

ВАРИАНТ 1. ОТДЕЛ КАДРОВ ИНСТИТУТА

Пусть требуется разработать информационную систему для автоматизации учета кадров и учебной нагрузки в институте. Система должна предусматривать ведение системного каталога кафедр и учебных дисциплин.

1. Системный каталог кафедр

Каждая кафедра в институте характеризуется следующими параметрами:

- Уникальный шифр;
- Название кафедры;
- Факультет.

Кафедры могут иметь одинаковые названия на разных факультетах, но однозначно идентифицируются по своему уникальному шифру.

2. Учет дисциплин

Каждая учебная дисциплина, представленная в институте, характеризуется следующими параметрами:

- Уникальный код дисциплины;
- Название;
- Курс, на котором читается дисциплина;
- Количество лекционных часов;
- Количество семинарских часов;
- Количество часов практических занятий.

Одна дисциплина может преподаваться на нескольких кафедрах, но ведется под одним уникальным кодом.

3. Картотека преподавателей

На каждого преподавателя в картотеку заносятся следующие сведения:

- Табельный номер (уникальный);
- ФИО (обязательное поле);
- Пол (допустимые значения: 'м' и 'ж', по умолчанию – 'ж');
- Дата рождения (обязательное поле);
- Кафедра (внешний ключ к таблице "Кафедры");
- Должность (ассистент, преподаватель, доцент, профессор);
- Ученая степень (кандидат или доктор наук);
- Ученое звание (доцент или профессор).

Один преподаватель может работать только на одной кафедре, но может вести несколько дисциплин.

4. Учет учебной нагрузки

Для учета нагрузки фиксируется, какой преподаватель и по какой дисциплине ведет определенный тип занятий. Каждая запись о нагрузке содержит:

- Уникальный идентификатор (ID);
- Преподаватель (внешний ключ);
- Дисциплина (внешний ключ);
- Тип нагрузки (лекция, семинар или практические занятия).

Одна дисциплина может быть распределена между несколькими преподавателями по разным видам нагрузки, но каждый конкретный тип нагрузки по дисциплине закрепляется за одним преподавателем.

Ограничения на информацию в системе:

1. Преподаватель должен быть привязан к существующей кафедре.
2. Нагрузка должна назначаться только существующим преподавателям и по существующим дисциплинам.
3. Поле «Ученая степень» и «Ученое звание» могут быть пустыми.
4. Каждый преподаватель может вести несколько дисциплин, но не более 5 одновременно.

С данной информационной системой должны работать следующие группы пользователей:

- Сотрудники отдела кадров;
- Деканаты факультетов;
- Бухгалтерия (для расчета заработной платы).

Задачи сотрудника отдела кадров:

1. Регистрировать новых преподавателей в системе, занося их полные данные.
2. Вносить изменения в данные о преподавателях (смена должности, ученой степени, звания, перевод на другую кафедру).
3. Вести учет кафедр и дисциплин.
4. Формировать отчеты по кадровому составу.

Задачи деканата:

1. Формировать учебные планы и распределять нагрузку по преподавателям.
2. Просматривать нагрузку по кафедрам и отдельным преподавателям.

3. Анализировать качественный состав кафедр (количество профессоров, доцентов и т.д.).
4. Получать списки преподавателей, отсортированные по различным критериям (например, по ученой степени, должности, кафедре).

Задачи бухгалтерии:

1. Использовать данные о должностях и нагрузке для расчета заработной платы.

Требования к отчетности и выборкам данных:

- Формирование списка докторов наук с указанием названия кафедры и должности.
- Выявление самых молодых преподавателей на каждой кафедре.
- Контроль назначения преподавателей на дисциплины: получение списка дисциплин и видов нагрузки, для которых не назначены преподаватели.
- Анализ кадрового потенциала кафедр: получение списка кафедр, на которых работает менее 2-х докторов наук.
- Статистика по кафедрам: подсчет количества преподавателей с разными учеными степенями.

