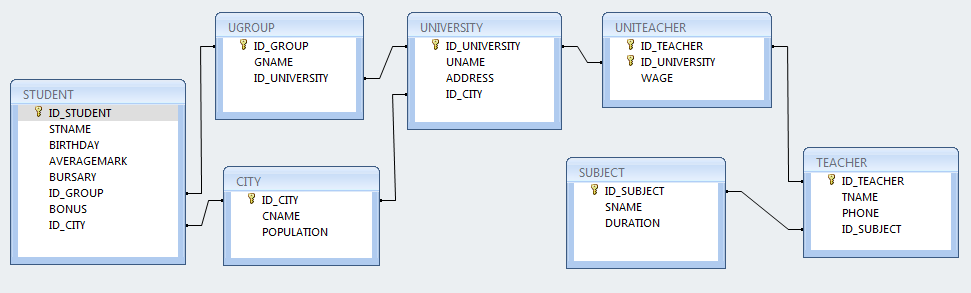
**Учебная база данных и тестовые задания**

ER-диаграмма учебной базы:



**Результат выполнения тестового задания – набор SQL-запросов для MSSQL.**

Тестовые задания:

1. Выбрать студентов из таблицы STUDENT, и место их проживания из таблицы CITY . Показать только тех студентов, которые учатся в университете по улице Бровки (Brovki) и отсортировать их по имени (Команда SELECT с условием, Несколько простых JOINs, LEFT OUTER JOIN, сортировка). Также необходимо решить задачу с помощью WITH. 1 sql-команда.

WITH infstudent (student, city, address)

AS

(SELECT s.stname, c.cname, u.address

FROM student s

LEFT JOIN city c ON s.id\_city = c.id\_city

JOIN ugroup ug ON s.id\_group = ug.id\_group

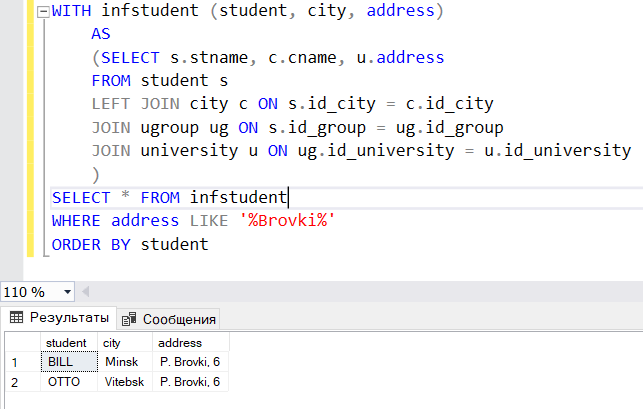
JOIN university u ON ug.id\_university = u.id\_university

)

SELECT \* FROM infstudent

WHERE address LIKE '%Brovki%'

ORDER BY student



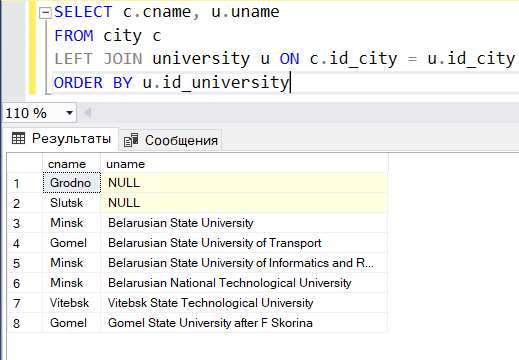
1. Показать все города и университеты, которые находятся в этих городах, отсортировать результат по номеру университета (LEFT JOIN). 1 sql-команда.

SELECT c.cname, u.uname

FROM city c

LEFT JOIN university u ON c.id\_city = u.id\_city

ORDER BY u.id\_university



1. Показать все предметы и преподавателей, которые их преподают, и среднюю зарплату каждого преподавателя по университетам (LEFT JOINS, групповая функция AVG, CONVERT). 1 sql-команда.

