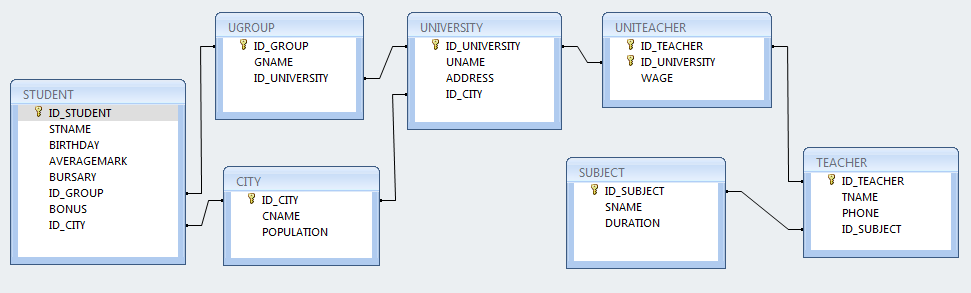
**Учебная база данных и тестовые задания**

ER-диаграмма учебной базы:



**Результат выполнения тестового задания – набор SQL-запросов для MSSQL. Просто поместите ваши ответы в этот документ! Задания, присланные в другом формате, проверяться не будут!**

**Отлаживать запросы можно с помощью любой из изученных утилит.**

Тестовые задания и ожидаемые результаты:

1. Выбрать все строки из таблицы City (команда SELECT со списком конкретных полей и без указания списка (символ “\*”)). 2 sql-команды.

select \* from city

Text

Description automatically generatedTable

Description automatically generated

select id\_city, cname from city

Graphical user interface, text

Description automatically generatedTable

Description automatically generated

1. Выбрать всех студентов из таблицы STUDENT учащихся в 101 группе (команда SELECT с условием). 1 sql-команда.

select\* from STUDENT

where ID\_GROUP = 101

Graphical user interface, text

Description automatically generated with medium confidence

Table

Description automatically generated

1. Выбрать всех студентов из таблицы STUDENT учащихся в 101, 105 и 106 группах (одна команда SELECT с несколькими условиями, SELECT c одним условием, три SELECT с одним условием в каждом и объединением результатов через UNION). 3 sql-команды.

select STNAME, AVERAGEMARK, ID\_GROUP from STUDENT

where ID\_GROUP = 101

UNION

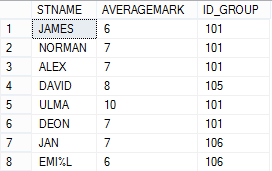
select STNAME, AVERAGEMARK, ID\_GROUP from STUDENT

where ID\_GROUP = 105

UNION

select STNAME, AVERAGEMARK, ID\_GROUP from STUDENT

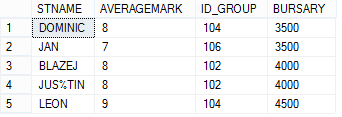
where ID\_GROUP = 106



1. Выбрать всех студентов из таблицы STUDENT учащихся в 102, 104 и 106 группах и получающих стипендию больше 3000 (команда SELECT с несколькими условиями). 1 sql-команда.

SELECT STNAME, AVERAGEMARK, ID\_GROUP, BURSARY FROM STUDENT

WHERE (ID\_GROUP IN (102,104,106) AND BURSARY >= 3000)



1. Выбрать всех студентов с именем, начинающимся на D, из таблицы STUDENT , средний балл которых от 7.4 до 9.5 (команда SELECT с несколькими условиями). 1 sql-команда.

SELECT STNAME, AVERAGEMARK, ID\_GROUP, BURSARY FROM STUDENT

WHERE (STNAME LIKE 'D%' AND AVERAGEMARK BETWEEN 7.4 AND 9.5)



1. \*Выбрать всех студентов с предпоследней буквой О в имени и их дни рождения в указанном формате (см. картинку) из таблицы STUDENT (функция convert() в MSSQL, команда SELECT c условием, псевдоним столбца). 1 sql-команда.

SELECT STNAME, CONVERT (varchar, BIRTHDAY, 113) AS 'Дата' FROM STUDENT

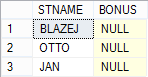
WHERE STNAME LIKE '%O\_'



1. Выбрать всех студентов из таблицы STUDENT не получающих премию (BONUS) и родившихся после первого января 1988 (команда SELECT с условием анализа на NULL-значение). 1 sql-команда.

SELECT STNAME, BONUS FROM STUDENT

WHERE (BIRTHDAY > '01/01/1988' AND BONUS IS NULL)



1. Показать уникальные стипендии (BURSARY) из таблицы STUDENT по городу Гомель (команда SELECT с условием).1 sql-команда.