Входные данные

Входные данные вводятся пользователями (в основном сотрудниками отдела кадров) через интерфейсы системы. Они включают обязательные и необязательные поля, с проверкой на соответствие ограничениям (например, уникальность ключей, допустимые значения для пола или должности).

1. **Для кафедр** (ввод при регистрации или редактировании кафедры):
 - Уникальный шифр (обязательное поле, строковый код, уникальный в системе).
 - Название кафедры (обязательное поле, строка).
 - Факультет (обязательное поле, строка или ссылка на справочник факультетов).
 - Дополнительно: дата создания кафедры (необязательное, дата) для исторического учета.
2. **Для преподавателей** (ввод при регистрации нового преподавателя или редактировании данных):
 - Табельный номер (обязательное поле, уникальный числовой идентификатор).
 - ФИО (обязательное поле, строка).
 - Пол (обязательное поле, значения: 'м' или 'ж', по умолчанию 'ж').

- Дата рождения (обязательное поле, дата).
- Кафедра (обязательное поле, внешний ключ — шифр кафедры из каталога кафедр).
- Должность (обязательное поле, значения: ассистент, преподаватель, доцент, профессор).
- Ученая степень (необязательное поле, значения: кандидат или доктор наук).
- Ученое звание (необязательное поле, значения: доцент или профессор).
- Дополнительно: дата приема на работу (необязательное, дата) для расчета стажа.

3. **Для дисциплин** (ввод при добавлении или редактировании дисциплины в каталог):

- Уникальный код дисциплины (обязательное поле, строковый код, уникальный в системе).
- Название (обязательное поле, строка).
- Курс, на котором читается дисциплина (обязательное поле, целое число, от 1 до 6).
- Количество лекционных часов (обязательное поле, целое число).
- Количество семинарских часов (обязательное поле, целое число).
- Количество часов практических занятий (обязательное поле, целое число).
- Дополнительно: семестр (необязательное, значения: осенний или весенний) для планирования.

4. **Для нагрузки** (ввод при распределении нагрузки, с проверкой на существующих преподавателей и дисциплины):

- Уникальный идентификатор (ID, автоматически генерируется системой).
- Преподаватель (обязательное поле, внешний ключ — табельный номер преподавателя).
- Дисциплина (обязательное поле, внешний ключ — код дисциплины).
- Тип нагрузки (обязательное поле, значения: лекция, семинар или практические занятия).
- Дополнительно: количество часов по типу (необязательное, целое число, рассчитывается на основе дисциплины) для уточнения распределения.

При вводе данных система автоматически проверяет ограничения: преподаватель привязывается только к одной кафедре, нагрузка не превышает 5 дисциплин на преподавателя, тип нагрузки уникален для дисциплины.

Выходные данные

Выходные данные генерируются в виде отчетов, списков, планов и статистики. Они доступны разным группам пользователей (отдел кадров — для кадрового состава; деканаты — для учебных планов и анализа; бухгалтерия — для расчета зарплаты). Форматы: таблицы, списки, PDF-отчеты или экспорт в Excel. Выходные данные основаны на запросах к базе данных и включают сортировку, фильтры и агрегацию.

1. Для кафедр:

- Штатное расписание кафедры: таблица с должностями, количеством ставок и занятymi позициями (например, "Профессор — 5 ставок, занято 4").
- Количество преподавателей на кафедре: общее число (например, "Всего 15 преподавателей") и breakdown по категориям (ассистенты, доценты и т.д.).
- Список кафедр с менее 2-х докторов наук (как в требованиях к отчетности).
- Статистика по кафедрам: подсчет количества преподавателей с разными учеными степенями (например, "Кандидаты наук — 8, Доктора наук — 3").

