**Лабораторная работа № 7. Использование представлений**

1. Разработать представление с именем Преподаватель. Представление должно быть построено на основе SELECT-запроса к таблице TEACHER и содержать следующие столбцы: код (TEACHER), имя преподавателя (TEACHER\_NAME), пол (GENDER), код кафедры (PULPIT).

CREATE VIEW [TEACHER\_V]

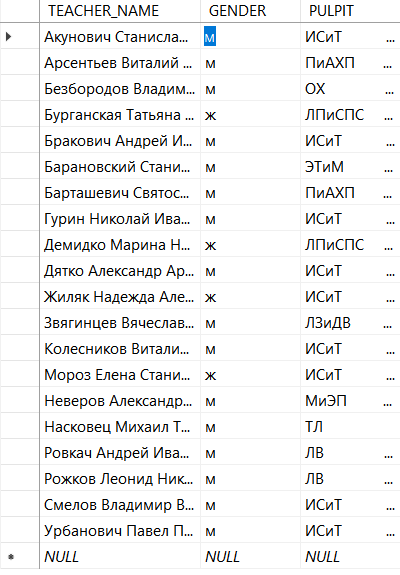
AS SELECT

TEACHER.TEACHER\_NAME,

TEACHER.GENDER,

TEACHER.PULPIT

FROM TEACHER;



1. Разработать и создать представление с именем Количество кафедр. Представление должно быть построено на основе SELECT-запроса к таблицам FACULTY и PULPIT. Представление должно содержать следующие столбцы: факультет (FACULTY.FACULTY\_ NAME), количество кафедр (вычисляется на основе строк таблицы PULPIT).

CREATE VIEW [PULPIT\_COUNT]

AS SELECT

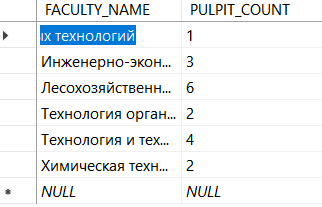
FACULTY.FACULTY\_NAME,

COUNT(PULPIT.FACULTY) [PULPIT\_COUNT]

FROM FACULTY

JOIN PULPIT ON FACULTY.FACULTY = PULPIT.FACULTY

GROUP BY PULPIT.FACULTY, FACULTY.FACULTY\_NAME;



1. Разработать и создать представление с именем **Аудитории**. Представление должно быть построено на основе таблицы **AUDITORIUM** и содержать столбцы: **код** (**AUDITORIUM**), **наименование аудитории** (**AUDITORIUM\_NAME**)**.**

GO

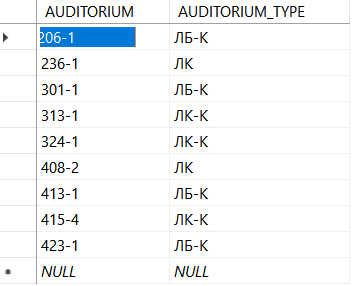
CREATE VIEW [AUDITORIUMS]

AS SELECT

AUDITORIUM.AUDITORIUM,

AUDITORIUM.AUDITORIUM\_TYPE

FROM AUDITORIUM;



1. Разработать и создать представление с именем **Лекционные\_аудитории**. Представление должно быть построено на основе SELECT-запроса к таблице **AUDITORIUM** и содержать следующие столбцы: **код** (**AUDITORIUM**), **наименование аудитории** (**AUDITORIUM\_NAME**)**.** Представление должно отображать только лекционные аудитории (в столбце **AUDITORIUM\_TYPE** строка, начинающаяся с символов **ЛК**). Выполнение INSERT и UPDATE допускается, но с учетом ограничения, задаваемого опцией WITH CHECK OPTION.

use UNIVER;

GO

CREATE VIEW [LECTURE\_AUDITORIUMS]

AS SELECT

AUDITORIUM.AUDITORIUM,

AUDITORIUM.AUDITORIUM\_TYPE

FROM AUDITORIUM

WHERE AUDITORIUM.AUDITORIUM\_TYPE LIKE 'ЛК%' WITH CHECK OPTION;