SELECT s.sname, t.tname,

CONVERT(bigint, AVG(ut.wage)),

u.uname

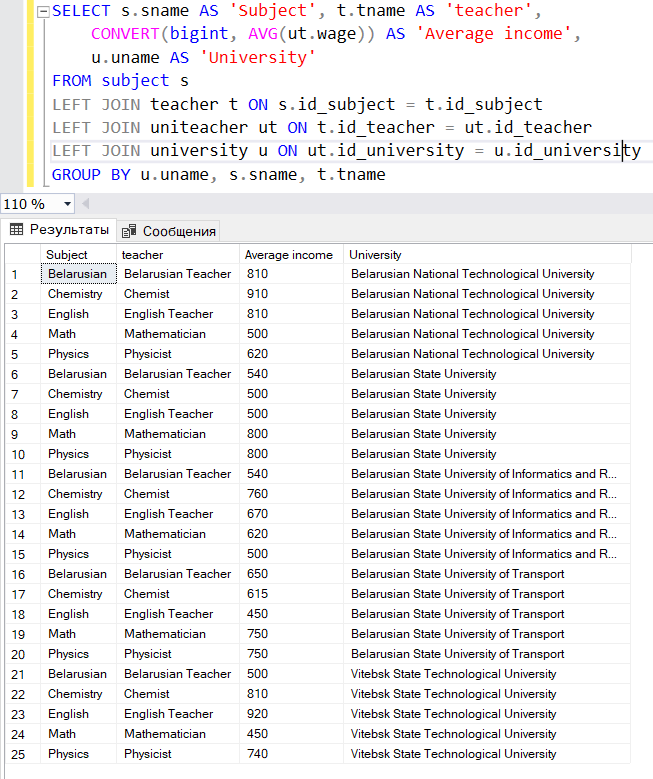
FROM subject s

LEFT JOIN teacher t ON s.id\_subject = t.id\_subject

LEFT JOIN uniteacher ut ON t.id\_teacher = ut.id\_teacher

LEFT JOIN university u ON ut.id\_university = u.id\_university

GROUP BY u.uname, s.sname, t.tname

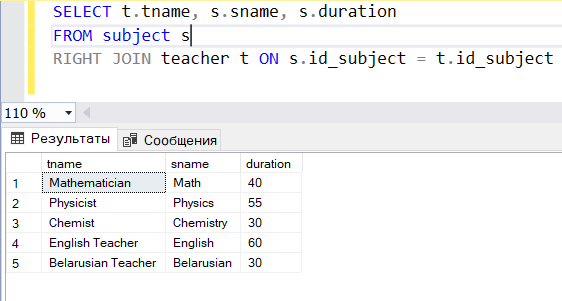


1. Показать всех преподавателей, предметы, которые они преподают, и их длительность (RIGHT JOIN). 1 sql-команда.

SELECT t.tname, s.sname, s.duration

FROM subject s

RIGHT JOIN teacher t ON s.id\_subject = t.id\_subject

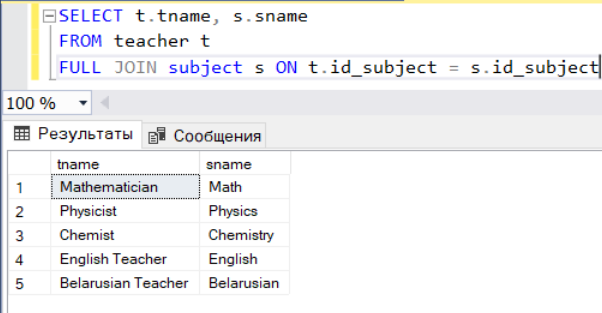


1. Показать всех преподавателей и все предметы (FULL JOIN). 1 sql-команда.

SELECT t.tname, s.sname

FROM teacher t

FULL JOIN subject s ON t.id\_subject = s.id\_subject



1. Подсчитать среднее число жителей, проживающих в городах из таблицы City (групповая функция AVG, CONVERT). 1 sql-команда.

