

Дается описание предметной области, необходимо для нее описать структуру БД и прикладное решение с ее использованием

Предметная область - рыболовецкая компания, прикладное решение - приложение с UI для работников для учета разного рода деятельности компании

Требуется придумать структуру таблиц и описать на PL/SQL API, которое можно было бы вызывать для различных компонент UI

Структура таблиц - требуется описать таблицы необходимость которых явно следует из описания ниже, но также можно придумать дополнительные сущности и описать таблицы для них

Допустимо ограничиваться лишь своим воображением

*“Есть рыболовецкая компания, занимающаяся промыслом рыбы в регионе. Она имеет некоторое количество различных кораблей и свои собственные ремонтные доки*

*У компании в штате есть рыбаки и обслуживающий корабли персонал, а также некоторое кол-во сотрудников занимающихся наймом, бухгалтерией и закупками*

*Компании нужно вести учет кораблей, их тех. обслуживания и выходов в море, учет персонала и возможных для такой компании закупок, etc.”*

Вариант дизайна UI для такой компании:

Веб-приложение, с логином по учетным записям

Содержит разделы для:

-учета кораблей: заведение новых кораблей в базу, списание кораблей в утиль, техобслуживание;

-учета персонала: управление карточками сотрудников, найм и увольнение, зарплаты;

-учета закупок: закупки кораблей и комплектующих;

-учета добычи рыбы: выходы в море, учет добытой рыбы;

В результате разработки была получена БД (схематическое представление смотри ниже) состоящая из 24 таблиц. Также, ниже приводится предназначение каждой таблицы.

1. Таблицы Service\_dock, Parking\_dock, Component\_info, Craft\_info. Предназначены для хранения базовой информации о составляющих компании (лодки, детали, доки). Выделение данных таблиц в отдельные сущности помогает сократить объем памяти для хранения БД, обеспечивает легкость обновления информации о составляющих компании.

2. Таблицы Component\_cost\_map и Craft\_cost\_map. Предназначены для хранения информации о стоимости (затрат) деталей и кораблях.