SELECT DISTINCT STUDENT.BURSARY, STUDENT.ID\_CITY

FROM STUDENT

LEFT JOIN CITY ON CITY.ID\_CITY = STUDENT.ID\_CITY

WHERE CITY.CNAME IN ('Gomel')

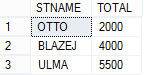


1. Выбрать всех студентов из таблицы STUDENT, проживающих в Витебске, и отсортировать их по возрастанию их дохода (стипендию (BURSARY) + премию (BONUS)) (команда SELECT с условием, выражением, сортировкой, преобразование NULL в конкретное значение (функция COALESCE в MSSQL), псевдоним столбца). 1 sql-команда.

SELECT STNAME, (BURSARY + COALESCE (BONUS, 0)) AS TOTAL FROM STUDENT

WHERE ID\_CITY = 5

ORDER BY TOTAL ASC



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 (функция COALESCE в Oracle), преобразование ID города в имя города при помощи конструкции CASE, псевдоним столбца) .1 sql-команда.

SELECT STNAME, (BURSARY + COALESCE (BONUS, 0)) AS TOTAL,

CASE

WHEN ID\_CITY = '1' THEN 'Minsk'

WHEN ID\_CITY = '2' THEN 'Gomel'

WHEN ID\_CITY = '3' THEN 'Grodno'

WHEN ID\_CITY = '4' THEN 'Slutsk'

WHEN ID\_CITY = '5' THEN 'Vitebsk'

ELSE 'Unknown'

END AS CITY

FROM STUDENT

WHERE BIRTHDAY BETWEEN '01/01/1990' AND '01/01/1991'

ORDER BY TOTAL DESC

Graphical user interface, text

Description automatically generatedTable

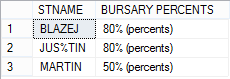
Description automatically generated

1. \*Максимальная стипендия 5000. Показать студентов из таблицы STUDENT, учащихся в группе 102 и их стипендию как процент от максимальной возможной стипендии (команда SELECT с условием, выражениями, оператор склейки строк – «+» для MSSQL). 1 sql-команда.

SELECT STUDENT.STNAME, CONCAT (CONVERT (int, ((STUDENT.BURSARY/5000) \* 100)),'% (percents)') AS "BURSARY PERCENTS"

FROM STUDENT

WHERE STUDENT.ID\_GROUP= 102



1. Показать студентов из таблицы STUDENT, не проживающих в Минске, Гомеле, Гродно. Показать также студентов, город проживания которых неизвестен. Отсортировать от A до Z (команда SELECT с условием, преобразование NULL в конкретное значение). 1 sql-команда.

SELECT STUDENT.STNAME

FROM STUDENT

LEFT JOIN CITY ON CITY.ID\_CITY = STUDENT.ID\_CITY

WHERE ISNULL (CITY.CNAME,'NULL') NOT IN ('Gomel','Minsk','Grodno')

ORDER BY STUDENT.STNAME

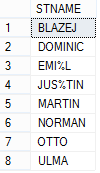
OR  
  
SELECT STUDENT.STNAME

FROM STUDENT

LEFT JOIN CITY ON CITY.ID\_CITY = STUDENT.ID\_CITY

WHERE COALESCE (CITY.CNAME,'NULL') NOT IN ('Gomel','Minsk','Grodno')

ORDER BY STUDENT.STNAME



1. Показать студентов из таблицы STUDENT, у которых в имени присутствует символ % (поиск по шаблону с использованием LIKE) . 1 sql-команда.

SELECT STNAME FROM STUDENT

WHERE STNAME LIKE '%' + replace ('%', '%', '[%]') + '%'

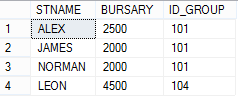


1. Показать студентов и их стипендию из таблицы STUDENT, которые родились до 23.04.1984. Отсортировать результаты по группе и имени (команда SELECT с условиями, сортировки) . 1 sql-команда.

SEKECT STNAME, BURSARY, ID\_GROUP FROM STUDENT

WHERE BIRTHDAY < '1984/04/23'

ORDER BY ID\_GROUP, STNAME ASC

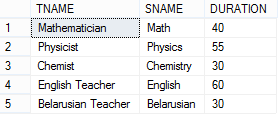


1. Показать имена преподавателей из таблицы TEACHER, и преподаваемый ими предмет и его длительность из таблицы SUBJECT (простой JOIN). 1 sql-команда.

SELECT TEACHER.TNAME, SUBJECT.SNAME, SUBJECT.DURATION

FROM SUBJECT

INNER JOIN TEACHER ON SUBJECT.ID\_SUBJECT = TEACHER.ID\_SUBJECT



1. Показать названия групп из таблицы UGROUP, названия университетов этих групп из таблицы UNIVERSITY, и город из таблицы CITY, в котором находиться этот университет. (Несколько простых JOINs). 1 sql-команда.

