

Лабораторное занятие №7

Инвариантная часть:

В предметной области, предложенной преподавателем, реализовать реляционную модель данных в MySQL Workbench. Спроектированная таблица должна отображать связи между сущностями, потенциальные ключи, условия проверки атрибутов.

Задания:

1. Определите потенциальные ключи для каждого из отношений и определите, какие из атрибутов отношений являются обязательными.

ИГРА

название id PK, NN	производитель PK, NN	мин игроков NN	макс игроков NN	мин возраст
-----------------------	-------------------------	-------------------	--------------------	-------------

КЛИЕНТ

id клиента PK, NN	имя NN	фамилия я	почта NN	дата рождения	телефон	адрес
----------------------	-----------	--------------	-------------	------------------	---------	-------

ЗАКАЗ

id заказа PK, NN	id клиента NN	id сотрудника NN	производитель NN	кол. шт NN	название NN	время создания NN	способ получения NN
------------------------	---------------------	------------------------	---------------------	---------------	----------------	-------------------------	---------------------------

СОТРУДНИК

id сотрудника PK, NN	фамилия NN	имя NN
-------------------------	---------------	-----------

3. Определите условия проверки значений для атрибутов.

Таблица ИГРА:

Мин игроков ≥ 1

Таблица КЛИЕНТ:

Ид_клиента > 0

Имя LIKE '%[A-Za-z]%'

Почта LIKE '%@%.%'

Таблица ЗАКАЗ:

Номер > 0

Получение = 'доставка' OR Получение = 'самовывоз'

Таблица СОТРУДНИК:

Таб. Номер > 0

Имя LIKE '%[A-Za-z]%'

4. Определите связи между отношениями.

ИГРА(title_id, manufacturer) - ЗАКАЗ(title, manufacturer), один заказ может включать несколько игр.

КЛИЕНТ(client_id) - ЗАКАЗ(client_id), один клиент может сделать любое кол. заказов.

СОТРУДНИК(worker_id) - ЗАКАЗ(workers), один сотрудник может определять любое кол. заказов.

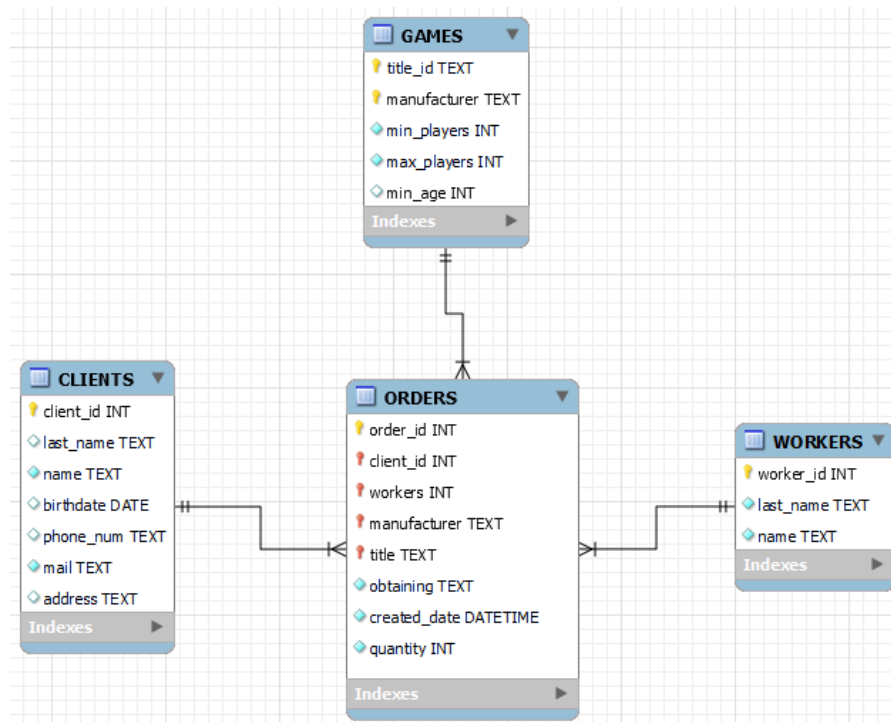
7. Ответьте на вопрос: Допустим, что каждый сотрудник может обработать только один заказ (после чего его увольняют). В этом случае, связь между отношениями ЗАКАЗ и СОТРУДНИК трансформируется в один-к-одному. Что нужно будет сделать дополнительно для поддержки этой связи? Что может помешать указать одного и того же сотрудника в отношении ЗАКАЗ, ответственным за обработку нескольких разных заказов?

1. В отношении ЗАКАЗ добавить внешний ключ, который ссылается на первичный ключ отношения СОТРУДНИК.
2. В отношении СОТРУДНИК добавить ограничение уникальности на внешний ключ, чтобы гарантировать, что каждый сотрудник может быть связан только с одним заказом.

за обработку нескольких разных заказов, может быть помешано следующими фактором:

1. Удаление записей.

