**Лабораторная работа 1**

**Тема 1:** Типы данных SQL Oracle. Стандартные функции. Арифметические и логические выражения.

*(в лекциях см. п. 2, 1.4, 3 1.1)*

**Тема 2:** Агрегатные функции. Группировка строк. Сортировка строк

*(в лекциях см. п.3.1.1)*

**Пример задания:**

1. Вывести имена всех продавцов и города, в которых они работают, кроме продавцов, работающих в Лондоне.

**Решение:**

select sname,city from sal where city<>'London'

**Результат:**

|  |  |
| --- | --- |
| **SNAME** | **CITY** |
| Serres | San Jose |
| Rifkin | Barcelona |
| Axelrod | New York |

**Задание:**

**Вариант 6.**   
1. Напишите запрос к таблице Покупателей, чей вывод может включить всех покупателей, причем с оценкой выше 200, если они не находятся в San Jose   
2. Напишите запрос, который вывел бы для каждого заказа его номер, стоимость в рублях (по текущему курсу) и имя заказчика. Данные вывести для заказчиков, размещенных в Лондоне и в Риме.   
3. Напишите запрос, который выбрал бы наименьший номер заказа на каждое число с сортировкой по убыванию чисел.   
4. Напишите запрос, который выводит имена и города всех продавцов, у которых процент комиссионных меньше, чем у Motiki, используя подзапрос.   
5. Создайте представление на основе запроса из задания 1 и, используя это представление, выведите данные о покупателях из San Jose.

**Лабораторная работа 2**

**Тема 1:** Выборка данных из объединенных таблиц *(в лекциях см. п.3 1.1).*

**Тема 2:** Подзапросы *(в лекциях см. п. 3.1.1).*

**Тема 3:** *Создание таблиц (в лекциях см. п.3.3.1).*

**Пример задания:**

1. Вывести для каждого продавца номера его заказов.

**Решение:**

select s.sname, o.onum

from sal s,ord o

where s.snum=o.snum

order by s.sname,o.onum;

**Результат:**

|  |  |
| --- | --- |
| **SNAME** | **ONUM** |
| Axelrod | 3009 |
| Motica | 3002 |
| Peel | 3003 |
| Peel | 3008 |
| Peel | 3011 |
| Rifkin | 3001 |
| Rifkin | 3006 |
| Serres | 3005 |
| Serres | 3007 |
| Serres | 3010 |

**Задание:**

**Вариант 6.**

1. Создать таблицу для хранения данных о марках автомобилей. Таблица должна содержать поле для уникального номера, названия автомобиля, стоимости.   
2. Напишите команды для вставки в таблицу 8-10 записей об автомобилях. Создайте последовательность и используйте ее в командах вставки для заполнения поля уникального номера.   
3. Напишите команду удаления строк с данными об автомобилях с нечетными номерами. Напишите команду отмены транзакции, а затем повторите команду удаления, но для четных номеров. Подтвердите транзакцию.   
4. Составить и выполнить программу PL/SQL, которая, используя SELECT … INTO …, считывает из базы данных имена продавцов, чьи комиссионные меньше 0.13, и выводит результат. Добавить в программу раздел Exception с обработчиком OTHERS, в котором определяется вид ошибки и выводится сообщения об этой ошибке.   
5. Составить и выполнить программу PL/SQL, которая, используя курсор, считывает из базы данных имена покупателей, чей рейтинг больше 100, и выводит результат.

**Лабораторная работа 3**

**Тема 1:** Создание таблиц. Последовательности *(в лекциях см. п.3.3.1, 3.3.2).*

**Тема 2:** Вставка, изменение и удаление данных из таблиц *(в лекциях см. п.3.1.2).*

**Тема 3:** Транзакции *(в лекциях см. п.3.2).*

**Пример задания:**

1. Создать таблицу с именем tab1 в базе данных одним числовым полем для целых двузначных чисел и двумя символьными полями размером по 50 байтов.

**Решение:**

create table tab1 (

col1 number(2),

col2 varchar2(50),

col3 varchar2(50)

)

**Результат:**

Table created.

