# Запросы с симметричным внутренним соединением с условием на внешний ключ

Информация о **тарифах**, объединённых с услугами. Отбор по услугам.

SELECT t.name, s.name, c.name, t.cost FROM tariffs AS t

INNER JOIN services AS s ON s.id = t.id\_service

LEFT OUTER JOIN charge\_types AS c ON c.id = t.id\_charge\_type

WHERE s.name = '%s' ORDER BY t.name, s.name, c.name

Информация о **контрактах**, объединённых с клиентами. Отбор по клиентам.

SELECT c.name, cl.name, c.preparation\_date, s.name FROM contracts AS c

INNER JOIN clients AS cl ON cl.id = c.id\_client

LEFT OUTER JOIN contract\_statuses AS s ON s.id = c.id\_status

WHERE cl.name = '%s' ORDER BY c.name, cl.name, s.name

# Запросы с симметричным внутренним соединением с условием на дату

Информация о **финанс**. **операциях**, объединённых с типами транзакций. Отбор по дате операции.

SELECT t.name, f.sum, f.operation\_date, f.comment FROM financial\_operations AS f

INNER JOIN transaction\_types AS t ON t.id = f.id\_transaction\_type

WHERE f.operation\_date = '%s' ORDER BY f.operation\_date, t.name

Информация о **контрактах**, объединённых со статусами. Отбор по дате завершения.

SELECT c.name, cs.name, c.start\_date, c.end\_date FROM contracts AS c

INNER JOIN contract\_statuses AS cs ON cs.id = c.id\_status

WHERE c.end\_date = '%s' ORDER BY c.name, cs.name

# Запросы с симметричным внутренним соединением без условия

Информация о **клиентах**, объединённых с городами.

SELECT cl.name, c.name, cl.address, cl.contact\_information FROM clients AS cl

INNER JOIN client\_cities AS c ON c.id = cl.id\_city ORDER BY cl.name, c.name

Информация о **тарифах**, объединённых с типами оплаты.

SELECT t.name, s.name, c.name, t.cost FROM tariffs AS t

INNER JOIN charge\_types AS c ON c.id = t.id\_charge\_type

LEFT OUTER JOIN services AS s ON s.id = t.id\_service ORDER BY t.name, c.name, s.name

Информация о **частях** **контрактов**, объединённых с контрактами и тарифами.

SELECT c.name, t.name FROM contract\_parts AS cp

INNER JOIN contracts AS c ON c.id = cp.id\_contract

INNER JOIN tariffs AS t ON t.id = cp.id\_tariff ORDER BY c.name, t.name

# Запрос с левым внешним соединением

Информация о **финанс**. **операциях** без указанных частей контрактов.

SELECT t.name, f.sum, f.comment, c.name FROM financial\_operations AS f

LEFT OUTER JOIN transaction\_types AS t ON t.id = f.id\_transaction\_type

LEFT OUTER JOIN contract\_parts AS cp ON cp.id = f.id\_contract\_part

LEFT OUTER JOIN contracts AS c ON c.id = cp.id\_contract

WHERE f.id\_contract\_part ISNULL ORDER BY f.operation\_date, c.name, t.name

# Запрос с правым внешним соединением

Информация о **клиентах**. Отобразить счета для указанного клиента.

SELECT c.name, c.address, c.contact\_information, b.balance FROM bills AS b

RIGHT OUTER JOIN clients AS c ON c.id = b.id\_client AND c.name = '%s'

ORDER BY c.name, b.balance

# Запрос на запросе

Информация о **финанс**. **операциях**, полученная на основе запроса с правым внешним соединением.

SELECT DISTINCT cl\_1, cl\_3 FROM (SELECT c.name AS cl\_1, c.address, c.contact\_information AS cl\_3, b.balance

FROM bills AS b RIGHT OUTER JOIN clients AS c ON c.id = b.id\_client) AS cl

ORDER BY 1, 2

# Итоговый запрос без условия

Информация о количестве **клиентов**, сгруппированных по скорости интернета.

SELECT COUNT(c.id), t."speed(mb/s)" FROM clients AS c

INNER JOIN technical\_information AS t ON t.id = c.id\_technical\_information

GROUP BY t."speed(mb/s)" ORDER BY 1, 2

# Итоговый запрос с условием на данные

Информация о средней стоимости **тарифов** с указанной услугой, группированных по типу оплаты.

