

HW 5: Data Warehouse Basics

Тема: Система высшего образования

1. Бизнес-процесс

Оценка и контроль качества обучения студентов по курсам и преподавателям в рамках учебных периодов оценки

2. Уровень детализации (grain)

Конкретная оценка, выставленная студенту за определённый курс, преподавателем, в рамках конкретного учебного периода

3. Определяем таблицы измерений (dimension tables)

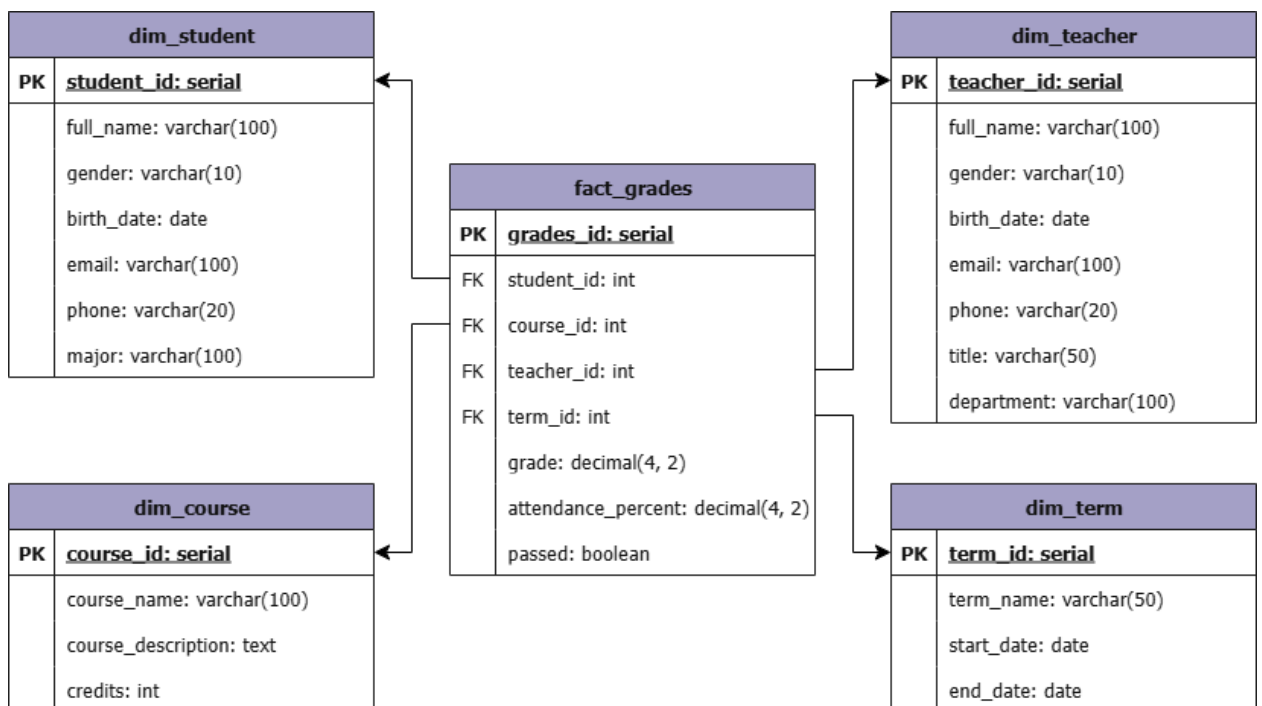
- dim_student: student_id, full_name, gender, birth_date, email, phone, major
- dim_course: course_id, course_name, course_description, credits
- dim_teacher: teacher_id, full_name, gender, email, phone, title, department
- dim_term: term_id, term_name, start_date, end_date

4. Таблица фактов (fact table)

fact_grades - фиксирует факты: оценки, посещаемость, прохождение курса:

- grades_id (PK)
- student_id (FK) – ссылка на студента
- course_id (FK) – ссылка на курс
- teacher_id (FK) – ссылка на преподавателя
- term_id (FK) – ссылка на семестр
- grade – оценка
- attendance_percent – процент посещаемости
- passed – значение прошел/не прошел

5. Физическая модель: Star



6. Создание таблиц. Заполнение. Аналитические запросы, которые отвечают на ключевые вопросы бизнес-процесса