ERR таблица:



Запросы по созданию:

-- MySQL Workbench Forward Engineering

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DI
VISION_BY_ZERO,NO_ENGINE_SUBSTITUTION';
```

-- Schema mydb

-- Schema LW

-- Schema LW

```
CREATE SCHEMA IF NOT EXISTS `LW` DEFAULT CHARACTER SET utf8mb3 ;
USE `LW` ;
```

-- Table `LW`.`CLIENTS`

```
CREATE TABLE IF NOT EXISTS `LW`.`CLIENTS` (
  `client_id` INT NOT NULL,
  `last_name` TEXT NULL,
  `name` TEXT NOT NULL,
  `birthdate` DATE NULL,
  `phone_num` TEXT NULL,
```

```
`mail` TEXT NOT NULL,  
`address` TEXT NULL,  
PRIMARY KEY (`client_id`))  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb3;
```

```
-----  
-- Table `LW`.`WORKERS`  
-----
```

```
CREATE TABLE IF NOT EXISTS `LW`.`WORKERS` (  
  `worker_id` INT NOT NULL,  
  `last_name` TEXT NOT NULL,  
  `name` TEXT NOT NULL,  
  PRIMARY KEY (`worker_id`))  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb3;
```

```
-----  
-- Table `LW`.`GAMES`  
-----
```

```
CREATE TABLE IF NOT EXISTS `LW`.`GAMES` (  
  `title_id` TEXT NOT NULL,  
  `manufacturer` TEXT NOT NULL,  
  `min_players` INT NOT NULL,  
  `max_players` INT NOT NULL,  
  `min_age` INT NULL,  
  PRIMARY KEY (`title_id`, `manufacturer`))  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb3;
```

```
-----  
-- Table `LW`.`ORDERS`  
-----
```

```
CREATE TABLE IF NOT EXISTS `LW`.`ORDERS` (  
  `order_id` INT NOT NULL,  
  `client_id` INT NOT NULL,  
  `workers` INT NOT NULL,  
  `manufacturer` TEXT NOT NULL,  
  `title` TEXT NOT NULL,  
  `obtaining` TEXT NOT NULL,  
  `created_date` DATETIME NOT NULL,  
  `quantity` INT NOT NULL,  
  PRIMARY KEY (`order_id`, `client_id`, `workers`, `manufacturer`, `title`),  
  INDEX `order_to_client_idx` (`client_id` ASC) VISIBLE,  
  INDEX `order_to_emp_idx` (`workers` ASC) VISIBLE,  
  INDEX `ORDERS_games_idx` (`manufacturer` ASC, `title` ASC) VISIBLE,  
  CONSTRAINT `ORDERS_to_client`  
    FOREIGN KEY (`client_id`)  
    REFERENCES `LW`.`CLIENTS` (`client_id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE,  
  CONSTRAINT `ORDERS_to_workers`  
    FOREIGN KEY (`workers`)  
    REFERENCES `LW`.`WORKERS` (`worker_id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE,  
  CONSTRAINT `ORDERS_to_games`  
    FOREIGN KEY (`manufacturer`, `title`)  
    REFERENCES `LW`.`GAMES` (`manufacturer`, `title_id`)  
    ON DELETE CASCADE  
    ON UPDATE CASCADE)  
ENGINE = InnoDB  
DEFAULT CHARACTER SET = utf8mb3;
```

```
SET SQL_MODE=@OLD_SQL_MODE;  
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;  
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

Запросы по добавлению данных:

```
INSERT INTO `LW`.`CLIENTS` (`client_id`, `last_name`, `name`, `birthdate`, `phone_num`, `mail`, `address`) VALUES (1,  
'Иванов', 'Олег', '21.02.2000', '+7923233', 'test1@mail.ru', 'пушкина д2');
```

```
INSERT INTO `LW`.`CLIENTS` (`client_id`, `last_name`, `name`, `birthdate`, `phone_num`, `mail`, `address`) VALUES (2,  
'Круглов', 'Миша', '22.03.2001', '+7923234', 'test2@mail.ru', 'птушкина д3');
```

```
INSERT INTO `LW`.`ORDERS` (`order_id`, `client_id`, `workers`, `manufacturer`, `title`, `obtaining`, `created_date`, `quantity`) VALUES (1, 1, 1, 'valve', 'tetris', 'самовывоз', '21.02.2003', 1);
```

```
INSERT INTO `LW`.`ORDERS` (`order_id`, `client_id`, `workers`, `manufacturer`, `title`, `obtaining`, `created_date`, `quantity`) VALUES (2, 1, 1, 'valve', 'antitetris', 'доставка', '21.02.2003', 2);
```

```
INSERT INTO `LW`.`ORDERS` (`order_id`, `client_id`, `workers`, `manufacturer`, `title`, `obtaining`, `created_date`, `quantity`) VALUES (3, 2, 2, 'ubisoft', 'raft', 'доставка', '12.03.2005', 1);
```

```
INSERT INTO `LW`.`ORDERS` (`order_id`, `client_id`, `workers`, `manufacturer`, `title`, `obtaining`, `created_date`, `quantity`) VALUES (4, 2, 2, 'ubisoft', 'cup', 'доставка', '12.03.2005', 3);
```

```
INSERT INTO `LW`.`WORKERS` (`worker_id`, `last_name`, `name`) VALUES (1, 'Дима', 'Васильев');
```

```
INSERT INTO `LW`.`WORKERS` (`worker_id`, `last_name`, `name`) VALUES (2, 'Максим', 'Пупкин');
```

```
INSERT INTO `LW`.`GAMES` (`title_id`, `manufacturer`, `min_players`, `max_players`, `min_age`) VALUES ('tetrisO', 'valve', 1, 5, 6);
```

```
INSERT INTO `LW`.`GAMES` (`title_id`, `manufacturer`, `min_players`, `max_players`, `min_age`) VALUES ('raftCUP', 'ubisoft', 1, 4, 18);
```