2. Для преподавателей:

- Список сотрудников (кадровый состав): полный список с ФИО, должностью, кафедрой, ученой степенью и званием, отсортированный по критериям (ученая степень, должность, кафедра).
- Самые молодые преподаватели на каждой кафедре (как в требованиях, с указанием возраста на основе даты рождения).
- Отчет по кадровому потенциалу: список преподавателей с указанием нагрузки (не более 5 дисциплин).
- Общее число человек на кафедре: включая всех привязанных преподавателей (для анализа загруженности).

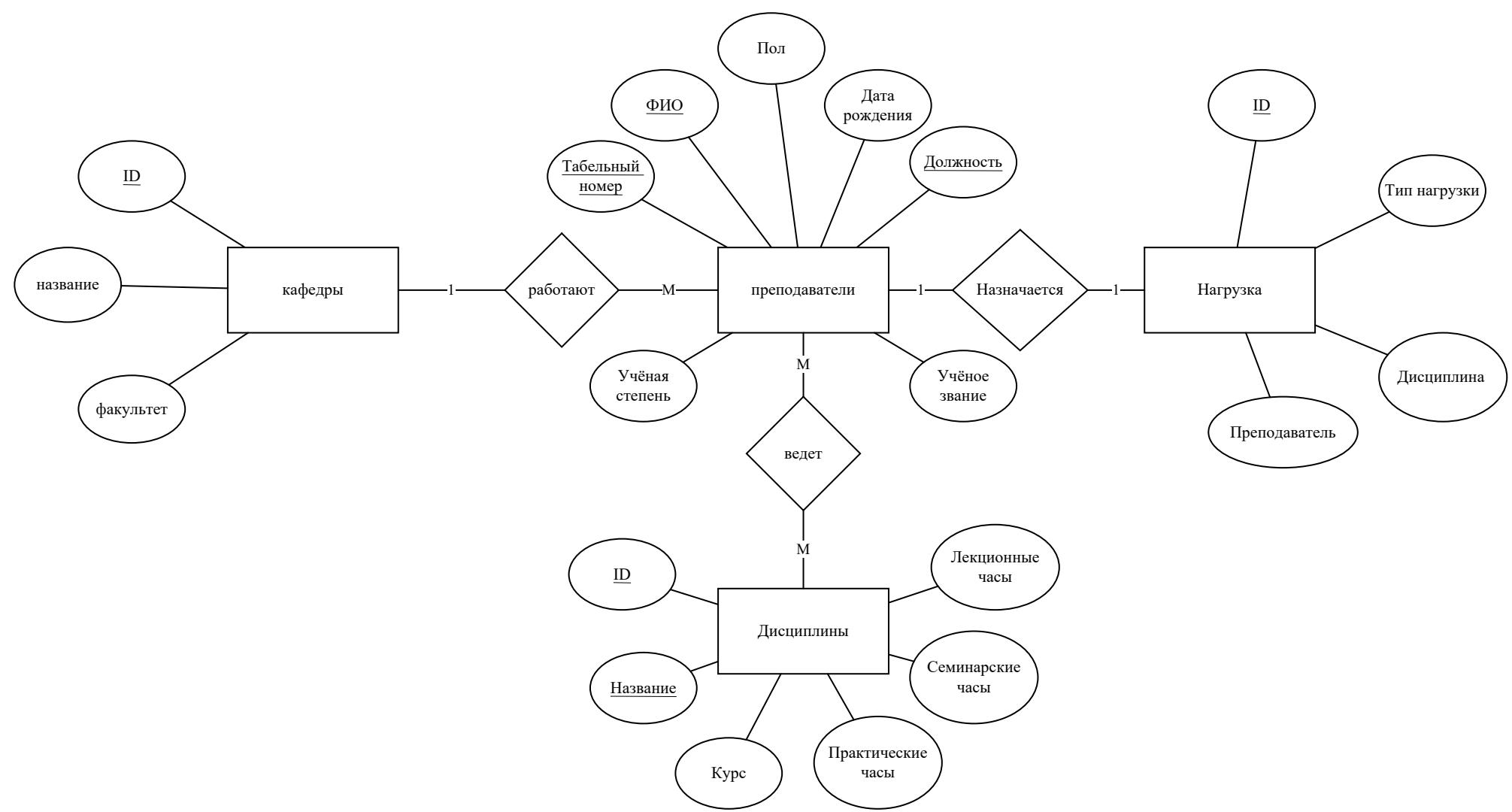
3. Для дисциплин:

