

| 1. Отобразите все записи из таблицы company по компаниям, которые закрылись. | | | | | | | | |
|---|-----------|-----------|-------------|--|---------|--|-----------|--|
| <pre>SELECT * FROM company WHERE status LIKE ('closed');</pre> | | | | | | | | |
| 2. Отобразите количество привлечённых средств для новостных компаний США. Используйте данные из таблицы company. Отсортируйте таблицу по убыванию значений в поле funding_total. | | | | | | | | |
| <pre>SELECT --name, SUM(funding_total) sum_total FROM company WHERE (category_code = 'news') AND (country_code = 'USA') GROUP BY name ORDER BY sum_total DESC;</pre> | | | | | | | | |
| Результат | | | | | | | | |
| <table><thead><tr><th></th><th>sum_total</th></tr></thead><tbody><tr><td>6.22553e+08</td><td></td></tr><tr><td>2.5e+08</td><td></td></tr><tr><td>1.605e+08</td><td></td></tr></tbody></table> | | sum_total | 6.22553e+08 | | 2.5e+08 | | 1.605e+08 | |
| | sum_total | | | | | | | |
| 6.22553e+08 | | | | | | | | |
| 2.5e+08 | | | | | | | | |
| 1.605e+08 | | | | | | | | |
| 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. | | | | | | | | |
| <pre>SELECT SUM(price_amount) FROM acquisition WHERE (term_code = 'cash') AND EXTRACT(YEAR FROM CAST(acquired_at AS date)) IN (2011, 2012, 2013)</pre> | | | | | | | | |
| Результат | | | | | | | | |
| <table><thead><tr><th></th><th>sum</th></tr></thead><tbody><tr><td>1.37762e+11</td><td></td></tr></tbody></table> | | sum | 1.37762e+11 | | | | | |
| | sum | | | | | | | |
| 1.37762e+11 | | | | | | | | |
| 4. Отобразите имя, фамилию и названия аккаунтов людей в твиттере, у которых названия аккаунтов начинаются на 'Silver'. | | | | | | | | |
| <pre>SELECT first_name, last_name, twitter_username FROM people WHERE (twitter_username LIKE ('Silver%'))</pre> | | | | | | | | |

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

```
SELECT *
FROM people
WHERE (twitter_username LIKE '%money%') AND (last_name LIKE 'K%')
```

Результат

| id | first_name | last_name | company_id | twitter_username | created_at |
|-------|------------|-----------|------------|------------------|---------------------|
| 63081 | Gregory | Kim | | gmoney75 | 2010-07-13 03:46:28 |

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.

```
SELECT
    country_code,
    SUM(funding_total) AS sum_total
FROM company
GROUP BY country_code
ORDER BY sum_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.

```
SELECT
    CAST(funded_at AS date) AS f_date,
    MIN(raised_amount) AS min_amount,
    MAX(raised_amount) AS max_amount
FROM funding_round
GROUP BY f_date
HAVING (MIN(raised_amount) != 0) AND (MIN(raised_amount) !=
MAX(raised_amount))
```

Результат

| f_date | min_amount | max_amount |
|------------|-------------|-------------|
| 2012-08-22 | 40000 | 7.5e+07 |
| 2010-07-25 | 3.27825e+06 | 9e+06 |
| 2002-03-01 | 2.84418e+06 | 8.95915e+06 |
| 2010-10-11 | 28000 | 2e+08 |

| | | |
|------------|----------|----------|
| 2007-01-18 | 5.5e+06 | 2.3e+07 |
| 2007-02-27 | 1.29e+06 | 3.6e+07 |
| 2006-01-05 | 8.9e+06 | 2.65e+07 |

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.

```
SELECT *,
CASE
  WHEN invested_companies >= 100 THEN 'high_activity'
  WHEN (invested_companies >=20) AND (invested_companies < 100) THEN
'middle_activity'
  WHEN (invested_companies <20) THEN 'low_activity'
END
FROM fund
```

Результат

| id | name | founded_at | domain | twitter_username | country_code |
|-------|-------------------|------------|--------------|------------------|--------------|
| 13131 | | | | | |
| 1 | Greylock Partners | 1965-01-01 | greylock.com | greylockvc | USA |
| 10 | | | | | |

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.