Пример результата:

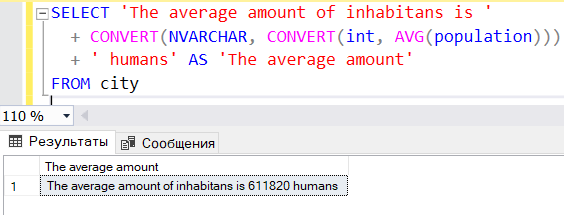


SELECT 'The average amount of inhabitans is '

+ CONVERT(NVARCHAR, CONVERT(int, AVG(population)))

+ ' humans' AS 'The average amount'

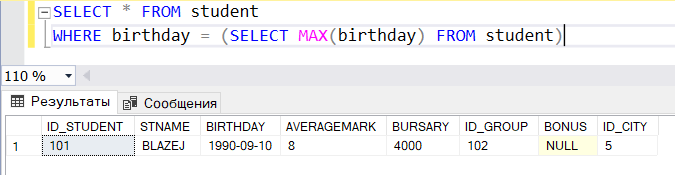
FROM city



1. Выбрать самого молодого студента из таблицы STUDENT (групповая функция MAX, подзапрос в условии). 1 sql-команда.

SELECT \* FROM student

WHERE birthday = (SELECT MAX(birthday) FROM student)



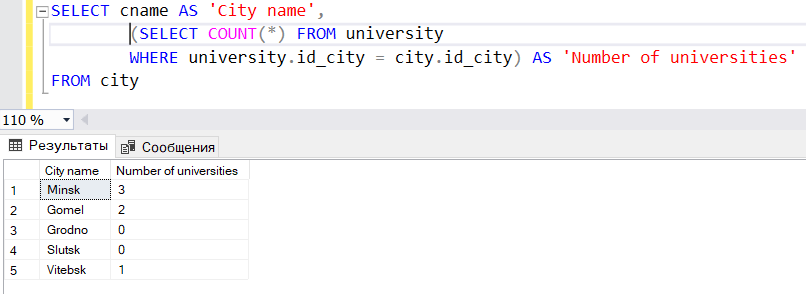
1. Показать все города и количество университетов в каждом городе (псевдостолбец-подзапрос, псевдонимы столбцов, групповая функция COUNT).1 sql-команда.

SELECT cname AS 'City name',

(SELECT COUNT(\*) FROM university

WHERE university.id\_city = city.id\_city) AS 'Number of universities'

FROM city



1. Показать средний доход (стипендию (BURSARY) + премию (BONUS)) студентов по группам из города Минска (Minsk) из виртуальной таблицы, которая является результатом объединения всех столбцов из таблиц STUDENT и CITY. Отсортировать группы по убыванию дохода. (Групповая функция AVG, подзапрос c простым JOIN в секции FROM, преобразование NULL в конкретное значение (функция ISNULL)) .1 sql-команда.

SELECT id\_group as 'Group',

CONVERT(int, AVG(ISNULL(s.bursary,0)+ISNULL(s.bonus,0))) AS 'Average student income in Minsk'

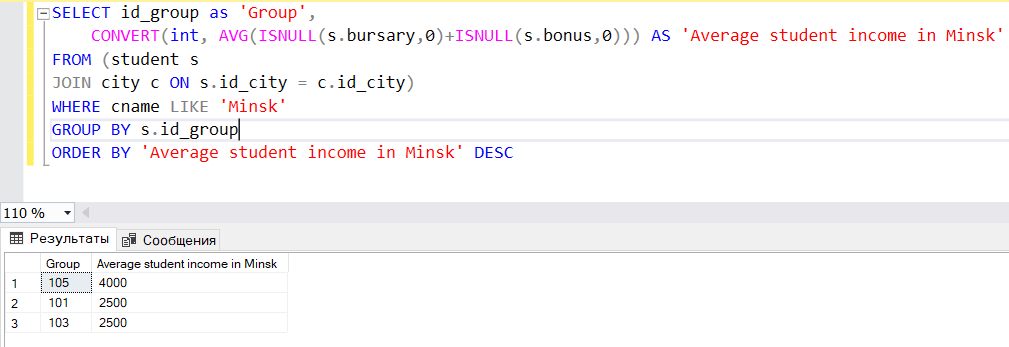
FROM (student s

JOIN city c ON s.id\_city = c.id\_city)

WHERE cname LIKE 'Minsk'

GROUP BY s.id\_group

ORDER BY 'Average student income in Minsk' DESC



1. Подсчитать среднюю стипендию по университетам и город, в котором он находится. Показать только те университеты, у которых суммарная сумма на выплаты студентам стипендии и премий больше 10000 (Несколько простых JOINs, групповые функции AVG, SUM, фильтр групп HAVING, преобразование NULL в конкретное значение). 1 sql-команда.

