

Міністерство освіти і науки України

Харківський національний університет імені В.Н. Каразіна
Кафедра теоретичної та прикладної інформатики

Звіт по дисципліні
Вступ до SQL баз даних

Додадкове завдання № 2

Студента: Чистякова Артема
Групи: МФ-31

Необхідний термін здачі завдання: _____

Фактичний термін здачі завдання: _____

Кількість балів: _____

Харків 2020

Постановка задачи

Разработать базу данных для хранения и обработки информации о внутреннем устройстве некоторой небольшой аутсорс IT-компании.

Компания хоть и небольшая, но может располагать несколькими филиалами. В каждом из офисов работают: менеджеры, программисты и дизайнеры. У каждого менеджера в подчинении могут находиться другие менеджеры, а также программисты и дизайнеры. У каждого программиста может быть программист ментор высшей должности.

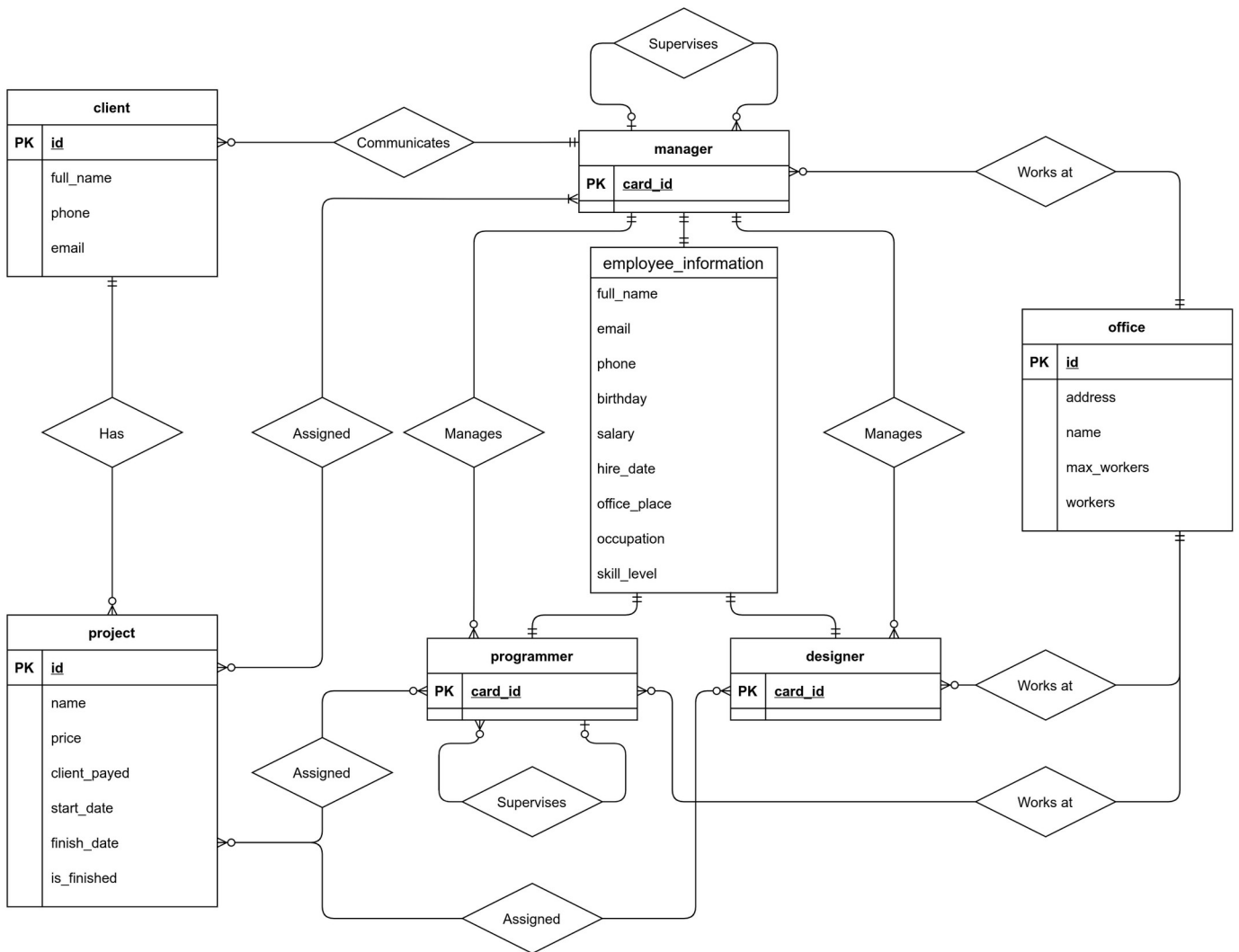
В компанию поступают заказы от клиентов в качестве проектов. Каждому клиенту предоставляется менеджер для общения, и над поступившими проектами начинается работа. В проекте могут быть задействованы менеджеры, программисты и дизайнеры.

В базе данных должна храниться информация о каждом сотруднике, это личные данные, зарплата, место в офисе и направление работы (для программиста, например, Java или C++).

