**Задание 1.**

Необходимо создать базу данных Академия (Academy), которая будет содержать информацию о сотрудниках и внутреннем устройстве академии.

**create database** Academy;

**Кафедры (Departments)**

* Идентификатор (Id). Уникальный идентификатор кафедры.
* Тип данных – int.
* Авто приращение.
* Не может содержать null-значения.
* Первичный ключ.
* Финансирование (Financing). Фонд финансирования кафедры.
* Тип данных – money.
* Не может содержать null-значения.
* Не может быть меньше 0.
* Значение по умолчанию - 0.
* Название (Name). Название кафедры.
* Тип данных – nvarchar(100).
* Не может содержать null-значения.
* Не может быть пустым.
* Должно быть уникальным.

**create table** Departments

(

Id **int** **not** **null** **auto\_increment**,

Financing **decimal**(10,2) **check**(Financing >= 0) **not** **null default** 0,

Name **varchar**(100) **not** **null unique**,

**primary** **key** (Id)

);

**Факультеты (Faculties)**

* Идентификатор (Id). Уникальный идентификатор факультета.
* Тип данных – int.
* Авто приращение.
* Не может содержать null-значения.
* Первичный ключ.
* Декан (Dean). Декан факультета.
* Тип данных – nvarchar(max).
* Не может содержать null-значения.
* Не может быть пустым.
* Название (Name). Название факультета.
* Тип данных – nvarchar(100).
* Не может содержать null-значения.
* Не может быть пустым.
* Должно быть уникальным.

**create table** Faculties

(

Id **int** **not** **null** **auto\_increment**,

Dean **varchar**(max) **not** **null**,

Name **varchar**(100) **not** **null unique**,

**primary** **key** (Id)

);

**Группы (Groups)**

* Идентификатор (Id). Уникальный идентификатор группы.
* Тип данных – int.
* Авто приращение.
* Не может содержать null-значения.
* Первичный ключ.
* Название (Name). Название факультета.
* Тип данных – nvarchar(10).
* Не может содержать null-значения.
* Не может быть пустым.
* Должно быть уникальным.
* Рейтинг (Rating). Рейтинг группы.
* Тип данных – int.
* Не может содержать null-значения.
* Должно быть в диапазоне от 0 до 5.
* Курс (Year). Курс (год) на котором обучается группа
* Тип данных – int.
* Не может содержать null-значения.
* Должно быть в диапазоне от 1 до 5.

**create table** Groups\_Academy

(

Id **int** **not** **null** **auto\_increment**,

Name **varchar**(10) **not** **null unique**,

Rating **int** **not** **null check**(Rating <= 0 **and** Rating >= 5),

Year **int** **not** **null check**(Year <= 1 **and** Year >= 5),

**primary** **key** (Id)

);

**Преподаватели (Teachers)**

* Идентификатор (Id). Уникальный идентификатор преподавателя.
* Тип данных – int.
* Авто приращение.
* Не может содержать null-значения.
* Первичный ключ.
* Дата трудоустройства (EmploymentDate). Дата приема преподавателя на работу.
* Тип данных – date.
* Не может содержать null-значения.
* Не может быть меньше 01.01.1990.
* Профессор (IsProfessor). Является ли преподаватель профессором.
* Тип данных – bit.
* Не может содержать null-значения.
* Значение по умолчанию - 0.
* Ассистент (IsAssistant). Является ли преподаватель ассистентом.
* Тип данных – bit.
* Не может содержать null-значения.
* Значение по умолчанию - 0.
* Название (Name). Название факультета.
* Тип данных – nvarchar(max).
* Не может содержать null-значения.
* Не может быть пустым.
* Должность (Position). Должность преподавателя.
* Тип данных – nvarchar(max).
* Не может содержать null-значения.
* Не может быть пустым.
* Надбавка (Premium). Надбавка преподавателя.
* Тип данных – money.
* Не может содержать null-значения.
* Не может быть меньше 0.
* Значение по умолчанию - 0.
* Ставка (Salary). Ставка преподавателя.
* Тип данных – money.
* Не может содержать null-значения.
* Не может быть меньше либо равно 0.
* Фамилия (Surname). Фамилия преподавателя.
* Тип данных – nvarchar(max).
* Не может содержать null-значения.
* Не может быть пустым.

