**Лабораторная работа №1.** Основы языка PL/SQL

**Цель**

Цель работы: Получить общее представление о PL/SQL и познакомиться с основными понятиями языка. Изучить реляционные свойства PL/SQL, включая синтаксис языка, типы данных, способы использования SQL, инструкции DML, а также особенности работы с транзакциями. Рассмотреть синтаксис создания подпрограмм (процедур и функций), которые могут храниться и выполнятся на стороне сервера БД..

**Задание**

1. Создайте таблицу MyTable(id number, val number)

2. Напишите анонимный блок, который записывает в таблицу MyTable 10 000 целых случайных записей

3. Напишите функцию, которая выводит TRUE если четных значений val в таблице MyTable больше, FALSE если больше нечетных значений и EQUAL если количество четных и нечетных равно

4. Напишите функцию, которая по введенному значению ID, сгенерирует и выведет в консоль текстовое значение команды insert для вставки указанной строки

5. Написать процедуры реализующие DML операции (INSERT, UPDATE, DELETE) для указанной таблицы

**Лабораторная работа №2. триггеры**

**Цель**

Цель работы: Получить представление о работе объектных и системных триггеров в Oracle, а также особенности работы с автономными транзакциями. Рассмотреть синтаксис создания триггеров.

**Задание**

1. Реализовать триггер логирующий каждую успешную попытку соединения с базой
2. Построить две таблицы STUDENTS и GROUPS реализующих соответственно справочник студентов и справочник групп

|  |  |  |
| --- | --- | --- |
| Поле | Тип | комментарий |
| **STUDENTS** | | |
| ID | Number | Код студента |
| NAME | VARCHAR2 | Имя студента |
| GROUP\_ID | Number | Код группы |
| **GROUPS** | | |
| ID | Number | Код группы |
| NAME | VARCHAR2 | Название группы |
| C\_VAL | Number | Количество студентов в группе |

1. Реализовать триггеры для таблиц задания 2 проверку целостности (проверка на уникальность полей ID), генерацию автоинкрементного ключа и проверку уникальности для поля GROUP.NAME
2. Реализовать триггер реализующий Foreign Key с каскадным удалением между таблицами STUDENTS и GROUPS
3. Реализовать триггер реализующий журналирование всех действий над данными таблицы STUDENTS
4. Исходя из данных предыдущей задачи, реализовать процедуру для восстановления информации на указанный временной момент и на временное смещение
5. Реализовать триггер который в случае изменения данных в таблице STUDENTS будет соответственно обновлять информацию C\_VAL таблицы GROUPS

Дополнительные требования:

В задании 7 необходимо использовать автономные транзакции

**Лабораторная работа №3. Словарь данных**

**Цель**

Цель работы: Получить представление о работе c словарем данных. Рассмотреть синтаксис создания Job’ов.

**Задание**

Реализовать таблицу, в которой будут хранится метрики производительности базы данных наименование, опасное значение, критическое значение, какие-либо дополнительные поля, необходимые для реализации. Необходимо обеспечить возможность добавлять и удалять метрики

Написать Oracle задание с заданной периодичностью проверяющую метрики производительности в случае достижения опасного значения отправляющее email извещение, в случае достижения критического значения кроме отправки email, в отдельную таблицу в виде xml-файлов записать информацию об  активных сессиях, выполняющихся запросах

Дополнительные требования:

Текущее значение метрик необходимо брать из представлений производительности словаря данных Oracle

**Лабораторная работа №4. Очереди**

**Цель**

Цель работы: Получить представление о работе c словарем данных. Рассмотреть синтаксис создания Джобов.

Задание

1. Создать таблицу MyTable(id number, val number), val –целое, положительное число, sequence, для формирования значений первичного ключа.
2. Написать анонимный блок, записывающий в таблицу 10 000 записей
3. Написать процедуру, которая помещает все значения таблицы MyTable в очередь
4. Создать таблицу ResultTable (id number, start\_dt timestamp, finish\_dt timestamp, id\_myTable number, id\_subscriber)
5. Создать три прослушивателя очереди, которые реализуют следующий функционал:
   1. Забирают сообщение из очереди
   2. Ожидают столько val милисекунд, (т.е. чему равно значение поля val)
   3. Записывают в таблицу ResultTable:
      1. Id – первичный ключ, берется из sequence
      2. Id\_my\_Table – значение id обрабатываемого поля (из таблицы MyTable)
      3. id\_subscriber – номер прослушивателя, обрабатывающего данное сообщение
      4. start\_dt, finish\_dt – дата начала и окончания обработки сообщения соответственно