SELECT AVG(cost::numeric), s.name FROM tariffs AS t

INNER JOIN services AS s ON s.id = t.id\_service AND s.name = '%s'

GROUP BY s.name ORDER BY 1, 2

# Итоговый запрос с условием на группы

Информация о количестве прошедших **финансовые** **операции** денег по дням, выведенная за указанный промежуток.

SELECT SUM(sum::numeric), f.operation\_date FROM financial\_operations AS f

WHERE operation\_date BETWEEN '%s' AND '%s'

GROUP BY operation\_date ORDER BY 1, 2

# Итоговый запрос с условиями на данные и группы

Информация об отрицательных **счетах** указанного клиента.

SELECT c.name, SUM(balance::numeric) FROM bills AS b

INNER JOIN clients AS c ON c.id = b.id\_client

WHERE balance::numeric < 0 GROUP BY c.name HAVING c.name = '%s' ORDER BY 1, 2

# Запрос на запросе по принципу итогового

Информация о средней стоимости **тарифов**, сгруппированных по типу оплаты, полученная на основе запроса с симметричным внутренним соединением без условия.

SELECT AVG(t\_4::numeric), t\_3

FROM (SELECT t.name, t.id\_service, c.name AS t\_3, t.cost AS t\_4 FROM tariffs AS t

INNER JOIN charge\_types AS c ON c.id = t.id\_charge\_type) AS t

GROUP BY t.t\_3 ORDER BY 1

# Запрос с подзапросом

Информация о **счетах** клиентов, наименования которых начинаются на указанный ряд символов.

SELECT c.name, b.balance FROM bills AS b, clients AS c

WHERE c.id = b.id\_client AND b.id\_client IN

(SELECT id FROM clients WHERE name LIKE '%s%') ORDER BY 1, 2

# Специальный запрос 1

Информация о трёх **клиентах** с наибольшим количеством финансовых операций среди всех и сортированный список по городам.

SELECT '=== СРЕДИ ВСЕХ ===', null, '===', '===' UNION ALL (

SELECT c.name, COUNT(f.id), (SELECT name FROM client\_cities WHERE id = c.id\_city), c.contact\_information

FROM financial\_operations AS f

INNER JOIN bills AS b ON b.id = f.id\_bill

INNER JOIN clients AS c ON c.id = b.id\_client

GROUP BY c.id ORDER BY 2 DESC LIMIT 3)

UNION ALL ( SELECT '=== ПО ГОРОДАМ ===', null, '===', '===' ) UNION ALL (

SELECT c.name, COUNT(f.id), (SELECT name FROM client\_cities WHERE id = c.id\_city), c.contact\_information

financial\_operations AS f

INNER JOIN bills AS b ON b.id = f.id\_bill

INNER JOIN clients AS c ON c.id = b.id\_client

GROUP BY c.id\_city, c.id ORDER BY 3, 2 DESC)

# Специальный запрос 2

Информация о среднем количестве финансовых операций для **клиентов** и для **городов**.

SELECT 'для клиентов', AVG(from\_1.count) FROM (

SELECT COUNT(f.id) AS count FROM financial\_operations AS f

INNER JOIN bills AS b ON b.id = f.id\_bill

INNER JOIN clients AS c ON c.id = b.id\_client

GROUP BY c.id) AS from\_1 UNION ALL (

SELECT 'для городов', AVG(from\_2.count) AS "avg count" FROM (

SELECT COUNT(f.id) AS count FROM financial\_operations AS f

INNER JOIN bills AS b ON b.id = f.id\_bill

INNER JOIN clients AS c ON c.id = b.id\_client

INNER JOIN client\_cities AS cc ON cc.id = c.id\_city

GROUP BY cc.id) AS from\_2 ) ORDER BY 1

# Специальный запрос 3

Информация о количестве **клиентов**, совершивших **финансовые** **операции** за указанный период, и сумму денег, прошедших через эти операции.

SELECT f.operation\_date, COUNT(c.id), COUNT(b.id), SUM(f.sum)

FROM financial\_operations AS f

INNER JOIN bills AS b ON b.id = f.id\_bill

INNER JOIN clients AS c ON c.id = b.id\_client

WHERE f.operation\_date > '%s' AND f.operation\_date < '%s'

GROUP BY f.operation\_date

ORDER BY 1