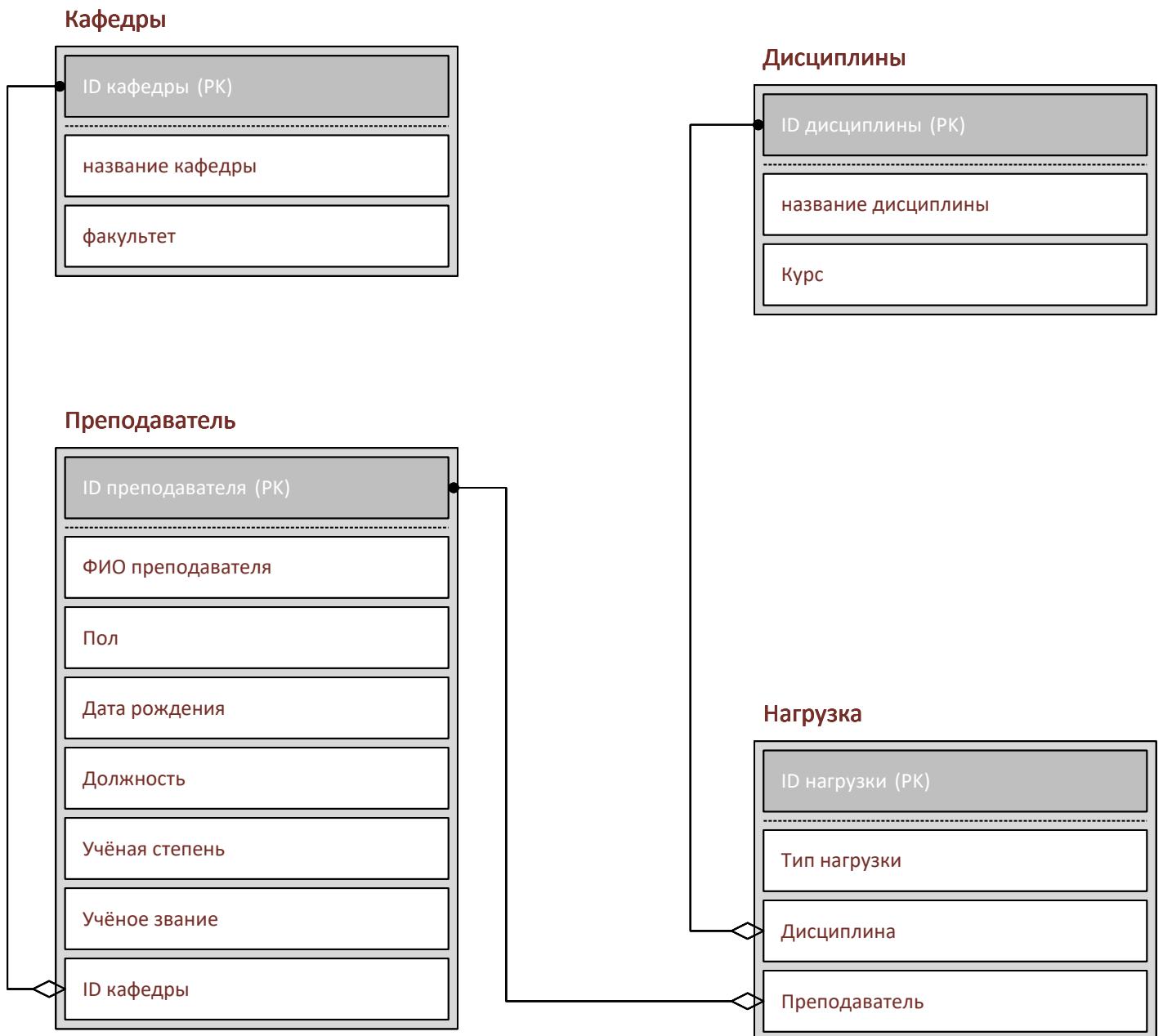
- Учебный план: таблица дисциплин по курсам и семестрам, с часами по типам (лекции, семинары, практики) и назначенными преподавателями.
- Список дисциплин без назначенных преподавателей: по типам нагрузки (как в требованиях к контролю назначения).
- Статистика по дисциплинам: общее количество часов по кафедрам или факультетам.
- Распределение дисциплин: список с указанием, сколько преподавателей ведут каждую (с учетом уникальности типа нагрузки).

