



Факультет программной инженерии и компьютерной техники

Информационные системы и базы данных

Лабораторная работа № 1

Вариант 1134

Преподаватель: Харитонова А. Е.

Выполнила: Блинова М.А.

Группа: Р33301

Текст задания:

На сей раз воздействие это было быстрым и уверенным, ткался новый узор. За многие эры, минувшие с предыдущей встречи, ткач многое узнал; и материал, на котором испытывал он ныне свое искусство, стал несравнимо совершеннее. Но достоин ли этот материал стать частью все еще разрастающейся ткани, покажет лишь грядущее.

(узор делается на материале и если он идеальный, то получается ткань)

Описание предметной области:

ТКАЧ (id, имя, фамилия, возраст, стаж работы) изготавливает **УЗОРЫ** (id, **ТИП_УЗОРА** - античный/ готический/ барочный; **ЦВЕТ** - разноцветный, однотонный). То, какой узор и материал сделал какой ткач(и), фиксируется **АВТОРСТВОМ**. Узоры вышиваются на **МАТЕРИАЛЕ** (id, название материала, совершенный/ несовершенный, достойный/ недостойный, цвет) **ТКАНИ** (id, **ТИП_ТКАНИ**, размер (м²), материал, узор), которая может быть изготовлена только из идеального материала.

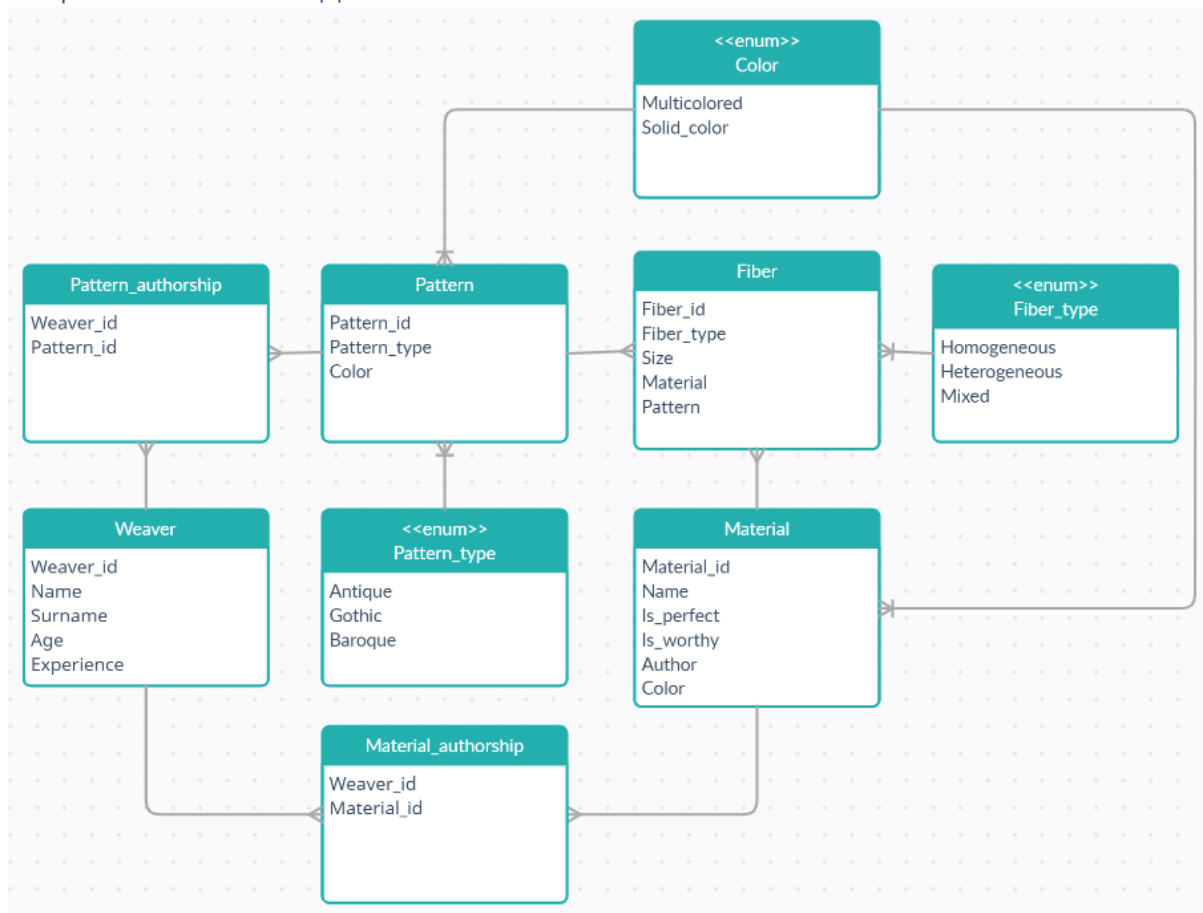
Классификация сущностей:

Характеристические: Pattern_type, Color, Fiber_type

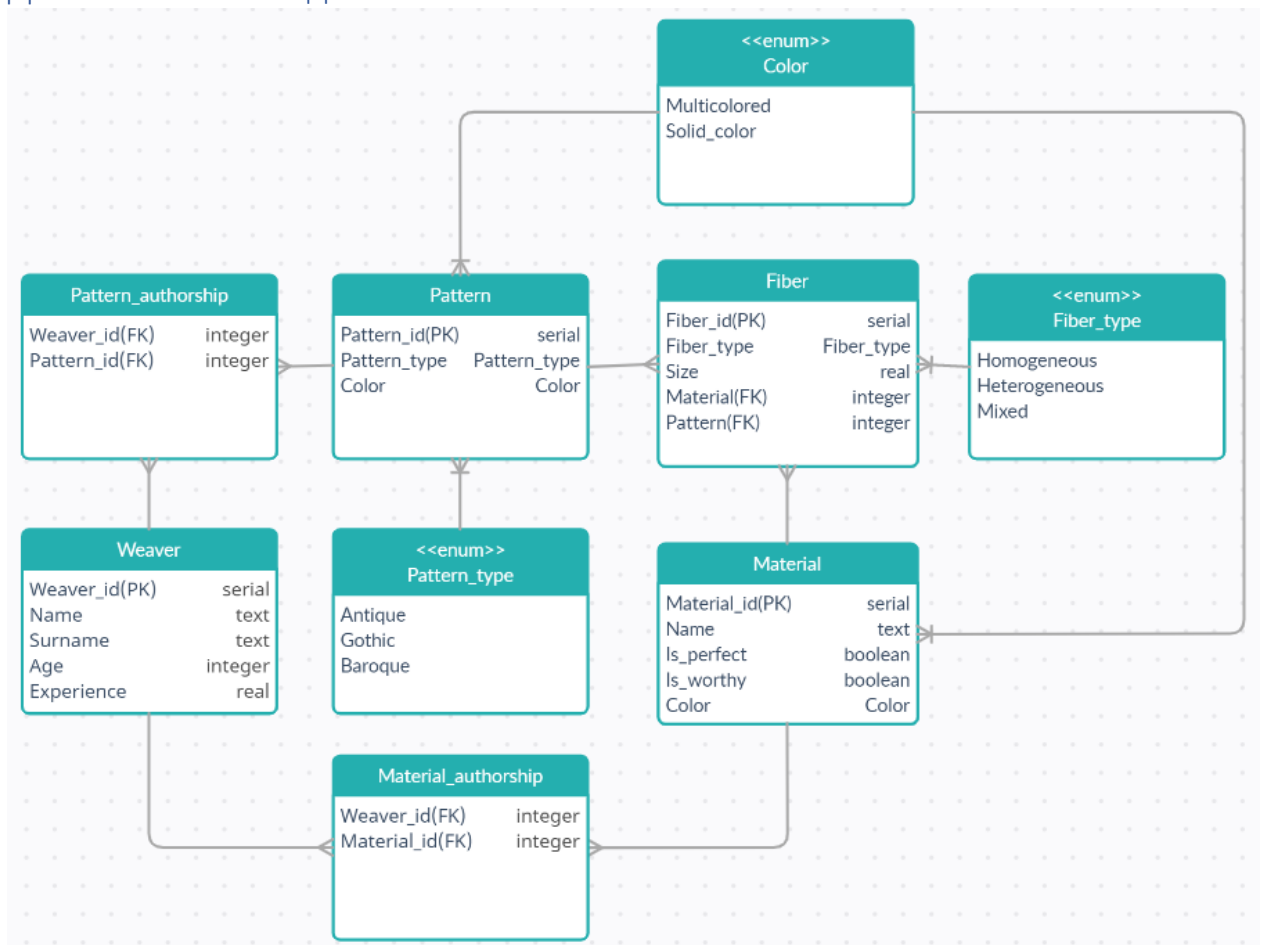
Ассоциативные: Pattern_authorship, Material_authorship.

