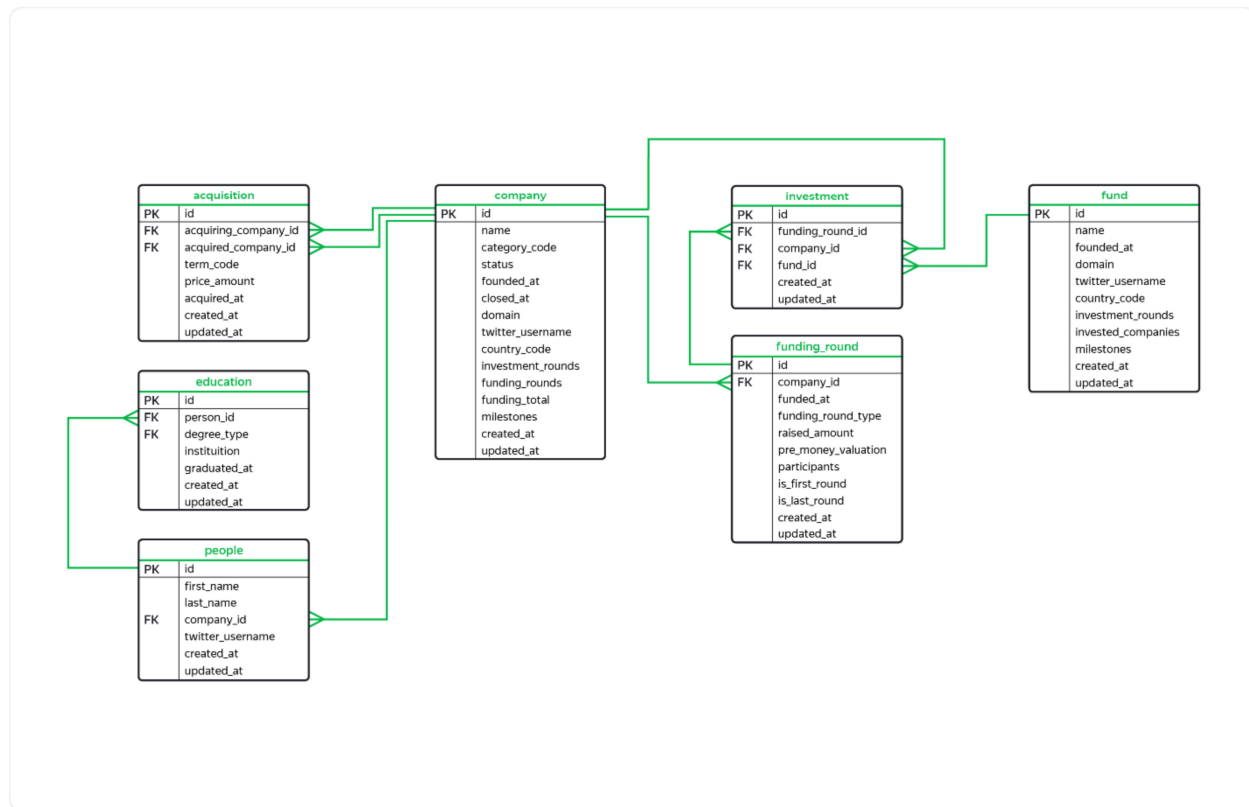


# Учебный проект №6 (базовый SQL)

ER-диаграмма БД



Данные в таблицах:

- acquisition содержит информацию о покупках одних компаний другими.
- company содержит информацию о компаниях-стартапах.
- education хранит информацию об уровне образования сотрудников компаний.
- fund хранит информацию о венчурных фондах.
- funding\_round содержит информацию о раундах инвестиций.
- investment содержит информацию об инвестициях венчурных фондов в компании-стартапы.
- people содержит информацию о сотрудниках компаний-стартапов

### Задача 1/23

1. Посчитайте, сколько компаний закрылось.	<pre>select COUNT(status) from company where status = 'closed';</pre>
---	---

### Задача 2/23

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 funding_total from company where country_code = 'USA' AND category_code = 'news' order by funding_total desc;</pre>
---	---

### Задача 3/23

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 CAST(acquired_at as date) BETWEEN '01.01.2011' AND '31.12.2013';</pre>
--	--

### Задача 4/23

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>
---	--

### Задача 5/23

5. Выведите на экран всю информацию о людях, у которых названия аккаунтов в твиттере содержат подстроку 'money', а фамилия начинается на 'K'.	<pre>SELECT * FROM people WHERE twitter_username LIKE '%money%' AND last_name LIKE 'K%';</pre>
--	--