Создание таблиц

```
--создание таблиц измерений

--студенты
create table dim_student (
    student_id SERIAL primary key,
    full_name VARCHAR(100) not null,
    gender VARCHAR(10),
    birth_date DATE,
    email VARCHAR(100),
    phone VARCHAR(20),
    major VARCHAR(100)
);

--преподаватели
create table dim_teacher(
    teacher_id SERIAL primary key,
    full_name VARCHAR(100) not null,
    gender VARCHAR(10),
    birth_date DATE,
    email VARCHAR(100),
    phone VARCHAR(20),
    title VARCHAR(50),
    department VARCHAR(100)
);

--курсы
create table dim_course (
    course_id SERIAL primary key,
    course_name VARCHAR(100) not null,
    course_description TEXT,
    credits INT check (credits >= 0)
);

--учебные периоды (семестры)
create table dim_term(
    term_id SERIAL primary key,
    term_name VARCHAR(50) not null,
    start_date DATE,
    end_date DATE
);

--таблица фактов
create table fact_grades (
    grades_id SERIAL primary key,
    student_id INT references dim_student(student_id),
    course_id INT references dim_course(course_id),
    teacher_id INT references dim_teacher(teacher_id),
    term_id INT references dim_term(term_id),
    grade DECIMAL(4,2) check (grade >= 0 and grade <= 10),
    attendance_percent DECIMAL(5,2) check (attendance_percent >= 0 and
attendance_percent <= 100),
    passed BOOLEAN
);
```

Заполнение таблиц

--заполнение таблиц

-- студенты

```
INSERT INTO dim_student (full_name, gender, birth_date, email, phone, major) VALUES
('Alice Johnson', 'Female', '2002-04-15', 'alice.johnson@example.com', '+1234567890',
'Computer Science'),
('Bob Smith', 'Male', '2001-09-30', 'bob.smith@example.com', '+1234567891', 'Data
Analytics'),
('Catherine Lee', 'Female', '2003-01-22', 'catherine.lee@example.com', '+1234567892',
'Information Systems'),
('David Kim', 'Male', '2002-07-08', 'david.kim@example.com', '+1234567893', 'Software
Engineering'),
('Eva Brown', 'Female', '2001-12-05', 'eva.brown@example.com', '+1234567894',
'Business Intelligence');
```

-- преподаватели

```
INSERT INTO dim_teacher (full_name, gender, birth_date, email, phone, title,
department) VALUES
('Dr. John Miller', 'Male', '1975-06-12', 'john.miller@university.edu',
'+1234567800', 'Professor', 'Computer Science'),
('Dr. Sarah Davis', 'Female', '1980-03-25', 'sarah.davis@university.edu',
'+1234567801', 'Associate Professor', 'Data Analytics'),
('Dr. Kevin Wilson', 'Male', '1985-11-10', 'kevin.wilson@university.edu',
'+1234567802', 'Lecturer', 'Information Systems'),
('Dr. Emily Clark', 'Female', '1978-08-18', 'emily.clark@university.edu',
'+1234567803', 'Senior Lecturer', 'Software Engineering'),
('Dr. Michael Scott', 'Male', '1982-02-02', 'michael.scott@university.edu',
'+1234567804', 'Professor', 'Business Intelligence');
```

-- курсы

```
INSERT INTO dim_course (course_name, course_description, credits) VALUES
('SQL Fundamentals', 'Learn basic SQL queries and relational concepts.', 3),
('Data Warehousing', 'Explore data warehouse architecture and ETL processes.', 4),
('Entity-Relationship Modeling', 'Design ER diagrams and normalize data.', 2),
('Advanced Analytics', 'Apply statistical methods to large datasets.', 3),
('Business Intelligence Tools', 'Use BI platforms for reporting and dashboards.', 3);
```

-- учебные периоды (семестры)

```
INSERT INTO dim_term (term_name, start_date, end_date) VALUES
('Spring 2025', '2025-02-01', '2025-06-01'),
('Fall 2025', '2025-09-01', '2025-12-20'),
('Winter 2024', '2024-01-10', '2024-04-30'),
('Summer 2024', '2024-06-15', '2024-08-30'),
('Spring 2024', '2024-02-01', '2024-05-30');
```

-- таблица фактов

```
INSERT INTO fact_grades (student_id, course_id, teacher_id, term_id, grade,
attendance_percent, passed) VALUES
(1, 1, 1, 1, 8.50, 95.00, TRUE),
(2, 2, 2, 2, 7.20, 88.50, TRUE),
(3, 3, 3, 3, 6.00, 70.00, FALSE),
(4, 4, 4, 4, 9.10, 98.00, TRUE),
(5, 5, 5, 5, 5.50, 65.00, FALSE);
```

Аналитические запросы и вопросы, на которые они отвечают

1. Средняя оценка по каждому курсу

Вопрос: какие курсы показывают высокую или низкую успеваемость?