SELECT u.uname AS 'University', c.cname AS 'City',

CONVERT(bigint, AVG(ISNULL(s.bursary,0)+ISNULL(s.bonus,0))) AS 'Average student income'

FROM university u

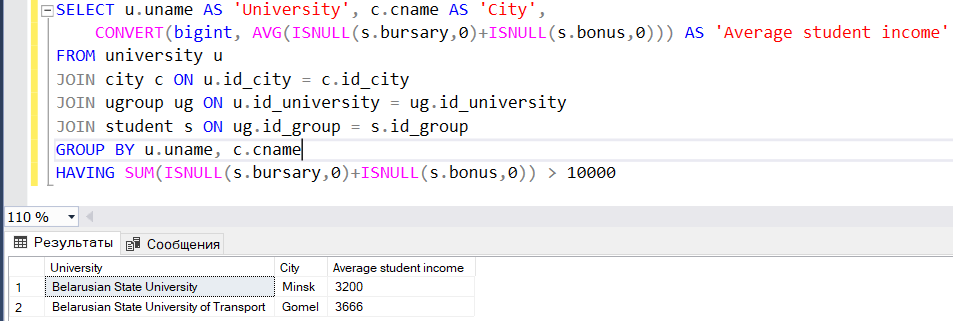
JOIN city c ON u.id\_city = c.id\_city

JOIN ugroup ug ON u.id\_university = ug.id\_university

JOIN student s ON ug.id\_group = s.id\_group

GROUP BY u.uname, c.cname

HAVING SUM(ISNULL(s.bursary,0)+ISNULL(s.bonus,0)) > 10000



1. Подсчитать суммарный доход преподавателей по университетам (групповая функция SUM, LEFT JOIN). 1 sql-команда.

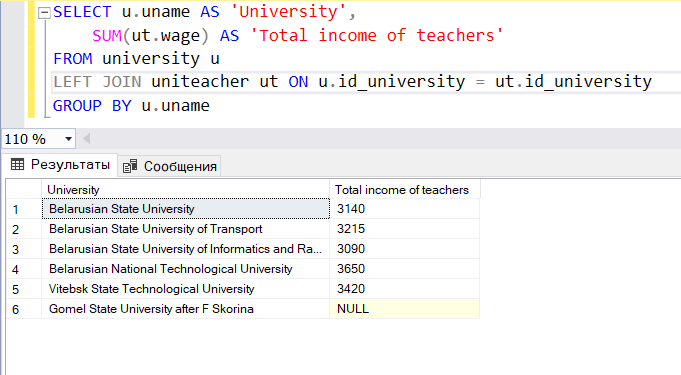
SELECT u.uname AS 'University',

SUM(ut.wage) AS 'Total income of teachers'

FROM university u

LEFT JOIN uniteacher ut ON u.id\_university = ut.id\_university

GROUP BY u.uname



1. Подсчитать количество студентов в каждом из университетов, 1 sql-команда.

SELECT u.uname AS 'University',

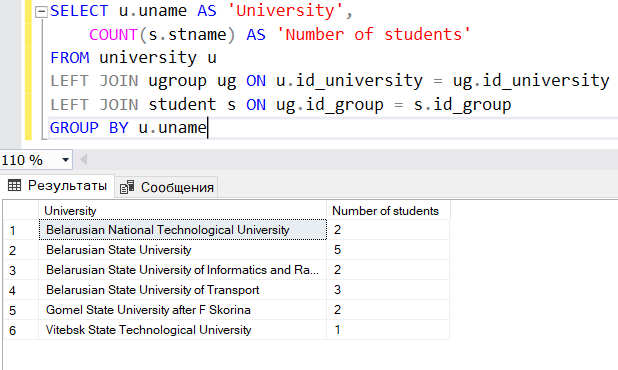
COUNT(s.stname) AS 'Number of students'

FROM university u

LEFT JOIN ugroup ug ON u.id\_university = ug.id\_university

LEFT JOIN student s ON ug.id\_group = s.id\_group

GROUP BY u.uname



1. Выбрать всех студентов из таблицы STUDENT, проживающих в Витебске, и отсортировать их по возрастанию их дохода (стипендию (BURSARY) + премию (BONUS)) (команда SELECT с условием, выражением, сортировкой, преобразование NULL в конкретное значение (функция COALESCE), псевдоним столбца). 1 sql-команда.