```
SELECT
ROUND(AVG(investment_rounds)) AS activity_rounds,
CASE
  WHEN invested_companies>=100 THEN 'high_activity'
  WHEN invested_companies>=20 THEN 'middle_activity'
  ELSE 'low_activity'
END AS activity
FROM fund
```

| | | | |
|---|-------------------|-------------------|-------------------|
| GROUP BY activity ORDER BY activity_rounds; | | | |
| Результат | | | |
| | activity_rounds | | activity |
| 2 | | low_activity | |
| 51 | | middle_activity | |
| 252 | | high_activity | |
| <p>10.</p> <p>Проанализируйте, в каких странах находятся фонды, которые чаще всего инвестируют в стартапы.</p> <p>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.</p> <p>Выгрузите десять самых активных стран-инвесторов: отсортируйте таблицу по среднему количеству компаний от большего к меньшему. Затем добавьте сортировку по коду страны в лексикографическом порядке.</p> | | | |
| <p>Это решение выдает ошибку. В таблице есть поле jvvested_companies в котором хранится количество компаний в которые инвестировали</p> | | | |
| <p>WITH</p> <p>f_fund AS (SELECT *</p> <p>FROM fund</p> <p>--WHERE EXTRACT(YEAR FROM CAST(founded_at AS DATE)) IN (2010, 2011, 2012)),</p> <p>WHERE CAST(founded_at AS DATE) BETWEEN '2010-01-01' AND '2012-12-31'),</p> <p>t_1 AS (SELECT</p> <p>f.country_code AS contry_fund_from,</p> <p>f.id AS fund_id,</p> <p>COUNT(i.company_id) AS count_company</p> <p>FROM f_fund AS f</p> <p>LEFT OUTER JOIN investment AS I ON i.fund_id=f.id</p> <p>GROUP BY f.country_code, f.id)</p> <p>SELECT</p> <p>t_1.contry_fund_from AS country,</p> <p>MIN(t_1.count_company) AS min_count_company,</p> <p>MAX(t_1.count_company) AS max_count_company,</p> <p>AVG(t_1.count_company) AS avg_count_company</p> <p>FROM t_1</p> <p>GROUP BY t_1.contry_fund_from</p> <p>HAVING MIN(t_1.count_company) != 0</p> <p>ORDER BY avg_count_company DESC, country ASC</p> <p>LIMIT 10;</p> | | | |
| Результат | | | |
| country | min_count_company | max_count_company | avg_count_company |
| BGR | 26 | 37 | 31.5 |
| CHL | 29 | 29 | 29 |

| | | | |
|---|--------------------------|--------------------------|---------|
| UKR | 8 | 11 | 9.5 |
| LTU | 8 | 8 | 8 |
| IRL | 4 | 6 | 5 |
| HKG | 3 | 4 | 3.33333 |
| JPN | 1 | 6 | 3 |
| KEN | 3 | 3 | 3 |
| LBN | 3 | 3 | 3 |
| MUS | 3 | 3 | 3 |
| Этот вариант правильный | | | |
| WITH f_fund AS (SELECT * FROM fund WHERE EXTRACT(YEAR FROM CAST(founded_at AS DATE)) IN (2010, 2011, 2012)) SELECT f_fund.country_code AS country, MIN(f_fund.invested_companies) AS min_count_company, MAX(f_fund.invested_companies) AS max_count_company, AVG(f_fund.invested_companies) AS avg_count_company FROM f_fund GROUP BY f_fund.country_code HAVING(MIN(f_fund.invested_companies)) != 0 ORDER BY avg_count_company DESC, country ASC LIMIT 10; | | | |
| Результат | | | |
| country | min_count_company | max_count_company | |
| BGR | 25 | 35 | |
| CHL | 29 | 29 | |
| UKR | 8 | 10 | |
| LTU | 5 | 5 | |
| IRL | 4 | 5 | |
| KEN | 3 | 3 | |
| LBN | 3 | 3 | |
| MUS | 3 | 3 | |

| | | |
|---|--------------|----------------------------------|
| JPN | 1 | 6 |
| 11. Отобразите имя и фамилию всех сотрудников стартапов. Добавьте поле с названием учебного заведения, которое окончил сотрудник, если эта информация известна. | | |
| SELECT p.first_name, p.last_name, e.institution FROM people AS p LEFT OUTER JOIN education AS e ON p.id=e.person_id | | |
| Результат | | |
| first_name | last_name | institution |
| John | Green | Washington University, St. Louis |
| John | Green | Boston University |
| David | Peters | Rice University |
| Dan | Birdwhistell | University of Cambridge |
| Gal | Cohen | Tel Aviv University |
| Дополнительная задача для самой себя. Стало интересно какое максимальное количество ученых степеней у сотрудников компаний со стартапами в которые инвестируют | | |
| SELECT p.id, COUNT(e.degree_type) FROM people AS p LEFT OUTER JOIN education AS e ON p.id=e.person_id GROUP BY p.id ORDER BY COUNT(e.degree_type) DESC LIMIT 10; | | |
| Результат | | |
| | id | |
| 183805 | | 9 |
| 93386 | | 9 |
| 65231 | | 8 |
| 216686 | | 7 |
| 143324 | | 7 |
| 234052 | | 7 |
| 143330 | | 7 |

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.