Стержневые: Weaver, Pattern, Fiber, Material

Инфологическая модель:



Даталогическая модель:



Реализация даталогической модели на SQL:

Характеристические:

```
CREATE TYPE color AS ENUM ('multicolored', 'solid_color');
```

```
CREATE TYPE fiber_type AS ENUM ('homogeneous', 'heterogeneous', 'mixed');
```

```
CREATE TYPE pattern_type AS ENUM ('antique', 'gothic', 'baroque');
```

Стержневые:

```
CREATE TABLE weaver (
    weaver_id SERIAL PRIMARY KEY,
    name TEXT NOT NULL,
    surname TEXT NOT NULL,
    age INTEGER NOT NULL CHECK (age > 0),
    experience REAL NOT NULL DEFAULT 0 CHECK (experience >= 0)
);
```

```
CREATE TABLE pattern (  
    pattern_id SERIAL PRIMARY KEY,  
    pattern_type PATTERN_TYPE NOT NULL,  
    color COLOR NOT NULL  
);
```

```
CREATE TABLE material (  
    material_id SERIAL PRIMARY KEY,  
    name TEXT NOT NULL,  
    is_perfect BOOLEAN NOT NULL DEFAULT FALSE,  
    is_worthy BOOLEAN NOT NULL DEFAULT FALSE,  
    color COLOR NOT NULL  
);
```

```
CREATE TABLE fiber (  
    fiber_id SERIAL PRIMARY KEY,  
    fiber_type FIBER_TYPE NOT NULL,  
    size REAL NOT NULL CHECK (size > 0),  
    material INTEGER REFERENCES material ON DELETE CASCADE,  
    pattern INTEGER REFERENCES pattern ON DELETE CASCADE  
);
```

Ассоциативные:

```
CREATE TABLE pattern_authorship (  
    weaver_id INTEGER REFERENCES weaver ON DELETE RESTRICT NOT NULL,  
    pattern_id INTEGER REFERENCES pattern ON DELETE CASCADE NOT NULL  
);
```

```
CREATE TABLE material_authorship (  
    weaver_id INTEGER REFERENCES weaver ON DELETE RESTRICT NOT NULL,  
    material_id INTEGER REFERENCES material ON DELETE CASCADE NOT NULL  
);
```

Заполнение данными:

```
INSERT INTO weaver (name, surname, age, experience) VALUES ('Leo', 'Gayler', 18, 1.0);
```

```
INSERT INTO weaver (name, surname, age, experience) VALUES ('Oleg', 'Kuvosh', 300, 280);
```

```
INSERT INTO pattern (pattern_type, color) VALUES ('antique', 'solid_color');
```

```
INSERT INTO pattern (pattern_type, color) VALUES ('baroque', 'multicolored');
```

```
INSERT INTO material (name, is_perfect, is_worthy, color) VALUES ('cotton', true, true, 'multicolored');
```

```
INSERT INTO material (name, is_perfect, is_worthy, color) VALUES ('calico', false, false, 'multicolored');
```

```
INSERT INTO fiber (fiber_type, size, material, pattern) VALUES ('homogeneous', 0.02, 1, 1);
```

```
INSERT INTO fiber (fiber_type, size, material, pattern) VALUES ('mixed', 4, 2, 2);
```

```
INSERT INTO pattern_authorship (weaver_id, pattern_id) VALUES (1, 1);
```

```
INSERT INTO pattern_authorship (weaver_id, pattern_id) VALUES (2, 2);
```

```
INSERT INTO material_authorship (weaver_id, material_id) VALUES (1, 1);
```

```
INSERT INTO material_authorship (weaver_id, material_id) VALUES (2, 2);
```

Вывод:

Архитектура ANSI-SPARC определяет принцип, согласно которому рекомендуется строить системы управления базами данных (СУБД). Выделяют 3 уровня системы:

1. Внешний (пользовательский)
2. Промежуточный (концептуальный)
3. Внутренний (физический).

В основе архитектуры ANSI-SPARC лежит концептуальный уровень. В современных СУБД он может быть реализован при помощи представления. Концептуальный уровень описывает данные и их взаимосвязи с наиболее общей точки зрения, — концепции архитекторов базы, используя реляционную или другую модель.

ER-модель (от англ. Entity-Relationship model, модель «сущность — связь») — модель данных, позволяющая описывать концептуальные схемы предметной области. ER-модель используется при высокоуровневом (концептуальном) проектировании баз данных. С её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между этими сущностями.

Классификация сущностей:

Стержневая сущность — это независимая сущность

Ассоциативная сущность (ассоциация) — это связь вида "многие-ко-многим" между сущностями или экземплярами сущности. Ассоциации рассматриваются как полноправные сущности

Характеристическая сущность (характеристика) — это связь вида "многие-к-одной" или "одна-к-одной" между двумя сущностями (частный случай ассоциации). Единственная цель характеристики в рамках рассматриваемой предметной области состоит в описании или уточнении некоторой другой сущности.

Обозначающая сущность или обозначение — это связь вида "многие-к-одной" или "одна-к-одной" между двумя сущностями и отличается от характеристики тем, что не зависит от обозначаемой сущности.

Запрос (доп. задание):

Нужно увидеть авторов недостойных материалов:

```
SELECT name, surname FROM weaver WHERE weaver_id = (  
    SELECT weaver_id FROM material_authorship WHERE material_id = (  
        SELECT material_id FROM material WHERE is_worthy = false  
    )  
);
```