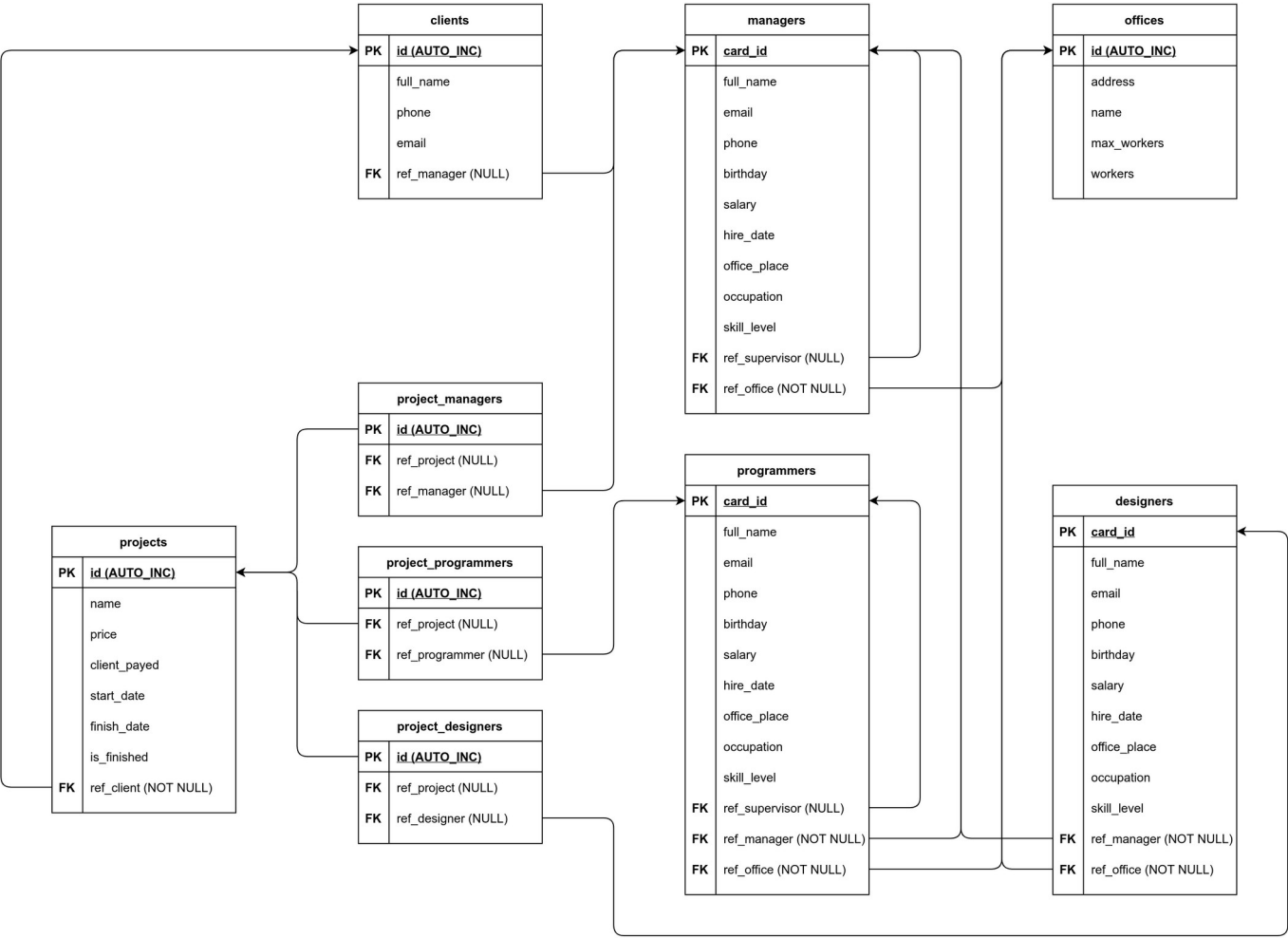
О проектах в базе данных тоже должна храниться информация, это стоимость проекта, текущая оплата клиента, время начала работы и дедлайн.

В базе данных также должна храниться информация о офисе.

Концептуальная модель БД



Представление БД в качестве таблиц и связей



Перечень запросов

1. Создать индексы для полей hire_date таблиц managers, designers и programmers.

Зачем?

Для ускорения сортировки по этим полям. Данные этих таблиц не будут очень часто изменяться, что не сильно скажется на производительности.

```
CREATE INDEX manag_hire ON managers(hire_date);
CREATE INDEX prog_hire ON programmers(hire_date);
CREATE INDEX design_hire ON designers(hire_date);
```

2. Создать индексы для полей name и full_name всех таблиц.

Зачем?

Для ускорения запросов на выборку по этим полям. Довольно часто возникает необходимость получить информацию о сотруднике (или проекте) зная только имя.

```
CREATE INDEX manag_name ON managers(full_name);
CREATE INDEX prog_name ON programmers(full_name);
CREATE INDEX design_name ON designers(full_name);
CREATE INDEX cli_name ON clients(full_name);
CREATE INDEX prog_name ON projects(name);
CREATE INDEX office_name ON offices(name);
```

Продemonстрирую работу индекса на таблица managers.

```
EXPLAIN SELECT full_name, email, phone
FROM managers WHERE full_name = "Abraham Lincoln";
```

| id | select_type | table | partitions | type | possible_keys | key | key_len | ref | rows | filtered | Extra |
|----|-------------|----------|------------|------|---------------|------|---------|------|------|----------|-------------|
| 1 | SIMPLE | managers | null | ALL | null | null | null | null | 5 | 20 | Using where |

Без индекса количество затронутых строк таблицы = 5.

| id | select_type | table | partitions | type | possible_keys | key | key_len | ref | rows | filtered | Extra |
|----|-------------|----------|------------|------|---------------|------------|---------|-------|------|----------|-------|
| 1 | SIMPLE | managers | null | ref | manag_name | manag_name | 403 | const | 1 | 100 | null |

С индексом количество затронутых строк таблицы = 1.

3. Можно было бы создать индексы для полей, по которым часто происходит объединение, но в данной базе данных такие поля – `foreign_key`, а `mysql` создает для них индексы автоматически.