### Задача 6/23

6. Для каждой страны отобразите общую сумму привлечённых инвестиций, которые получили	<pre>SELECT country_code, SUM(funding_total) FROM company GROUP BY country_code</pre>
--	---

компании, зарегистрированные в этой стране. Страну, в которой зарегистрирована компания, можно определить по коду страны. Отсортируйте данные по убыванию суммы.	ORDER BY SUM(funding_total) DESC;
--	-----------------------------------

### Задача 7/23

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.	<pre> SELECT t1.fund_date,        t1.min_amount,        t1.max_amount FROM (   SELECT CAST(funded_at AS DATE) AS fund_date,        MIN(raised_amount) AS min_amount,        MAX(raised_amount) AS max_amount FROM funding_round GROUP BY fund_date) AS t1 WHERE t1.min_amount != 0 AND t1.min_amount != t1.max_amount;</pre>
---	--

### Задача 8/23

<p>8.</p> <p>Создайте поле с категориями:</p> <ul style="list-style-type: none"> <li>Для фондов, которые инвестируют в 100 и более компаний, назначьте категорию <code>high_activity</code>.</li> <li>Для фондов, которые инвестируют в 20 и более компаний до 100, назначьте категорию <code>middle_activity</code>.</li> <li>Если количество инвестируемых компаний фонда не достигает 20, назначьте категорию <code>low_activity</code>.</li> </ul> <p>Отобразите все поля таблицы <code>fund</code> и новое поле с категориями.</p>	<pre> SELECT *, CASE   WHEN invested_companies &gt;= 100 THEN 'high_activity'   WHEN invested_companies &gt;= 20 AND invested_companies &lt; 100 THEN 'middle_activity'   WHEN invested_companies &lt; 20 THEN 'low_activity' END AS cat FROM fund;</pre>
---	---

### Задача 9/23

<p>9.</p> <p>Для каждой из категорий, назначенных в предыдущем задании, посчитайте округлённое до ближайшего целого числа среднее количество инвестиционных раундов,</p>	<pre> SELECT CASE   WHEN invested_companies &gt;= 100 THEN 'high_activity'   WHEN invested_companies &gt;= 20 THEN 'middle_activity'   ELSE 'low_activity'</pre>
--	--

в которых фонд принимал участие. Выведите на экран категории и среднее число инвестиционных раундов. Отсортируйте таблицу по возрастанию среднего.	END AS activity, ROUND(AVG(investment_rounds)) AS avg_rounds FROM fund GROUP BY activity ORDER BY avg_rounds;
--	---

### Задача 10/23

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.	<pre> SELECT country_code,        MIN(invested_companies) AS min_inv_comp,        MAX(invested_companies) AS max_inv_comp,        AVG(invested_companies) AS avg_inv_comp FROM fund WHERE CAST(founded_at AS DATE) BETWEEN '01.01.2010' AND '31.12.2012' GROUP BY country_code HAVING MIN(invested_companies) != 0 ORDER BY avg_inv_comp DESC, country_code LIMIT 10; </pre>
---	--

### Задача 11/23

11. Отобразите имя и фамилию всех сотрудников стартапов. Добавьте поле с названием учебного заведения, которое окончил сотрудник, если эта информация известна.	<pre> SELECT p.first_name,        p.last_name,        e.institution FROM people AS p LEFT JOIN education AS e ON p.id = e.person_id; </pre>
--	---

### Задача 12/23

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.	<pre> /*SELECT c.name,        COUNT(DISTINCT e.institution) as count_inst FROM company AS c LEFT JOIN people AS p ON c.id = p.company_id        LEFT JOIN education as e ON p.id = e.person_id GROUP BY c.name ORDER BY count_inst DESC LIMIT 5;*/ /*WITH i AS (SELECT p.company_id as comp_id,        COUNT(DISTINCT e.institution) AS count_inst </pre>
--	---

	<pre> FROM people AS p LEFT JOIN education as e ON p.id = e.person_id GROUP BY p.company_id) SELECT c.name,        i.count_inst FROM company AS c LEFT JOIN i ON c.id = comp_id ORDER BY i.count_inst DESC LIMIT 5;*/ SELECT c.name,        i.count_inst FROM company AS c JOIN               (SELECT p.company_id as comp_id, COUNT(DISTINCT e.institution) AS count_inst               FROM people AS p LEFT JOIN education as e ON p.id = e.person_id               GROUP BY p.company_id) AS i               ON c.id = i.comp_id ORDER BY i.count_inst DESC LIMIT 5; </pre>
--	---