4. Для нагрузки:

- Нагрузка по преподавателям: индивидуальный отчет с дисциплинами, типами нагрузки и часами (для деканатов и бухгалтерии, используется для расчета зарплаты).

- Нагрузка по кафедрам: агрегированный отчет с общим объемом часов (лекции, семинары, практики) и распределением по преподавателям.
- Контроль нагрузки: список перегруженных преподавателей (более 5 дисциплин) или неназначенных типов нагрузки.
- Учебный план с нагрузкой: интегрированный отчет, показывающий дисциплины с привязанными преподавателями и типами занятий.





```
USE hr_department_sch;

SET FOREIGN_KEY_CHECKS = 0;

DROP TABLE IF EXISTS Disciplines;

DROP TABLE IF EXISTS Workload;

DROP TABLE IF EXISTS Teachers;

DROP TABLE IF EXISTS Departments;

SET FOREIGN_KEY_CHECKS = 1;
```

```
CREATE TABLE Departments (
    department_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    faculty VARCHAR(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE Workload (
    workload_id INT AUTO_INCREMENT PRIMARY KEY,
    workload_type ENUM('Lectures','Seminars','Practical','Laboratory',
    'Consultations','Exams','CourseProjects','Other') NOT NULL,
    academic_year YEAR NOT NULL,
    semester TINYINT NOT NULL CHECK (semester IN (1,2)),
    assigned_hours INT NOT NULL DEFAULT 0
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE Disciplines (
    discipline_id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    course_year TINYINT NOT NULL,
    lecture_hours INT NOT NULL DEFAULT 0,
```

```
        seminar_hours INT NOT NULL DEFAULT 0,  
        practical_hours INT NOT NULL DEFAULT 0  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE TABLE Teachers (  
    personnel_number VARCHAR(20) PRIMARY KEY,  
    full_name VARCHAR(150) NOT NULL,  
    gender ENUM('M','F') NOT NULL,  
    birth_date DATE NOT NULL,  
    position VARCHAR(100) NOT NULL,  
    academic_degree VARCHAR(100),  
    academic_title VARCHAR(100),  
    department_id INT NOT NULL,  
    workload_id INT NULL,  
    discipline_id INT NULL,  
    CONSTRAINT fk_teacher_department  
        FOREIGN KEY (department_id) REFERENCES Departments(department_id)  
        ON DELETE RESTRICT ON UPDATE CASCADE,  
    CONSTRAINT fk_teacher_workload  
        FOREIGN KEY (workload_id) REFERENCES Workload(workload_id)  
        ON DELETE SET NULL ON UPDATE CASCADE,  
    CONSTRAINT fk_teacher_discipline  
        FOREIGN KEY (discipline_id) REFERENCES Disciplines(discipline_id)  
        ON DELETE SET NULL ON UPDATE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
CREATE INDEX idx_teacher_dept ON Teachers(department_id);  
CREATE INDEX idx_teacher_workload ON Teachers(workload_id);  
CREATE INDEX idx_teacher_discipline ON Teachers(discipline_id);
```