GO

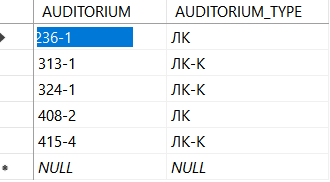
INSERT LECTURE\_AUDITORIUMS VALUES('315-4', 'ЛК-К');

GO

UPDATE [LECTURE\_AUDITORIUMS]

SET AUDITORIUM = '415-4'

WHERE AUDITORIUM = '315-4';



1. Разработать представление с именем **Дисциплины**. Представление должно быть построено на основе SELECT-запроса к таблице **SUBJECT**, отображать все дисциплины в алфавитном порядкеи содержать следующие столбцы: **код** (**SUBJECT**), **наименование дисциплины** (**SUBJECT\_NAME**) и **код кафедры** (**PULPIT**). Использовать TOP и ORDER BY.

use UNIVER;

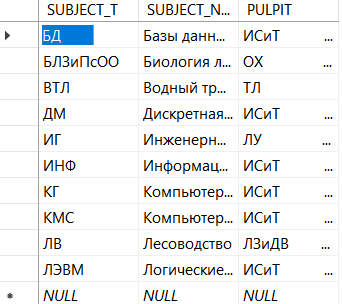
GO

CREATE VIEW [SUBJECTS]

AS SELECT TOP(10) \*

FROM SUBJECT\_T

ORDER BY SUBJECT\_T.SUBJECT\_NAME;



1. Изменить представление **Количество\_кафедр**, созданное в задании 2 так, чтобы оно было привязано к базовым таблицам. Продемонстрировать свойство привязанности представления к базовым таблицам. Примечание: использовать опцию SCHEMABINDING.

use UNIVER;

GO

ALTER VIEW [PULPIT COUNT] WITH SCHEMABINDING

AS SELECT

FACULTY\_NAME,

COUNT(PULPIT) [COUNT]

FROM dbo.FACULTY

JOIN dbo.PULPIT ON FACULTY.FACULTY = PULPIT.FACULTY

GROUP BY FACULTY\_NAME;

1. Разработать представления для базы данных **X\_MyBASE**.

use Dobriyan\_MyBase;

-- #1

GO

CREATE VIEW [TEACHER\_V]

AS SELECT

Teachers.Surname,

Teachers.Name,

Teachers.Telephone

FROM Teachers;

-- #2

use Dobriyan\_MyBase;

GO

CREATE VIEW [Active\_Courses\_Count]

AS SELECT

Teachers.Surname,

Teachers.Name,

COUNT(Teachers.Active\_Course) [COUNT]

FROM Teachers

JOIN Courses ON Teachers.Active\_Course = Courses.Course\_Subject

GROUP BY Teachers.Active\_Course, Teachers.Surname, Teachers.Name;

-- #3

use Dobriyan\_MyBase;

GO

CREATE VIEW [Groups\_V]

AS SELECT

Groups.Speciality,

Groups.Students\_Count

FROM Groups;

GO

INSERT Groups\_V VALUES (3, 19);

-- #4

use Dobriyan\_MyBase;

GO

CREATE VIEW [Design\_Groups]

AS SELECT

Groups.Number,

Groups.Students\_Count

FROM Groups

WHERE Groups.Speciality = 2 WITH CHECK OPTION;

GO

use Dobriyan\_MyBase;

INSERT [Design\_Groups] VALUES(10, 21);

GO

UPDATE [Design\_Groups]

SET Students\_Count = 20

WHERE Number = 5;

-- #5

use Dobriyan\_MyBase;

GO

CREATE VIEW [Courses\_V]

AS SELECT TOP(5) \*

FROM Courses

ORDER BY Courses.Course\_Subject;

-- #6

use Dobriyan\_MyBase;

GO

ALTER VIEW [Active\_Courses\_Count] WITH SCHEMABINDING

AS SELECT

Teachers.Surname,

Teachers.Name,

COUNT(Teachers.Active\_Course) [COUNT]

FROM dbo.Teachers

JOIN dbo.Courses ON Teachers.Active\_Course = Courses.Course\_Subject

GROUP BY Teachers.Active\_Course, Teachers.Surname, Teachers.Name;