### Задача 13/23

<p>13. Составьте список с уникальными названиями закрытых компаний, для которых первый раунд финансирования оказался последним.</p>	<pre> SELECT name FROM company WHERE id IN       (SELECT DISTINCT company_id       FROM funding_round       WHERE is_first_round = 1 AND is_last_round = 1) AND status LIKE 'closed'; </pre>
---	--

### Задача 14/23

<p>14. Составьте список уникальных номеров сотрудников, которые работают в компаниях, отобранных в предыдущем задании.</p>	<pre> WITH c AS (SELECT id AS id_comp FROM company WHERE id IN       (SELECT DISTINCT company_id       FROM funding_round       WHERE is_first_round = 1 AND is_last_round = 1) AND status LIKE 'closed') SELECT DISTINCT id FROM people WHERE company_id IN (SELECT *                      FROM c) </pre>
--	--

### Задача 15/23

15. Составьте таблицу, куда войдут уникальные пары с номерами сотрудников из предыдущей задачи и учебным заведением, которое окончил сотрудник.	WITH c AS (SELECT id AS id_comp FROM company WHERE id IN (SELECT DISTINCT company_id FROM funding_round WHERE is_first_round = 1 AND is_last_round = 1) AND status LIKE 'closed') SELECT DISTINCT p.id, e.institution FROM people AS p JOIN education AS e ON p.id = e.person_id WHERE company_id IN (SELECT * FROM c);
--	---

### Задача 16/23

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.	WITH c AS (SELECT id AS id_comp FROM company WHERE id IN (SELECT DISTINCT company_id FROM funding_round WHERE is_first_round = 1 AND is_last_round = 1) AND status LIKE 'closed') SELECT DISTINCT p.id AS person, COUNT(e.institution) FROM people AS p JOIN education AS e ON p.id = e.person_id WHERE company_id IN (SELECT * FROM c) GROUP BY person;
---	---

### Задача 17/23

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 c AS (SELECT id AS id_comp FROM company WHERE id IN (SELECT DISTINCT company_id FROM funding_round WHERE is_first_round = 1 AND is_last_round = 1) AND status LIKE 'closed')  SELECT AVG(pi.num_inst) FROM ( SELECT DISTINCT p.id AS person, COUNT(e.institution) AS num_inst FROM people AS p JOIN education AS e ON p.id = e.person_id WHERE company_id IN (SELECT * FROM c) GROUP BY person ) AS pi ;
--	---

### Задача 18/23

18. Напишите похожий запрос: выведите среднее число учебных заведений (всех, не только уникальных), которые окончили сотрудники Facebook*. *(сервис, запрещённый на территории РФ)	WITH c AS (SELECT id AS id_comp FROM company WHERE name LIKE 'Facebook')  SELECT AVG(pi.num_inst) FROM ( SELECT DISTINCT p.id AS person, COUNT(e.institution) AS num_inst FROM people AS p JOIN education AS e ON p.id = e.person_id WHERE company_id IN (SELECT * FROM c) GROUP BY person ) AS pi ;
---	--

### Задача 19/23

19. Составьте таблицу из полей: <ul style="list-style-type: none"><li>name_of_fund — название фонда;</li><li>name_of_company — название компании;</li><li>amount — сумма инвестиций, которую привлекла компания в раунде.</li></ul> В таблицу войдут данные о компаниях, в истории которых было больше шести важных этапов, а раунды финансирования проходили с 2012 по 2013 год включительно.	SELECT f.name AS name_of_fund, c.name AS name_of_company, fr.raised_amount AS amount FROM investment as i JOIN company AS c ON i.company_id = c.id 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(funded_at AS DATE) BETWEEN '01.01.2012' AND '31.12.2013' ;
---	--

### Задача 20/23

20. Выгрузите таблицу, в которой будут такие поля: <ul style="list-style-type: none"><li>название компании-покупателя;</li><li>сумма сделки;</li><li>название компании, которую купили;</li><li>сумма инвестиций, вложенных в купленную</li></ul>	SELECT c.name, d.name, a.price_amount, d.funding_total, ROUND(a.price_amount / d.funding_total) AS leverage FROM acquisition AS a LEFT JOIN company AS c ON a.acquiring_company_id = c.id LEFT JOIN company AS d ON a.acquired_company_id = d.id WHERE a.price_amount != 0 AND d.funding_total != 0
---	--