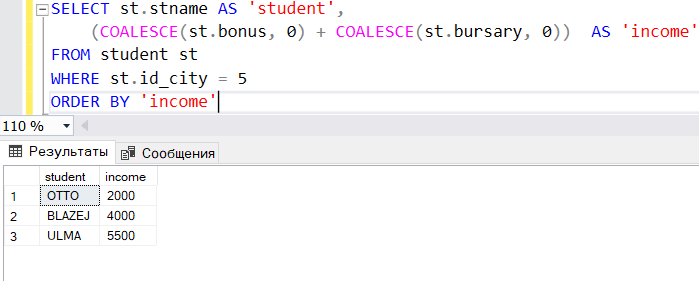
SELECT st.stname AS 'student',

(COALESCE(st.bonus, 0) + COALESCE(st.bursary, 0)) AS 'income'

FROM student st

WHERE st.id\_city = 5

ORDER BY 'income'



1. Показать всех студентов из таблицы STUDENT , родившихся в период от 01 января 1990 до 01 января 1991, имя города, в котором они проживают, и отсортировать их по убыванию их дохода (стипендию (BURSARY) + премию (BONUS)). (Команда SELECT с условием, выражением, сортировкой, преобразование NULL в конкретное значение, преобразование ID города в имя города при помощи конструкции CASE, псевдоним столбца) .1 sql-команда.

SELECT st.stname AS 'student',

(COALESCE(st.bonus, 0) + COALESCE(st.bursary, 0)) AS 'income',

'city' = (

CASE

WHEN st.id\_city = 1 THEN 'Minsk'

WHEN st.id\_city = 2 THEN 'Gomel'

WHEN st.id\_city = 3 THEN 'Grodno'

WHEN st.id\_city = 4 THEN 'Slutsk'

WHEN st.id\_city = 5 THEN 'Vitebsk'

ELSE 'Unknown city'

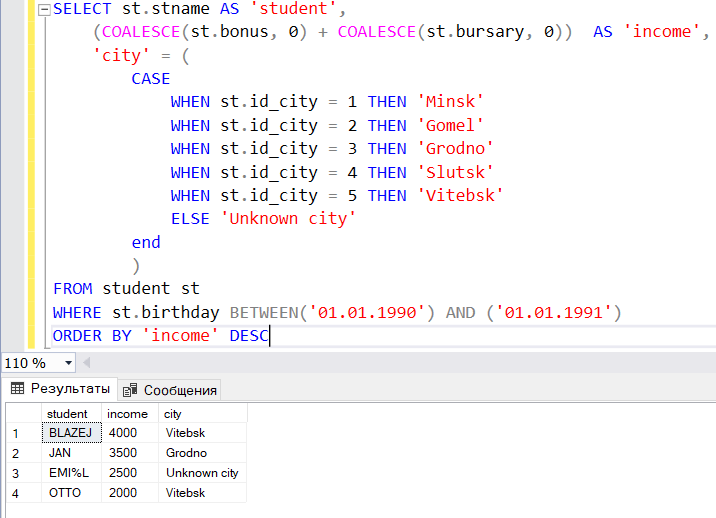
END

)

FROM student st

WHERE st.birthday BETWEEN('01.01.1990') AND ('01.01.1991')

ORDER BY 'income' DESC



1. Вывести студентов со средней оценкой (AVERAGEMARK) не менее 8 и обучающихся в университетах, содержащих в своих названиях (UNAME) слово «STATE» (использовать функции SUBSTRING и PATINDEX).

