

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

Індивідуальне завдання № 3

Студента: Чистякова Артема
Групи: МФ-31

Необхідний термін здачі завдання: 29.10.2020

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.

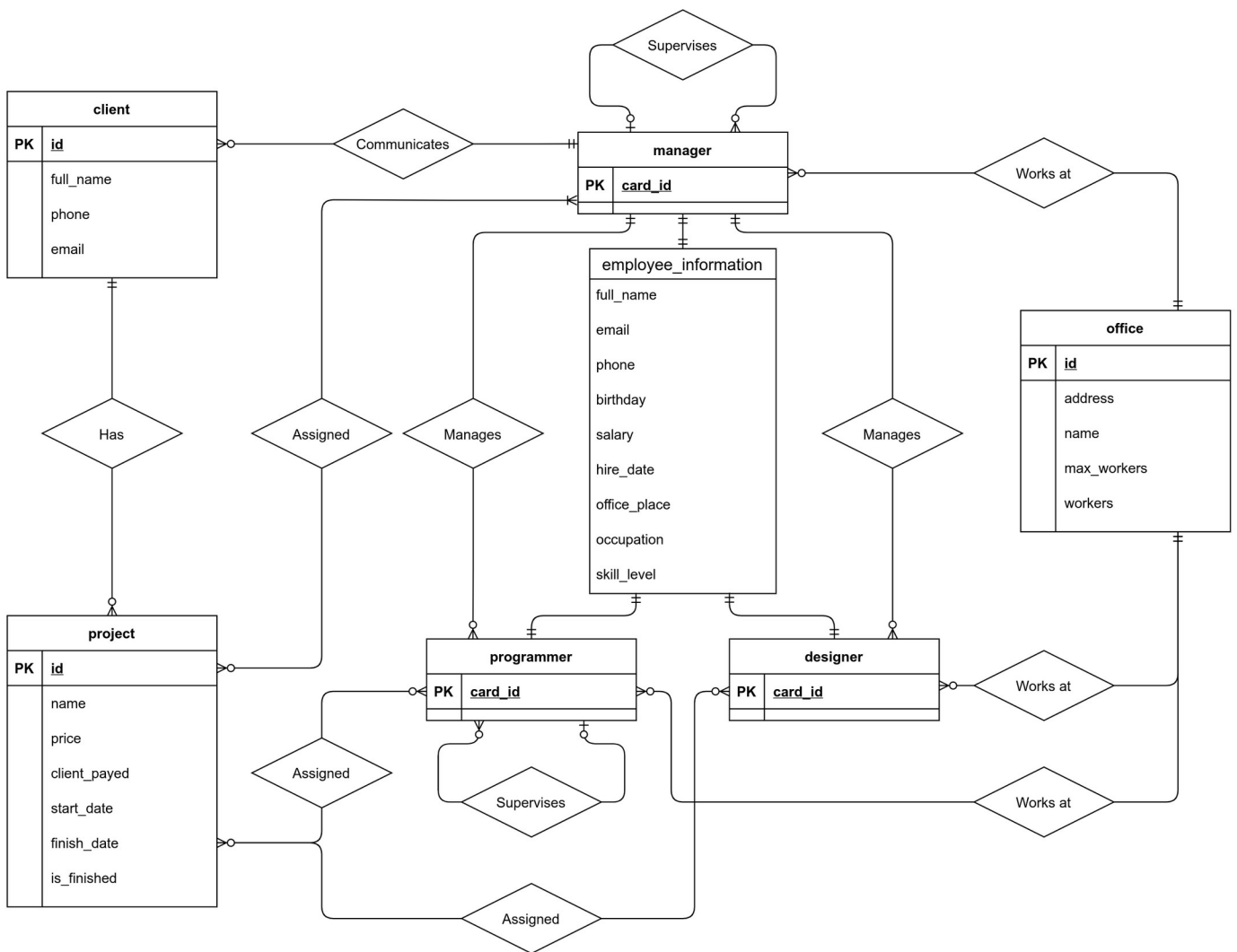
В компанию поступают заказы от клиентов в качестве проектов. Каждому клиенту предоставляется менеджер для общения, и над поступившими проектами начинается работа. В проекте могут быть задействованы менеджеры, программисты и дизайнеры.

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++).

