**#1 Write a query to determine the number of startups in each nation.**

SELECT country\_code AS Country,count(market) AS startups

FROM investments\_vc

WHERE country\_code <> ' '

GROUP BY country\_code

ORDER BY count(market) desc;

**#2 Write a Query to find out how many startups are there under each category.**

SELECT market,count(market) AS Companies

FROM investments\_vc

WHERE market <> ' '

GROUP BY market

ORDER BY count(market) desc;

**#3 Write a query to learn which company received the second-highest funding.**

SELECT Name AS Company,funding\_total\_usd AS Fundings

FROM investments\_vc

WHERE funding\_total\_usd <(SELECT Max(funding\_total\_usd) FROM investments\_vc)

ORDER BY funding\_total\_usd DESC

LIMIT 1

;

**#4 Write a query to find out which markets have gotten capital totaling more than $70 million on average.**

SELECT market,avg(funding\_total\_usd) AS AVG\_funding\_in\_million

FROM investments\_vc

GROUP BY market

HAVING avg(funding\_total\_usd) > 70

ORDER BY AVG\_funding\_in\_million DESC;

**#5 Create a query to list the top 5 companies that received the most funding from venture capital.**

SELECT name,venture,dense\_rank() OVER (ORDER BY venture DESC) as Dense\_Top

FROM investments\_vc

ORDER BY venture DESC

LIMIT 5;

**#6 Create a query to find how many days have passed since the company was founded?**

SELECT Name,founded\_at,current\_date(),

Datediff(current\_date(),founded\_at) AS Days\_Since\_Founded

FROM investments\_vc

WHERE founded\_at <> '';

**#7 Create a query to learn how much money a company received from seed investment and angle investors throughout the early stages of development.**

SELECT Name,sum(seed+angel) AS Funding

FROM investments\_vc

GROUP BY name;

**#8 Create a query to determine the typical time it takes for each category to receive its initial financing.**

SELECT a.Market,

AVG(a.Number\_of\_Days) AS avg\_days\_to\_receive\_funding

FROM(SELECT Market,(Datediff(first\_funding\_at,founded\_at)) AS Number\_of\_Days

FROM investments\_vc

WHERE founded\_at <> '') AS a

GROUP BY a.market

ORDER BY avg\_days\_to\_receive\_funding ASC;

**#9 Write a query to find out how many startups are founded every 20 years.**

SELECT

SUM(CASE WHEN Founded\_Year > 1993 THEN 1 ELSE 0 END) AS 1994\_TO\_2014,

SUM(CASE WHEN Founded\_Year > 1972 AND Founded\_Year < 1994 THEN 1 ELSE 0 END) AS 1973\_TO\_1993,

SUM(CASE WHEN Founded\_Year > 1951 AND Founded\_Year < 1973 THEN 1 ELSE 0 END) AS 1952\_TO\_1972,

SUM(CASE WHEN Founded\_Year > 1930 AND Founded\_Year < 1952 THEN 1 ELSE 0 END) AS 1931\_TO\_1951,

SUM(CASE WHEN Founded\_Year > 1909 AND Founded\_Year < 1931 THEN 1 ELSE 0 END) AS 1910\_TO\_1930

FROM investments\_vc;

**#10 Using CTE, create a query to find companies that have received more than average private equity funding.**

WITH CTE1 AS (

SELECT name, private\_equity FROM investments\_vc

WHERE private\_equity > (SELECT AVG(private\_equity) FROM investments\_vc))

SELECT \* FROM CTE1;

**#11 Create a query to change the date format to 'DD-MM-YYYY'.**

SELECT Founded\_at,DATE\_FORMAT(Founded\_at, '%d/%m/%Y') AS Formatted\_date

FROM investments\_vc

WHERE founded\_at <> '';

**#12 Create a query to find the most money raised in each market and compare it to a company in the same market**.

SELECT e.name,e.market,e.funding\_total\_usd,

max(funding\_total\_usd) OVER (partition by market) AS Maximum\_Funding\_in\_that\_Market

FROM investments\_vc e

WHERE market <> ''

ORDER BY market;

**#13 Write a query to find the maximum and minimum funding raised by each market,rank them on the basis of fundind as well as the company name and its city, region, and country**.

SELECT e.name,e.city,e.region,e.country\_code,e.market,e.funding\_total\_usd,

max(funding\_total\_usd) OVER (partition by market) AS Maximum\_Funding\_in\_that\_Market,

min(funding\_total\_usd) OVER (partition by market) AS Minimum\_Funding\_in\_that\_Market

FROM investments\_vc e

WHERE market <> '';

**#14 create a query to rank all the companies based on the amount they have received.**

SELECT x.Name,x.Country\_code,Ranking,

Case when x.Ranking = 1 Then 'Good'

when x.Ranking = 2 Then 'Avg'

when x.Ranking = 3 Then 'Low' END Ranking

FROM(SELECT name,country\_code ,

ntile(3) over (ORDER BY Venture DESC) AS Ranking

FROM investments\_vc

) as x;

**#15 In Venture capital, create a query to convert an integer value to dollar currency.**

SELECT CONCAT('$', FORMAT(venture,0)) AS fundingdollars\_VentureCapital

FROM investments\_vc;