```
SELECT
  c.name AS name_company,
  COUNT(DISTINCT e.institution) AS count_degrees
FROM company AS c
JOIN people AS p ON c.id=p.company_id
JOIN education AS e ON e.person_id = p.id
GROUP BY c.name
ORDER BY count_degrees DESC
LIMIT 5;
```

13.

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

```
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.

```
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.

```
SELECT
  distinct p.id,
  e.institution
FROM people AS p
JOIN education AS e ON e.person_id=p.id
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))) | |
| 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. | |
| SELECT distinct p.id, COUNT(e.institution) FROM people AS p JOIN education AS e ON e.person_id=p.id 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 p.id | |
| 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. | |
| WITH t_1 AS (SELECT distinct p.id, COUNT(e.institution) AS count_inst FROM people AS p JOIN education AS e ON e.person_id=p.id 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 p.id) SELECT AVG(t_1.count_inst) FROM t_1 | |
| Результат | avg |
| 1.41509 | |
| 18. Напишите похожий запрос: выведите среднее число учебных заведений (всех, не только уникальных), которые окончили сотрудники Facebook*. *(сервис, запрещённый на территории РФ) | |
| WITH | |

| | | |
|---|--------------|-----------------|
| <pre> t_1 AS (SELECT id AS company_id FROM company WHERE name LIKE 'Facebook'), t_2 AS (SELECT id FROM people AS p JOIN t_1 ON p.company_id = t_1.company_id), t_3 AS (SELECT t_2.id, COUNT(institution) AS count_inst FROM education AS e JOIN t_2 ON t_2.id = e.person_id GROUP BY t_2.id) SELECT AVG(t_3.count_inst) from t_3 </pre> | | |
| Результат | | |
| | | avg |
| 1.51111 | | |
| <p>19.</p> <p>Составьте таблицу из полей:</p> <ul style="list-style-type: none"> • name_of_fund — название фонда; • name_of_company — название компании; • amount — сумма инвестиций, которую привлекла компания в раунде. <p>В таблицу войдут данные о компаниях, в истории которых было больше шести важных этапов, а раунды финансирования проходили с 2012 по 2013 год включительно.</p> | | |
| <pre> SELECT f.name AS name_of_fund, c.name AS name_of_company, fr.raised_amount FROM investment AS i RIGHT OUTER JOIN company AS c ON i.company_id=c.id RIGHT OUTER JOIN fund AS f ON i.fund_id=f.id JOIN funding_round AS fr ON i.funding_round_id = fr.id WHERE (c.milestones > 6) AND(CAST(fr.funded_at AS DATE) BETWEEN '2012-01-01' AND '2013-12-31') </pre> | | |
| Результат | | |
| | name_of_fund | name_of_company |
| Advance Publication | | Gigya |
| Mayfield Fund | | Gigya |