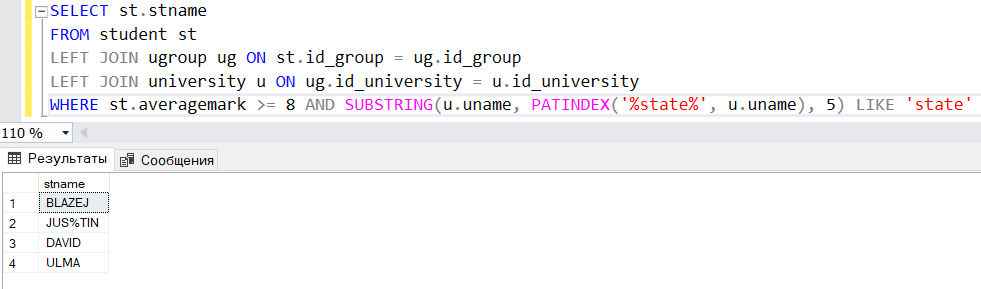
SELECT st.stname

FROM student st

LEFT JOIN ugroup ug ON st.id\_group = ug.id\_group

LEFT JOIN university u ON ug.id\_university = u.id\_university

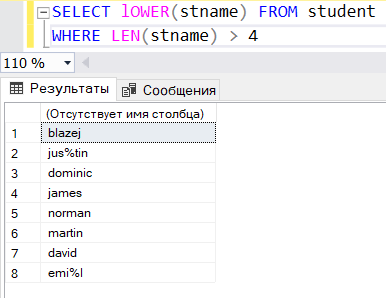
WHERE st.averagemark >= 8 AND SUBSTRING(u.uname, PATINDEX('%state%', u.uname), 5) LIKE 'state'



1. Вывести строчными буквами имена студентов (STNAME) с длиной имени более 4 символов.

SELECT lOWER(stname) FROM student

WHERE LEN(stname) > 4



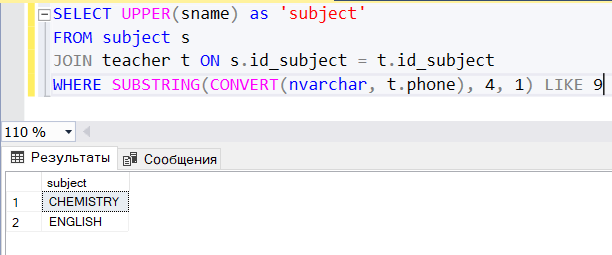
1. Вывести заглавными буквами названия предметов (SNAME), которые ведут преподаватели с телефонами (PHONE), содержащими цифру 9 на 4-й позиции (использовать функцию SUBSTRING).

SELECT UPPER(sname) as 'subject'

FROM subject s

JOIN teacher t ON s.id\_subject = t.id\_subject

WHERE SUBSTRING(CONVERT(nvarchar, t.phone), 4, 1) LIKE 9

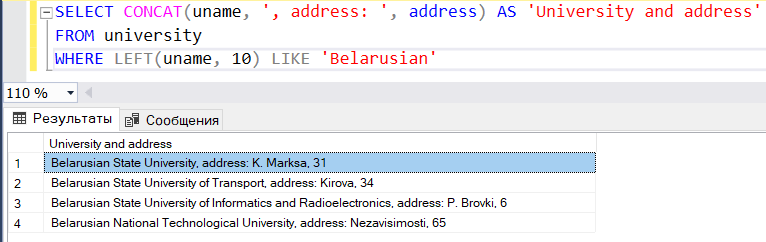


1. Объединить и вывести названия университетов (UNAME) и их адресов (ADDRESS), удовлетворяющих следующему условию: название университета начинается со слова «Belarusian» (использовать функции CONCAT и LEFT).

SELECT CONCAT(uname, ', address: ', address) AS 'University and address'

FROM university

WHERE LEFT(uname, 10) LIKE 'Belarusian'



1. Объединить и вывести названия университетов (UNAME) и их адресов (ADDRESS), удовлетворяющих следующему условию: название университета заканчивается словом «University» (использовать функции CONCAT и RIGHT).

SELECT CONCAT(uname, ', address: ', address) AS 'University and address'

FROM university

WHERE RIGHT(uname, 10) LIKE 'University'