<p>компанию;</p> <ul style="list-style-type: none"> <li>• доля, которая отображает, во сколько раз сумма покупки превысила сумму вложенных в компанию инвестиций, округлённая до ближайшего целого числа.</li> </ul> <p>Не учитывайте те сделки, в которых сумма покупки равна нулю. Если сумма инвестиций в компанию равна нулю, исключите такую компанию из таблицы.</p> <p>Отсортируйте таблицу по сумме сделки от большей к меньшей, а затем по названию купленной компании в лексикографическом порядке. Ограничьте таблицу первыми десятью записями.</p>	<pre>ORDER BY a.price_amount DESC, d.name LIMIT 10 ;</pre>
--	--

### Задача 21/23

<p>21.</p> <p>Выгрузите таблицу, в которую войдут названия компаний из категории <code>social</code>, получившие финансирование с 2010 по 2013 год включительно. Проверьте, что сумма инвестиций не равна нулю. Выведите также номер месяца, в котором проходил раунд финансирования.</p>	<pre>WITH fr AS (SELECT company_id,       EXTRACT(MONTH FROM CAST(funded_at AS DATE)) AS month       FROM funding_round       WHERE CAST(funded_at AS DATE) BETWEEN '01.01.2010' AND '31.12.2013'       AND raised_amount != 0       ) SELECT name,       fr.month FROM company AS c JOIN fr ON c.id = fr.company_id WHERE category_code LIKE 'social'</pre>
---	--

### Задача 22/23

<p>22.</p> <p>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:</p> <ul style="list-style-type: none"> <li>• номер месяца, в котором проходили раунды;</li> <li>• количество уникальных названий фондов из США,</li> </ul>	<pre>WITH -- round ids with USA funds ids iu AS (SELECT i.funding_round_id AS funding_round_id,       i.fund_id       FROM investment AS i       WHERE fund_id IN (SELECT id       FROM fund       WHERE country_code LIKE 'USA')       ), -- deals with companies by months dc AS (SELECT COUNT(acquired_company_id) AS company_count,       SUM(price_amount) AS deals_total,</pre>
---	---



<p>которые инвестировали в этом месяце;</p> <ul style="list-style-type: none"> <li>• количество компаний, купленных за этот месяц;</li> <li>• общая сумма сделок по покупкам в этом месяце.</li> </ul>	<pre> EXTRACT(MONTH FROM CAST(acquired_at AS DATE)) AS month FROM acquisition WHERE CAST(acquired_at AS DATE) BETWEEN '01.01.2010' AND '31.12.2013' GROUP BY month )  SELECT t1.month,        t1.unic_funds,        dc.company_count,        dc.deals_total -- основную таблицу пришлось вынести в подзапрос, -- иначе выдавались разные ошибки об отсутствии -- столбцов FROM (SELECT EXTRACT(MONTH FROM CAST(fr.funded_at AS DATE)) AS month,        COUNT(DISTINCT iu.fund_id) AS unic_funds FROM funding_round AS fr LEFT JOIN iu ON fr.id = iu.funding_round_id WHERE CAST(fr.funded_at AS DATE) BETWEEN '01.01.2010' AND '31.12.2013' GROUP BY month) AS t1 -- склеиваем основную таблицу, основанную на -- funding_rounds и funds -- с таблицей про компании, с которой связи нет, но -- есть идентичная разбивка на месяцы FULL OUTER JOIN dc ON t1.month = dc.month ; </pre>
--	---

### Задача 23/23

<p>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.</p>	<pre> WITH list_2011 AS (SELECT country_code AS country_list,                     AVG(funding_total) AS avg_invest_2011 FROM company WHERE founded_at BETWEEN '01.01.2011' AND '31.12.2011' GROUP BY country_code), list_2012 AS (SELECT country_code,                     AVG(funding_total) AS avg_invest_2012 FROM company WHERE founded_at BETWEEN '01.01.2012' AND '31.12.2012' GROUP BY country_code), list_2013 AS (SELECT country_code,                     AVG(funding_total) AS avg_invest_2013 FROM company WHERE founded_at BETWEEN '01.01.2013' AND '31.12.2013' GROUP BY country_code) SELECT country_list,        avg_invest_2011, </pre>
--	--

	<pre>avg_invest_2012, avg_invest_2013 FROM list_2011 JOIN list_2012 ON country_list = list_2012.country_code       JOIN list_2013 ON country_list = list_2013.country_code ORDER BY avg_invest_2011 DESC ;</pre>
--	--