Скрипт:

```
-- 1. средняя оценка по каждому курсу
```

```
select
    c.course_name,
    round(AVG(f.grade), 2) as average_grade
from fact_grades f
join dim_course c on f.course_id = c.course_id
group by c.course_name
order by average_grade desc;
```

Результат:

AZ course_name	123 average_grade
Advanced Analytics	9,1
SQL Fundamentals	8,5
Data Warehousing	7,2
Entity-Relationship Modeling	6
Business Intelligence Tools	5,5

Описание: этот запрос показывает, какие курсы дают студентам более высокие или низкие оценки. Он помогает выявить сложные дисциплины или те, где преподаватели ставят более строгие оценки. ROUND используется для округления среднего значения до двух знаков после запятой.

2. Процент успешно сдавших студентов по каждому преподавателю

Вопрос: какие преподаватели демонстрируют высокую результативность?

Скрипт:

```
-- 2. процент успешно сдавших студентов по каждому преподавателю
select
    t.full_name as teacher,
    COUNT(*) as total_students,
    COUNT(case when f.passed then 1 end) * 100.0 / COUNT(*) as pass_rate_percent
from fact_grades f
join dim_teacher t on f.teacher_id = t.teacher_id
group by t.full_name
order by pass_rate_percent desc;
```

Результат:

AZ teacher	123 total_students	123 pass_rate_percent
Dr. Sarah Davis	1	100
Dr. Emily Clark	1	100
Dr. John Miller	1	100
Dr. Kevin Wilson	1	0
Dr. Michael Scott	1	0

Описание: этот запрос оценивает эффективность преподавателей: сколько студентов успешно прошли курсы у каждого из них. CASE WHEN f.passed THEN 1 END считает только тех, кто прошёл курс, а затем вычисляется процент от общего числа.

3. Посещаемость студентов по семестрам

Вопрос: как меняется вовлечённость студентов по учебным периодам?

Скрипт:

```
-- 3. посещаемость студентов по семестрам
select
```

```
term.term_name,
term.start_date,
ROUND(AVG(f.attendance_percent), 2) as avg_attendance
from fact_grades f
join dim_term term ON f.term_id = term.term_id
group by term.term_name, term.start_date
order by term.start_date;
```

Результат:

AZ term_name	start_date	avg_attendance
Winter 2024	2024-01-10	70
Spring 2024	2024-02-01	65
Summer 2024	2024-06-15	98
Spring 2025	2025-02-01	95
Fall 2025	2025-09-01	88,5

Описание: позволяет отследить, как меняется средняя посещаемость студентов от семестра к семестру. Это может быть полезно для выявления сезонных спадов вовлечённости или влияния внешних факторов (например, экзаменационных периодов).

4. Студенты, не сдавшие хотя бы один курс

Вопрос: кто из студентов нуждается в академической поддержке?

Скрипт:

```
-- 4. студенты, не сдавшие хотя бы один курс
select
s.full_name,
COUNT(*) as failed_courses
from fact_grades f
join dim_student s on f.student_id = s.student_id
where f.passed = false
group by s.full_name
order by failed_courses desc;
```

Результат:

AZ full_name	failed_courses
Eva Brown	1
Catherine Lee	1

Описание: выявляет студентов, у которых есть проблемы с прохождением курсов. Это может быть основой для академической поддержки, индивидуальных консультаций или корректировки учебной нагрузки.

5. Самые популярные курсы (по количеству записей)

Вопрос: Какие курсы выбирают чаще всего?

Скрипт:

```
-- 5. самые популярные курсы (по количеству записей)
select
c.course_name,
COUNT(*) as enrollment_count
from fact_grades f
join dim_course c on f.course_id = c.course_id
group by c.course_name
```

```
order by enrollment_count desc;
```

Результат:

AZ course_name	123 enrollment_count
Business Intelligence Tools	1
SQL Fundamentals	1
Data Warehousing	1
Advanced Analytics	1
Entity-Relationship Modeling	1

Описание: показывает, какие курсы выбираются чаще всего. Это может быть полезно для планирования расписания, распределения преподавателей и оценки интересов студентов.