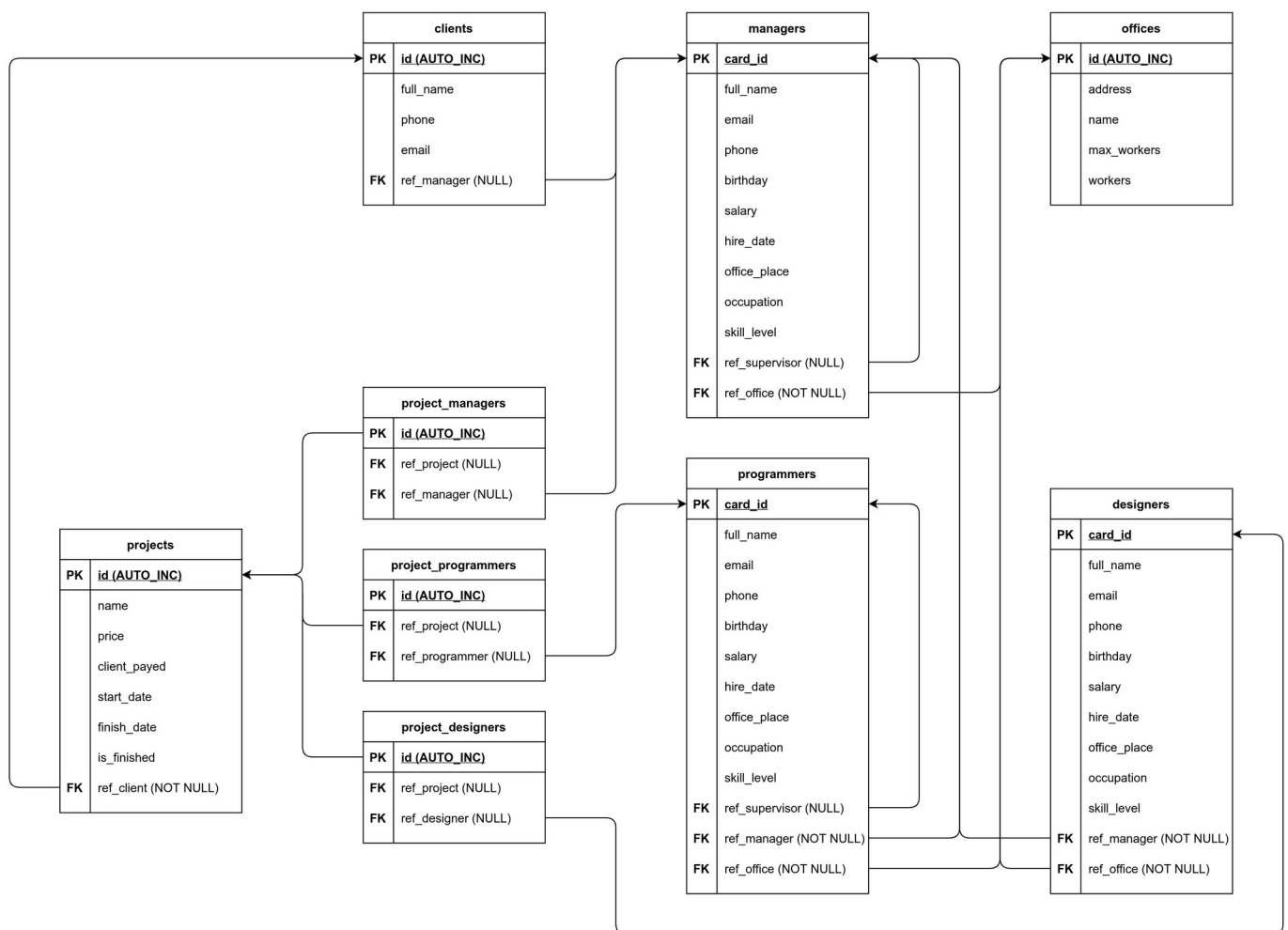
О проектах в базе данных тоже должна храниться информация, это стоимость проекта, текущая оплата клиента, время начала работы и дедлайн.

В базе данных также должна храниться информация о офисе.

Концептуальная модель БД



Представление БД в качестве таблиц и связей



Перечень запросов на выборку

1. Вывести максимальную, минимальную и среднюю цену всех проектов.

```
SELECT max(price) AS "MAX", min(price) AS "MIN", avg(price) AS "AVG"
FROM projects;
```

2. Вывести количество программистов, работающих в каждом из офисов.

```
SELECT COUNT(*) AS "number", name
FROM offices, programmers
WHERE programmers.ref_office = offices.id
GROUP BY (offices.name);
```

3. Для каждого менеджера вывести количество клиентов, с которыми он работает.

```
SELECT COUNT(*) AS "number", managers.full_name
FROM managers, clients
WHERE clients.ref_manager = managers.card_id
GROUP BY managers.full_name;
```

4. Вывести количество разных должностей менеджеров.

```
SELECT COUNT(DISTINCT occupation) AS "occupations"
FROM managers;
```

5. Вывести количество программистов, работающих над проектами с ценой > 10000.

```
SELECT COUNT(*) AS "number", projects.name
FROM project_programmers
INNER JOIN projects ON project_programmers.ref_project = projects.id
INNER JOIN programmers ON
project_programmers.ref_programmer = programmers.card_id
WHERE projects.price > 10000
GROUP BY projects.name;
```

6. Вывести название проектов, над которыми работает более чем 1 дизайнер.

```
SELECT projects.name
FROM project_designers
INNER JOIN projects ON project_designers.ref_project =
projects.id
INNER JOIN designers ON project_designers.ref_designer =
designers.card_id
GROUP BY projects.name
HAVING COUNT(*) > 1;
```

7. Вывести имена менеджеров, у которых в подчинении более чем 1 программист.

```
SELECT managers.full_name
FROM managers
INNER JOIN programmers ON programmers.ref_manager =
managers.card_id
GROUP BY managers.full_name
HAVING COUNT(*) > 1;
```

8. Вывести количество программистов, у которых есть хотя бы один подчинённый.

```
SELECT count(DISTINCT pr.full_name)
FROM programmers
INNER JOIN programmers AS pr ON programmers.ref_supervisor =
pr.card_id;
```

9. Вывести суммарную зарплату дизайнеров, работающих над проектом «Google.com».

```
SELECT sum(designers.salary) AS "designers salary"
FROM project_designers
INNER JOIN projects ON project_designers.ref_project =
projects.id
INNER JOIN designers ON project_designers.ref_designer =
designers.card_id
WHERE projects.name = 'Google.com';
```

10. Вывести средний возраст всех менеджеров.

```
SELECT floor(avg(year(current_date()) -
year(birthday) -
```

```
(DATE_FORMAT(current_date(), '%m%d') < DATE_FORMAT(birthday,  
'%m%d')))) AS "avg age"  
FROM managers;
```