**Задания**

В самостоятельном проекте вам нужно проанализировать данные о фондах и инвестициях и написать запросы к базе. Задания будут постепенно усложняться, но всё необходимое для их выполнения: операторы, функции, методы работы с базой — вы уже изучили на курсе.

1.Посчитайте, сколько компаний закрылось.

Решение:

SELECT COUNT(status)

FROM company

WHERE status = 'closed'

2.Отобразите количество привлечённых средств для новостных компаний США. Используйте данные из таблицы company. Отсортируйте таблицу по убыванию значений в поле funding\_total .

Решение:

SELECT funding\_total

FROM company

WHERE category\_code = 'news'

AND country\_code = 'USA'

ORDER BY funding\_total DESC;

3.Найдите общую сумму сделок по покупке одних компаний другими в долларах. Отберите сделки, которые осуществлялись только за наличные с 2011 по 2013 год включительно.

Решение:

SELECT SUM(price\_amount)

FROM acquisition

WHERE term\_code = 'cash'

AND EXTRACT(YEAR FROM CAST (acquired\_at AS date)) between '2011' and '2013'

4.Отобразите имя, фамилию и названия аккаунтов людей в твиттере, у которых названия аккаунтов начинаются на 'Silver'.

Решение:

SELECT first\_name,

last\_name,

twitter\_username

FROM people

WHERE twitter\_username LIKE 'Silver%'

5.Выведите на экран всю информацию о людях, у которых названия аккаунтов в твиттере содержат подстроку 'money', а фамилия начинается на 'K'.

Решение:

SELECT \*

FROM people

WHERE twitter\_username LIKE '%money%'

AND last\_name LIKE 'K%'

6.Для каждой страны отобразите общую сумму привлечённых инвестиций, которые получили компании, зарегистрированные в этой стране. Страну, в которой зарегистрирована компания, можно определить по коду страны. Отсортируйте данные по убыванию суммы.

Решение:

SELECT country\_code,

SUM(funding\_total)

FROM company

GROUP BY country\_code

ORDER BY SUM(funding\_total)DESC

7.Составьте таблицу, в которую войдёт дата проведения раунда, а также минимальное и максимальное значения суммы инвестиций, привлечённых в эту дату.

Оставьте в итоговой таблице только те записи, в которых минимальное значение суммы инвестиций не равно нулю и не равно максимальному значению.

Решение:

SELECT CAST (funded\_at AS DATE),

MIN(raised\_amount),

MAx(raised\_amount)

FROM funding\_round

GROUP BY funded\_at

HAVING MIN(raised\_amount)<>MAx(raised\_amount) AND MIN(raised\_amount)<> 0

8.Создайте поле с категориями:

* Для фондов, которые инвестируют в 100 и более компаний, назначьте категорию high\_activity.
* Для фондов, которые инвестируют в 20 и более компаний до 100, назначьте категорию middle\_activity.
* Если количество инвестируемых компаний фонда не достигает 20, назначьте категорию low\_activity.

Отобразите все поля таблицы fund и новое поле с категориями.

Решение:

SELECT \*,

CASE

WHEN invested\_companies>=100 THEN 'high\_activity'

WHEN invested\_companies>=20 THEN 'middle\_activity'

ELSE 'low\_activity'

END

FROM fund

9.Для каждой из категорий, назначенных в предыдущем задании, посчитайте округлённое до ближайшего целого числа среднее количество инвестиционных раундов, в которых фонд принимал участие. Выведите на экран категории и среднее число инвестиционных раундов. Отсортируйте таблицу по возрастанию среднего.

Решение:

SELECT

CASE

WHEN invested\_companies>=100 THEN 'high\_activity'

WHEN invested\_companies>=20 THEN 'middle\_activity'

ELSE 'low\_activity'

END AS activity,

ROUND(AVG(investment\_rounds))

FROM fund

GROUP BY activity

ORDER BY ROUND(AVG(investment\_rounds))

10.Проанализируйте, в каких странах находятся фонды, которые чаще всего инвестируют в стартапы.

Для каждой страны посчитайте минимальное, максимальное и среднее число компаний, в которые инвестировали фонды этой страны, основанные с 2010 по 2012 год включительно. Исключите страны с фондами, у которых минимальное число компаний, получивших инвестиции, равно нулю. Выгрузите десять самых активных стран-инвесторов.

Отсортируйте таблицу по среднему количеству компаний от большего к меньшему, а затем по коду страны в лексикографическом порядке.

Решение:

SELECT country\_code,

MIN(invested\_companies),

MAX(invested\_companies),

AVG(invested\_companies)

FROM fund

WHERE EXTRACT (YEAR FROM founded\_at) BETWEEN '2010' AND '2012'

GROUP BY country\_code

HAVING MIN(invested\_companies)>0

ORDER BY AVG(invested\_companies)DESC

LIMIT 10

11.Отобразите имя и фамилию всех сотрудников стартапов. Добавьте поле с названием учебного заведения, которое окончил сотрудник, если эта информация известна.