SELECT UGROUP.GNAME, UNIVERSITY.UNAME, CITY.CNAME

FROM UNIVERSITY

JOIN UGROUP ON UNIVERSITY.ID\_UNIVERSITY = UGROUP.ID\_UNIVERSITY

JOIN CITY ON CITY.ID\_CITY = UNIVERSITY.ID\_CITY



1. Выбрать студентов и их средний балл из таблицы STUDENT, их группу из таблицы UGROUP, их город из таблицы CITY, отфильтровать только тех студентов, где средний балл не превышает 6.2 (Команда SELECT с условием, Несколько JOINs). 1 sql-команда.

SELECT STUDENT.STNAME, STUDENT.AVERAGEMARK, UGROUP.GNAME, CITY.CNAME

FROM STUDENT

JOIN UGROUP ON UGROUP.ID\_GROUP = STUDENT.ID\_GROUP

JOIN CITY ON CITY.ID\_CITY = STUDENT.ID\_CITY

WHERE AVERAGEMARK < 6.2



1. Выбрать студентов из таблицы STUDENT, их университет из таблицы UNIVERSITY, их город из таблицы CITY, отфильтровать только тех студентов, в чьем городе население не больше 340000 и название группы содержит строчку «Uni» (Команда SELECT с условиями, Несколько JOINs) . 1 sql-команда

SELECT STUDENT.STNAME, UNIVERSITY.UNAME, CITY.CNAME

FROM STUDENT

JOIN CITY ON CITY.ID\_CITY = STUDENT.ID\_CITY

JOIN UGROUP ON STUDENT.ID\_GROUP = UGROUP.ID\_GROUP

JOIN UNIVERSITY ON UNIVERSITY.ID\_UNIVERSITY = UGROUP.ID\_UNIVERSITY

WHERE POPULATION <= 340000

AND UGROUP.GNAME LIKE '%Uni%'



1. Выбрать преподавателей из таблицы TEACHER, их заработную плату из таблицы UNITEACHER и название университета, в котором они ее получают, из таблицы UNIVERSITY, отфильтровать только тех преподавателей, чья заработная плата не менее 750. (символ “\*” для указанной таблицы, команда SELECT с условием, Несколько простых JOINs) . 1 sql-команда

SELECT TEACHER.ID\_TEACHER, TEACHER.TNAME, TEACHER.PHONE, TEACHER.ID\_SUBJECT, UNIVERSITY.UNAME, UNITEACHER.WAGE

FROM TEACHER

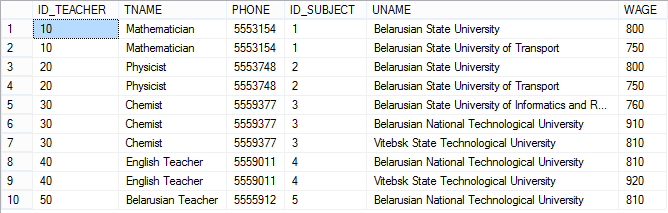
JOIN UNITEACHER ON UNITEACHER.ID\_TEACHER = TEACHER.ID\_TEACHER

JOIN UNIVERSITY ON UNIVERSITY.ID\_UNIVERSITY = UNITEACHER.ID\_UNIVERSITY

WHERE WAGE >= 750

A picture containing table

Description automatically generated



1. Выбрать преподавателей из таблицы TEACHER, их заработную плату из таблицы UNITEACHER, их предмет из таблицы SUBJECT и группы, в которых они его читают, из таблицы UGROUP, показать только группы из городов Минска и Гродно (Minsk, Grodno), не показывать предмет английский (English). Отсортировать по названию предмета, зарплате и по номеру группы. (Команда SELECT с условиями, Несколько простых JOINs) 1 sql-команда

SELECT TEACHER.TNAME, UGROUP.GNAME, UNITEACHER.WAGE, SUBJECT.SNAME

FROM TEACHER

JOIN UNITEACHER ON TEACHER.ID\_TEACHER = UNITEACHER.ID\_TEACHER

JOIN UGROUP ON UGROUP.ID\_UNIVERSITY = UNITEACHER.ID\_UNIVERSITY

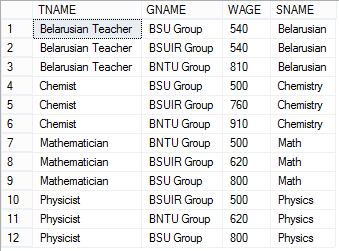
JOIN SUBJECT ON SUBJECT.ID\_SUBJECT = TEACHER.ID\_SUBJECT

JOIN UNIVERSITY ON UNITEACHER.ID\_UNIVERSITY = UNIVERSITY.ID\_UNIVERSITY

JOIN CITY ON UNIVERSITY.ID\_CITY = CITY.ID\_CITY

WHERE SUBJECT.SNAME NOT LIKE 'ENGLISH' AND CITY.CNAME IN ('Minsk', 'GRODNO')

ORDER BY SNAME, WAGE, GNAME, TNAME



**Результат выполнения тестового задания – набор SQL-запросов для MSSQL. Отлаживать запросы можно с помощью любой из изученных утилит.**