**Задание:**

**Вариант 6.**   
1. Создать пакет, в который поместить функцию, которая принимает в параметрах Ваши фамилию, имя и отчество, а возвращает фамилию и инициалы. Вызвать процедуру пакета из безымянного блока.   
2. Включите в пакет процедуру, которая считывает из базы данных информацию о трех самых поздних заказах, выполненных до даты, переданной в параметре. Вызовите процедуру пакета из безымянного блока.   
3. Включите в пакет процедуру, которая изменяет стоимости в таблице марок автомобилей из задания 1 лабораторной работы 2. Стоимость должна увеличиться на 10 % для автомобилей, у которых названия длиннее, чем у автомобиля, указанного в параметре.   
4. Создать триггер PL/SQL, который запрещает удаление продавцов, если количество выполненных им заказов больше трех.   
5. Создать триггер PL/SQL, который регистрирует в журнальной таблице операции вставки, изменения и удаление заказов. Регистрация выполняется только в нерабочее время и включает имя пользователя, вид операции и стоимость заказа. Вывести содержимое журнала регистрации.

**ЗАДАНИЕ НА Контрольную Работу**

**1. Рекомендации по выполнению контрольной работы**

Прежде, чем приступать к выполнению контрольной работы, необходимо выполнить все лабораторные работы данного курса.

Контрольная работа состоит из задания, которое выбирается согласно своему варианту. Если работа выполнена в соответствии с заданием, в ней нет грубых ошибок, то вы получаете положительную оценку по контрольной работе. В противном случае работа отправляется вам на доработку.

**!!! Номер варианта контрольной работы определяется по последней цифре пароля**

Работу следует выполнять в среде программирования Oracle 10g XE или более высокой версии XE.

**2. Требования к оформлению работы**

Необходимо представить следующее:

1. Текст задания ( с подставленными данными своего варианта);
2. Текст программы (текстовый файл или файл с расширением .sql) ;
3. Результат работы программы (т.е. все данные, которые выводятся на экран, – текстовый файл или файл, содержащий скрин-шоты панели вывода среды разработки Oracle XE).

**3. Критерии оценки контрольной работы**

Оценка "отлично" выставляется, если: созданы все объекты, указанные в задании; созданные объекты отвечают указанным требованиям полноты и функциональности; при обработке данных максимально используются возможности, предоставляемые языками программирования SQL и PL/SQL.

Оценка «хорошо» выставляется, если: созданы все объекты, указанные в задании; не более, чем два объекта имеют несущественные отклонения от указанных требований, не влияющие на результирующую функциональность; программные процедуры дают правильный результат, но обработка данных выполняется нерационально, не используя возможности языков SQL и PL/SQL.

Оценка «удовлетворительно» выставляется, если: созданы не все объекты, указанные в задании, но остается возможность хранить и обрабатывать данные в соответствии с правилами предметной области; не более, чем две программные процедуры не отвечают требованиям функциональности.

Оценка «неудовлетворительно» выставляется, если имеет место одно из следующих: созданные объекты не обеспечивают хранение и обработку данных в соответствии с правилами предметной области; более, чем две программные процедуры не отвечают требованиям функциональности.

**4. Общая часть задания**

Создать две таблицы, имеющие уникальные поля и первичные ключи. Таблицы должны быть связаны с помощью внешнего ключа.

Для одной из таблиц разработать триггер для обеспечения дополнительных действий при изменение данных таблицы (см. свой вариант задания).

Создать представление для связанных таблиц, которое позволяет вывести только часть полей и часть строк таблиц.

Создать пакет PL/SQL, содержащий процедуру начального заполнения таблиц данными (по 5-7 записей в таблице) и процедуру очистки таблиц (удаления записей).

Добавить в пакет процедуру (см. свой вариант задания) изменения данных в таблицах. Значения изменяемых данных должны передаваться в процедуру как параметры.

В процедурах предусмотреть обработку исключений.

Обеспечить подтверждение транзакций при их успешном выполнении и откат - в случае возникновения исключительной ситуации.

**5. Варианты задания на выполнение контрольной работы**

**6.** Таблицы должны содержать данные об Издательствах и их Спонсорах, причем каждый спонсор может финансировать несколько издательств. Издательства должны иметь номер, наименование и сумму спонсорской помощи. Спонсоры должны иметь номер, наименование и страну пребывания. Процедура должна изменять спонсорскую помощь тем издательствам, чей спонсор размещается в указанной стране; страну и сумму изменения задать в параметре процедуры. Триггер должен регистрировать изменение с указанием пользователя, даты и времени операции, если сумма помощи увеличивается. Включить в пакет еще одну процедуру, которая выводит N самых щедрых спонсоров; N должно передаваться как параметр процедуры. Выводиться должны спонсоры и их суммарная спонсорская помощь.