Решение:

SELECT p.first\_name,

p.last\_name,

e.instituition

FROM people AS p

LEFT OUTER JOIN education AS e ON p.id = e.person\_id

12.Для каждой компании найдите количество учебных заведений, которые окончили её сотрудники. Выведите название компании и число уникальных названий учебных заведений. Составьте топ-5 компаний по количеству университетов.

Решение:

SELECT c.name,

count(DISTINCT e.instituition) as ff

FROM people AS p

JOIN education AS e ON p.id = e.person\_id

JOIN company AS c ON c.id = p.company\_id

WHERE p.company\_id is not null AND e.person\_id is not null

Group by c.name

ORDER BY ff DESC

LIMIT 5

13.Составьте список с уникальными названиями закрытых компаний, для которых первый раунд финансирования оказался последним.

Решение:

SELECT DISTINCT name

FROM company

WHERE status = 'closed'

AND

id IN (SELECT company\_id

FROM funding\_round

WHERE is\_first\_round = 1

AND is\_last\_round = 1)

14.Составьте список уникальных номеров сотрудников, которые работают в компаниях, отобранных в предыдущем задании.

Решение:

SELECT id

FROM people

WHERE company\_id IN (SELECT id

FROM company

WHERE status = 'closed'

AND id IN (SELECT company\_id

FROM funding\_round

WHERE is\_first\_round = 1

AND is\_last\_round = 1))

15.Составьте таблицу, куда войдут уникальные пары с номерами сотрудников из предыдущей задачи и учебным заведением, которое окончил сотрудник.

Решение:

SELECT DISTINCT p.id,

e.instituition

FROM(SELECT id

FROM people

WHERE company\_id IN (SELECT id

FROM company

WHERE status = 'closed'

AND id IN (SELECT company\_id

FROM funding\_round

WHERE is\_first\_round = 1

AND is\_last\_round = 1

GROUP BY company\_id))) AS p

join education as e ON p.id=e.person\_id

16.Посчитайте количество учебных заведений для каждого сотрудника из предыдущего задания. При подсчёте учитывайте, что некоторые сотрудники могли окончить одно и то же заведение дважды.

Решение:

SELECT DISTINCT person\_id AS person,

COUNT(instituition)

FROM education

WHERE person\_id IN (SELECT id

FROM people

WHERE company\_id IN

(SELECT id

FROM company

WHERE status = 'closed'

AND id IN

(SELECT company\_id

FROM funding\_round

WHERE is\_first\_round = 1

AND is\_last\_round = 1)))

GROUP BY person;

17.Дополните предыдущий запрос и выведите среднее число учебных заведений (всех, не только уникальных), которые окончили сотрудники разных компаний. Нужно вывести только одну запись, группировка здесь не понадобится.

Решение:

SELECT AVG (n\_educations) AS avg\_ed

FROM

(SELECT pp.id, COUNT(ed.instituition) AS n\_educations

FROM people AS pp

INNER JOIN company AS c ON pp.company\_id = c.id

INNER JOIN education AS ed ON ed.person\_id = pp.id

WHERE c.name IN

(SELECT DISTINCT co.name

FROM company AS co

INNER JOIN funding\_round AS fr ON fr.company\_id=co.id

WHERE co.status LIKE 'closed'

AND fr.is\_first\_round = fr.is\_last\_round

AND fr.is\_first\_round = 1

AND co.name IS NOT NULL)

GROUP BY pp.id) AS table\_1;

18.Напишите похожий запрос: выведите среднее число учебных заведений (всех, не только уникальных), которые окончили сотрудники Facebook\*.

\*(сервис, запрещённый на территории РФ)

Решение

SELECT AVG(cs.num)

FROM(

SELECT COUNT(edu.instituition) as num

FROM people as p

INNER JOIN company as comp ON comp.id=p.company\_id

INNER JOIN education as edu on p.id=edu.person\_id

WHERE comp.name='Facebook'

GROUP BY p.id

) AS cs

19.Составьте таблицу из полей:

* name\_of\_fund — название фонда;
* name\_of\_company — название компании;
* amount — сумма инвестиций, которую привлекла компания в раунде.

В таблицу войдут данные о компаниях, в истории которых было больше шести важных этапов, а раунды финансирования проходили с 2012 по 2013 год включительно.