| | |
|---|-------|
| Benchmark | Gigya |
| DAG Ventures | Gigya |
| Mitsui Global Investment | OpenX |
| Accel Partners | OpenX |
| Presidio Ventures | OpenX |
| Index Ventures | OpenX |
| Samsung Ventures | OpenX |
| SAP Ventures | OpenX |
| Mayfield Fund | Gigya |
| Greenspring Associates | Gigya |
| Benchmark | Gigya |
| <p>20. Выгрузите таблицу, в которой будут такие поля:</p> <ul style="list-style-type: none"> • 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. <p>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.</p> | |
| <p>WITH buyers AS (SELECT c.name AS byer_company, a.price_amount AS amount, a.id AS id FROM company AS c JOIN acquisition AS a ON c.id = a.acquiring_company_id WHERE a.price_amount != 0),</p> <p>sales AS (SELECT c.name AS saled_company, c.funding_total AS total,</p> | |

```

a.id AS id
FROM company AS c
JOIN acquisition AS a ON c.id = a.acquired_company_id
WHERE c.funding_total != 0)

```

```

SELECT
  DISTINCT buyers.byer_company,
  buyers.amount,
  sales.saled_company,
  sales.total,
  ROUND(buyers.amount/sales.total) AS delta
FROM buyers JOIN sales ON buyers.id = sales.id
ORDER BY buyers.amount DESC, sales.saled_company
LIMIT 10;

```

Результат

| byer_company | amount | saled_company |
|------------------------|----------|---|
| Microsoft | 8.5e+09 | Skype |
| Scout Labs | 4.9e+09 | Varian Semiconductor Equipment Associates |
| Broadcom | 3.7e+09 | Aeluros |
| Broadcom | 3.7e+09 | NetLogic Microsystems |
| Level 3 Communications | 3e+09 | Global Crossing |
| Yahoo! | 2.87e+09 | GeoCities |
| eBay | 2.6e+09 | Skype |
| Salesforce | 2.5e+09 | ExactTarget |
| Johnson & Johnson | 2.3e+09 | Crucell |
| IAC | 1.85e+09 | Ask.com |

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.

```

SELECT
  c.name,
  EXTRACT(MONTH FROM CAST(fr.funded_at AS DATE))
FROM company AS c
LEFT OUTER JOIN funding_round AS fr ON fr.company_id=c.id
WHERE EXTRACT(YEAR FROM CAST(fr.funded_at AS DATE)) IN (2010,2011,2012,2013)
  AND (c.category_code='social')
  AND(fr.raised_amount !=0)

```

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.

WITH

```
inv_2011 AS (SELECT
    c.country_code AS country,
    AVG(c.funding_total) AS avg_total_2011
FROM company AS c
WHERE EXTRACT(YEAR FROM c.founded_at) = 2011
GROUP BY c.country_code), -- сформируйте первую временную таблицу
```

```
inv_2012 AS (SELECT
    c.country_code AS country,
    AVG(c.funding_total) AS avg_total_2012
FROM company AS c
WHERE EXTRACT(YEAR FROM c.founded_at) = 2012
GROUP BY c.country_code),
```

```
inv_2013 AS (SELECT
    c.country_code AS country,
    AVG(c.funding_total) AS avg_total_2013
FROM company AS c
WHERE EXTRACT(YEAR FROM c.founded_at) = 2013
GROUP BY c.country_code)
```

```
SELECT inv_2011.country,
    inv_2011.avg_total_2011,
    inv_2012.avg_total_2012,
    inv_2013.avg_total_2013
```

```
FROM inv_2011
```

```
INNER JOIN inv_2012 ON inv_2011.country=inv_2012.country
```

```
INNER JOIN inv_2013 ON inv_2012.country=inv_2013.country
```

```
ORDER BY inv_2011.avg_total_2011 DESC
```