

Цель курсовой работы:

Получение навыков и знаний, необходимых для создания проекта базы данных в произвольной предметной области. Применение этих навыков и знаний для реализации БД по выбранной исполнителями предметной области.

Команда:

не более 2х человек

Обязательные задачи:

1. Представить предметную область, для которой производится разработка БД.
2. Сформировать ER-модель базы данных, которая должна включать в себя не менее 10 реляционных таблиц, содержать хотя бы одно отношение «многие-ко-многим».
3. Реализовать ER-модель в объектно-реляционной СУБД Oracle, с использованием языков SQL и PL/SQL.
4. Обеспечить целостность данных и автоматическое формирование уникальных первичных ключей (при помощи триггеров).
5. Реализовать процедуры, функции и программные пакеты для изменения данных в БД, которые могут быть вызваны из командной строки.
6. Разработать скрипт для заполнения таблиц наборами данных, необходимыми для выполнения зачетного задания №1.
7. Защитить разработанные программные решения.

Зачетные задачи:

1. * **ДАНО:** база данных должна включать в себя хотя бы одну стержневую сущность, которая, в свою очередь, должна содержать более 10000 строк (столбцы строковых, числовых типов, типов DATE или TIMESTAMP). Для этой стержневой сущности должно быть создано 3 или более характеристических сущности, содержащих, в свою очередь минимум 10000 строк каждая, со значениями, актуальными на определенные даты.

ЗАДАНИЕ: необходимо реализовать 1 (один) SQL-запрос, каждая строка результата которого должна содержать строку из стержневой сущности и, объединенные агрегационными функциями (например, LISTAGG), данные из характеристических сущностей. При этом должно быть достигнуто минимальное количество операций полного сканирования таблиц (FULL SCAN), обеспечен поиск по строковым атрибутам (на маску "%поисковыйзапрос%", с возможностью использования неправильной раскладки клавиатуры, например, "fjs lfyys[" = "базы данных").

2. ** **ДАНО:** сущности из зачетного задания №1.

ЗАДАНИЕ: реализовать алгоритмы Левенштейна, Джаро-Винклера и любой

* - достаточно для получения оценки «удовлетворительно»

*, **, **** - достаточно для получения оценки «хорошо»

*, **, ***, **** - обязательно для получения оценки «отлично»

другой третий метод нечеткого поиска. Применить результаты задания №2 для решения задания №1.

3. *** **ДАНО:** сущности из зачетного задания №1, вся ваша БД.

ЗАДАНИЕ: реализовать, вместо SQL-запросов, получение всех данных из БД при помощи конвейерных функций и объектов на языке PL/SQL, а также – в формате XML, схема и паке расе должны быть определены вами.

4. **** **ДАНО:** ???

ЗАДАНИЕ: дополнительное задание, формируемое непосредственно при защите курсовой работы.

Требования к ER-модели и БД:

1. ER-модель должна соответствовать предметной области (например, БД интернет-магазина по продаже автомобилей должна отражать характеристики и параметры автомобилей, БД интернет-магазина по продаже шуб – должна отражать ворсистость меха и т.д.)
2. В качестве СУБД должна использоваться СУБД Oracle версии 11.2 и новее.
3. Должна быть обеспечена корректность вводимых данных как на уровне DML-операций, так и на уровне отдельных процедур, функций и программных пакетов БД.
4. Необходимо учесть возможность многопользовательского использования БД, в частности, обеспечить поддержку транзакций.
5. Триггеры автоматического формирования первичных ключей должны быть реализованы для всех таблиц БД.
6. Целостность данных должна быть гарантирована использованием всех необходимых CONSTRAINT'ов.

Артефакты:

1. Титульный лист.
2. Описание предметной области.
3. ER-модель базы данных.
4. Описание сущностей БД.
5. Описание процедур, функций и программных пакетов в формате комментариев к коду.

Порядок выполнения курсовой работы:

1. Представление предметной области и БД.
2. Представление и защита ER-модели
3. Представление программного кода изменения данных.
4. Выполнение зачетных заданий для получения оценки за КР
5. Защита курсовой работы.

* - достаточно для получения оценки «удовлетворительно»

*, **, **** - достаточно для получения оценки «хорошо»

*, **, ***, **** - обязательно для получения оценки «отлично»