Решение:

WITH

fr AS (SELECT \*

FROM funding\_round

WHERE EXTRACT (YEAR FROM funded\_at) BETWEEN 2012 AND 2013) -- раунды финансирования 2012-13

SELECT f.name AS name\_of\_fund,

c.name AS name\_of\_company,

fr.raised\_amount AS amount

FROM investment AS i

INNER JOIN company AS c ON c.id = i.company\_id

INNER JOIN fund AS f ON i.fund\_id = f.id

INNER JOIN fr ON i.funding\_round\_id = fr.id

WHERE c.milestones > 6

GROUP BY f.name , c.name , fr.raised\_amount

20.Выгрузите таблицу, в которой будут такие поля:

* название компании-покупателя;
* сумма сделки;
* название компании, которую купили;
* сумма инвестиций, вложенных в купленную компанию;
* доля, которая отображает, во сколько раз сумма покупки превысила сумму вложенных в компанию инвестиций, округлённая до ближайшего целого числа.

Не учитывайте те сделки, в которых сумма покупки равна нулю. Если сумма инвестиций в компанию равна нулю, исключите такую компанию из таблицы.

Отсортируйте таблицу по сумме сделки от большей к меньшей, а затем по названию купленной компании в лексикографическом порядке. Ограничьте таблицу первыми десятью записями.

Решение:

SELECT c\_ing.name,

a.price\_amount,

c\_ed.name,

c\_ed.funding\_total,

ROUND(a.price\_amount / c\_ed.funding\_total)

FROM acquisition AS a

JOIN company AS c\_ing ON a.acquiring\_company\_id = c\_ing.id

JOIN company AS c\_ed ON a.acquired\_company\_id = c\_ed.id

WHERE a.price\_amount != 0

AND c\_ed.funding\_total != 0

ORDER BY a.price\_amount DESC, c\_ed.name

LIMIT 10;

21.Выгрузите таблицу, в которую войдут названия компаний из категории social, получившие финансирование с 2010 по 2013 год включительно. Проверьте, что сумма инвестиций не равна нулю. Выведите также номер месяца, в котором проходил раунд финансирования.

Решение:

SELECT cs.name AS company\_name,

EXTRACT(MONTH FROM CAST (fr.funded\_at AS date)) AS month

FROM (SELECT id,

name

FROM company

WHERE category\_code = 'social') AS cs

LEFT OUTER JOIN funding\_round AS fr ON cs.id = fr.company\_id

WHERE EXTRACT (YEAR FROM CAST (fr.funded\_at AS date)) BETWEEN 2010 AND 2013

22.Отберите данные по месяцам с 2010 по 2013 год, когда проходили инвестиционные раунды. Сгруппируйте данные по номеру месяца и получите таблицу, в которой будут поля:

* номер месяца, в котором проходили раунды;
* количество уникальных названий фондов из США, которые инвестировали в этом месяце;
* количество компаний, купленных за этот месяц;
* общая сумма сделок по покупкам в этом месяце.

Решение:

with

ac\_bought as

(select

extract(month from ac.acquired\_at) as \_month,

count(ac.acquired\_company\_id) as \_count\_acquired,

sum(ac.price\_amount) as \_price\_amount

from acquisition ac

where extract (year from cast(ac.acquired\_at as date)) between '2010' and '2013'

group by \_month),

ac\_fund\_names as

(select

extract(month from fr.funded\_at) as \_month,

count(distinct fu.name) as \_count

from investment i

inner join fund fu on i.fund\_id = fu.id

inner join funding\_round fr on i.funding\_round\_id = fr.id

where fu.country\_code = 'USA' and extract (year from cast(fr.funded\_at as date)) between '2010' and '2013'

group by \_month)

select

b.\_month as "month",

f.\_count as count\_usa\_fund\_names,

b.\_count\_acquired as count\_acquired,

b.\_price\_amount as sum\_price\_amount

from ac\_bought b

inner join ac\_fund\_names f on f.\_month = b.\_month

23.Составьте сводную таблицу и выведите среднюю сумму инвестиций для стран, в которых есть стартапы, зарегистрированные в 2011, 2012 и 2013 годах. Данные за каждый год должны быть в отдельном поле. Отсортируйте таблицу по среднему значению инвестиций за 2011 год от большего к меньшему.

Решение:

with

a as (select country\_code, avg (funding\_total) as i\_2011

from company as c

where extract (year from c.founded\_at)=2011

group by country\_code),

b as (select country\_code, avg (funding\_total) as i\_2012

from company as c

where extract (year from c.founded\_at)=2012

group by country\_code),

c as (select country\_code, avg (funding\_total) as i\_2013

from company as c

where extract (year from c.founded\_at)=2013

group by country\_code)

SELECT a.country\_code,

i\_2011,

i\_2012,

i\_2013

FROM a

INNER JOIN b ON a.country\_code = b.country\_code

INNER JOIN c ON a.country\_code = c.country\_code

ORDER BY i\_2011 DESC