**create table** Teachers

(

Id **int** **not** **null** **auto\_increment**,

EmploymentDate **date** **check**(EmploymentDate >= **'1990-01-01'**) **not** **null**,

IsProfessor **bit** **not** **null**  **default** 0,

IsAssistant **bit** **not** **null**  **default** 0,

Name **varchar**(max) **not** **null**,

Position **varchar**(max) **not** **null**,

Premium **decimal**(10,2) **check**(Premium >= 0) **not** **null default** 0,

Salary **decimal**(10,2) **check**(Salary > 0) **not** **null**,

Surname **varchar**(max) **not** **null**,

**primary** **key** (Id)

);

**Задание 2.**

Для базы данных Академия создайте такие запросы:

1. Вывести таблицу кафедр, но расположить ее поля в обратном порядке.

**select** Name, Financing, Id **from** departments;

2. Вывести названия групп и их рейтинги с уточнением имен полей именем таблицы.

**select** Name, Rating **from** groups\_academy;

3. Вывести для преподавателей их фамилию, процент ставки по отношению к надбавке и процент ставки по отношению к зарплате (сумма ставки и надбавки).

**select** Name, Salary, Premium, Salary + Premium, Premium \* 100 / Salary **from** teachers;

4. Вывести таблицу факультетов в виде одного поля в следующем формате: "The dean of faculty [faculty] is [dean].".

**select** **concat**(**'The dean of faculty'**, Name, **' is '**, Dean, **'.'**) **from** faculties;

5. Вывести фамилии преподавателей, которые являются профессорами и ставка которых превышает 1050.

**select** Surname **from** teachers **where** Salary > 1050;

6. Вывести названия кафедр, фонд финансирования которых меньше 11000 или больше 25000.

**select** Name, Financing **from** departments **where** Financing < 11000 **and** Financing > 25000;

7. Вывести названия факультетов кроме факультета “Computer Science”.

**select** Name **from** faculties **where** Name <> **'Computer Science'**;

8. Вывести фамилии и должности преподавателей, которые не являются профессорами.

**select** Surname, Position\_ **from** teachers **where** Position\_ <> **'Professor'**;

9. Вывести фамилии, должности, ставки и надбавки ассистентов, у которых надбавка в диапазоне от 160 до 550.

**select** Surname, Position\_, Salary, Premium **from** teachers **where** Position\_ = **'Assistant'** **and** (Premium >= 160 **and** Premium <= 550);

10. Вывести фамилии и ставки ассистентов.

**select** Surname, Salary **from** teachers **where** Position\_ = **'Assistant'**;

11. Вывести фамилии и должности преподавателей, которые были приняты на работу до 01.01.2000.

**select** Surname, Position\_ **from** teachers **where** EmploymentDate < **'2000-01-01'**;

12. Вывести названия кафедр, которые в алфавитном порядке располагаются до кафедры "Software Development”. Выводимое поле должно иметь название “Name of Department”.

**select** Name **as** **'[Name] of Department'** **from** departments **where** Name < **'Software Development'** **order** **by** Name;

13. Вывести фамилии ассистентов, имеющих зарплату (сумма ставки и надбавки) не более 1200.

**select** Surname **from** teachers **where** Position\_ = **'Assistant'** **and** (Salary + Premium) < 1200;

14. Вывести названия групп 5-го курса, имеющих рейтинг в диапазоне от 2 до 4.

**select** Name **from** groups\_academy **where** Year == 5 **and** (Rating >= 2 **and** Rating <= 4);

15. Вывести фамилии ассистентов со ставкой меньше 550 или надбавкой меньше 200.

**select** Surname **from** teachers **where** Position\_ = **'Assistant'** **and** Salary < 550 **and** Premium < 200;