_tesing,(b.total_deceased*100.0/B.Population)
as Percentage_decesed
from #Categories as A inner join  district1 as b
on A.District=B.District
) as a
group by Category
```

```sql
-- Deaths vs Confirmed cases
select CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date)) as Date,
Avg(total_deceased*100.0/total_confirmed) As Deceased_Rate
from time_series
group by CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date))
order by MIN(Date)

-- Deaths vs Recovered cases
select CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date)) as Date,
Avg(total_recovered*100.0/total_confirmed) As Recovered_Rate
from time_series
group by DATENAME(month,Date),DATENAME(YEAR,Date)
order  by MIN(Date)


--Confirmed  vs tested cases
select CONCAT_WS('-', DATENAME(month,Date),DATENAME(YEAR,Date)) as Date,
Avg(total_confirmed*100.0/total_tested) As Confirmation_Rate
from time_series
group by DATENAME(month,Date),DATENAME(YEAR,Date)
order by MIN(Date)


--Most Deaths per population state_wise
select State_Code,round((total_deceased*100.0/meta_population),3) as
Deceased_percentage from State_covid19
order by Deceased_percentage desc

-- Vaccinated and Death Rates
select State_Code,round((total_vaccinated2*100.0/meta_population),3) as
Vaccination
,round((total_deceased*100.0/meta_population),3) as Deceased from
State_covid19
order by Vaccination desc
```

```sql
--tested vs deaths
select State_Code,round((total_tested*100.0/meta_population),3) as tested
,round((total_deceased*100.0/meta_population),3) as Deceased from
State_covid19
where meta_population > 500000
order by tested desc

--statewise Testing
select State_Code,cast (round((total_tested*1.00/meta_population),3) as float) as
tested
 from State_covid19
where meta_population > 500000
order by State_Code asc
```