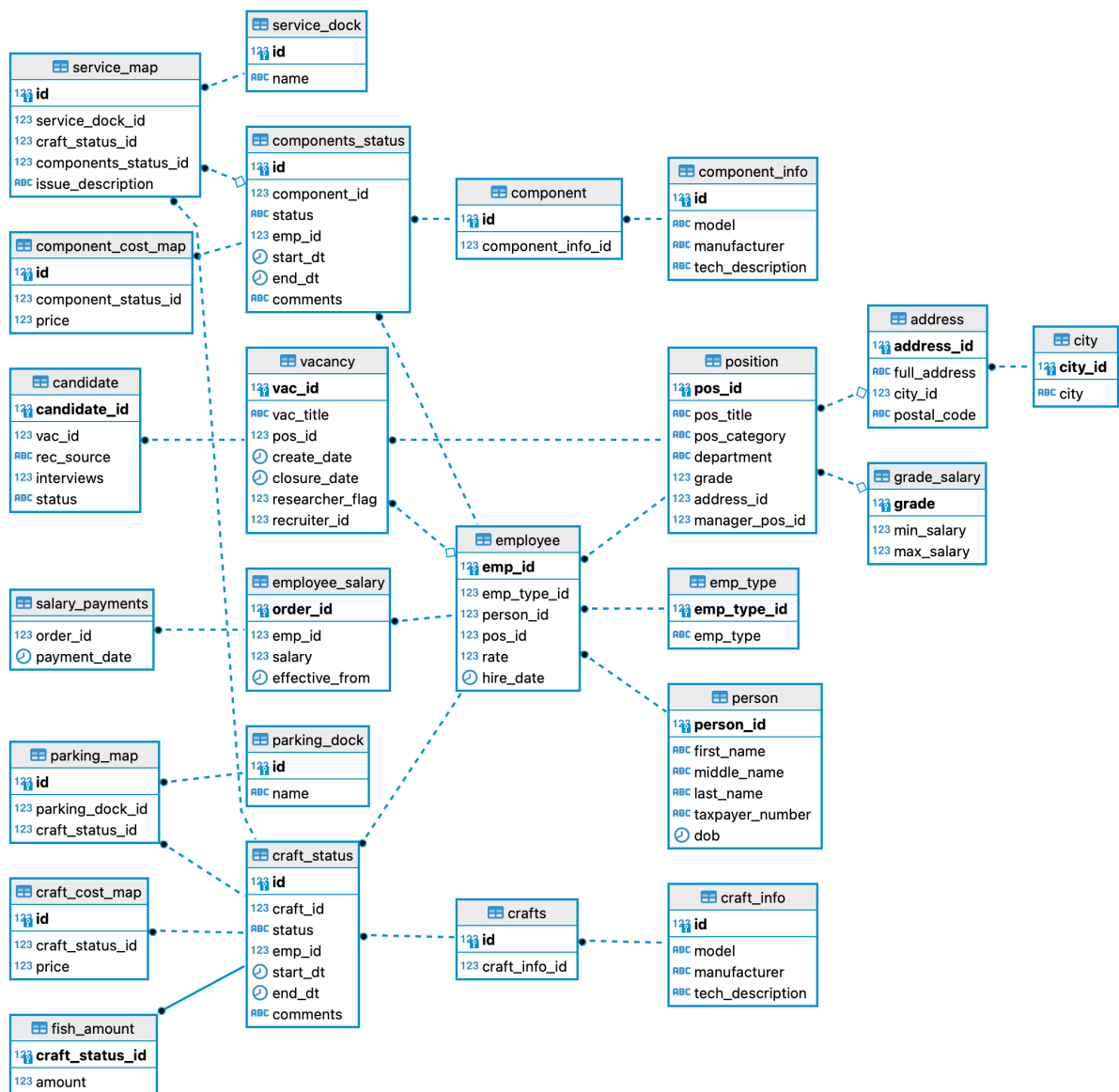
3. Таблица Service\_map. Предназначена для отслеживания проведенных сервисных мероприятий, кораблей на ремонте, деталей для ремонта и места проведения ремонта.

4. Таблица Components\_status. Предназначена для отслеживания текущего статуса деталей, истории полного цикла детали.

5. Таблицы Component и Crafts. Предназначены для присвоения уникального идентификатора всем деталям/кораблям. Данная структура помогает отследить путь

каждой конкретной детали/корабля и сократить объем памяти для БД, так как информация о каждой детали/корабле не дублируется.

6. Таблица City. Хранит информацию о городе и его идентификаторе.
7. Таблица Address. Хранит информацию о каждом адресе компании.
8. Таблица Position. Предназначена для хранения информации о рабочей позиции в компании, отделе, уровне сотрудника, и прямом руководителе.
9. Таблица Grade\_salary. Информация о зарплатном диапазоне для каждой градации позиций.
10. Таблица Vacancy. Содержит информацию о открытых вакансиях и их статусе.
11. Таблица Candidate. Информация о кандидатах и связи с вакансией.
12. Таблица emp\_type. Справочник типов устройства сотрудников.
13. Таблица Employee. Информация о сотрудниках, ссылки на персональную информацию, позицию и дата устройства.
14. Таблица Employee\_salary. Информация о заработных платах сотрудников и ее изменениях.
15. Таблица Salary\_payments. История выплат заработных плат сотрудникам.
16. Таблица Person. Персональная информация о сотрудниках.
17. Таблица Parking\_map. Таблица для отслеживания парковок кораблей.
18. Таблица Craft\_status. Предназначена для отслеживания текущего статуса кораблей, истории полного цикла корабля.
19. Таблица Fish\_amount. Информация о пойманной рыбе.



**PL/SQL API должно быть структурировано по смыслу в соответствии с компонентами UI**

**Методы API должны быть документированы**

**По желанию, можно словами описать как выглядит интерфейс UI, чтобы объяснить выбор параметров тех или иных методов на PL/SQL**

**Требуется написать SQL запросы для различного рода отчетности: см в SQL скрипте Task\_1.sql**