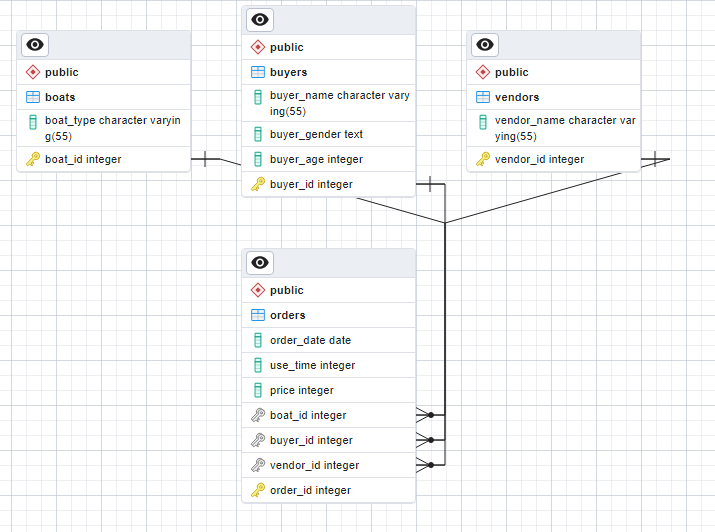
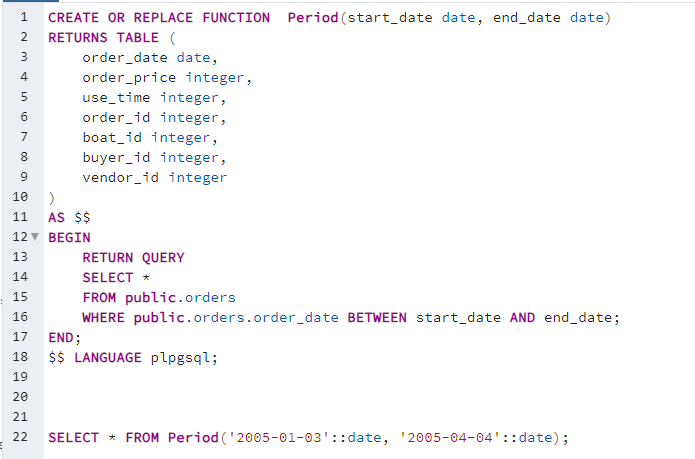
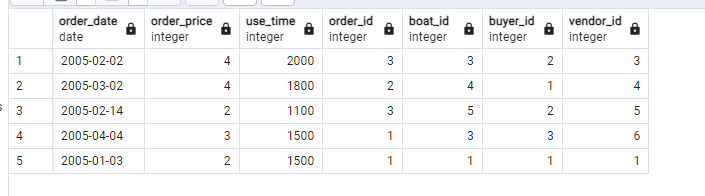
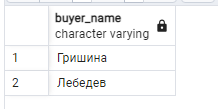
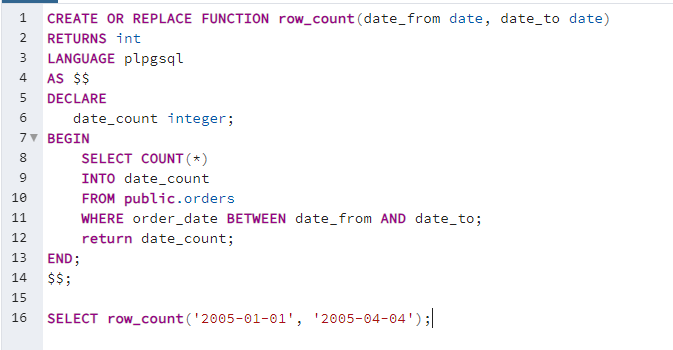
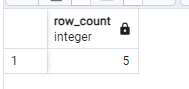
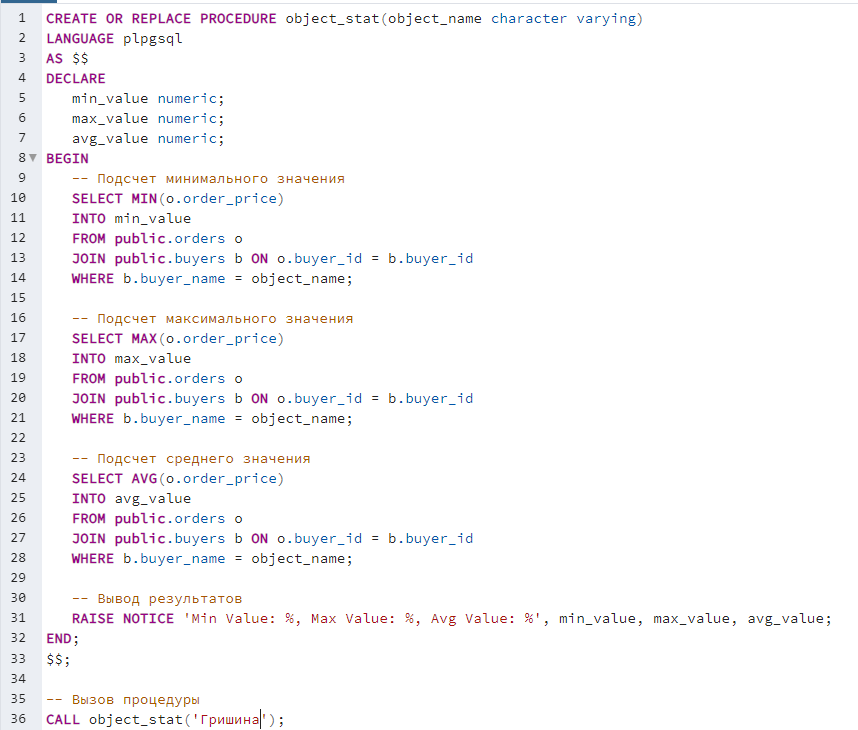
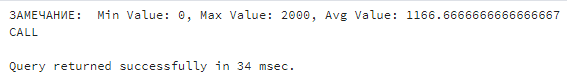
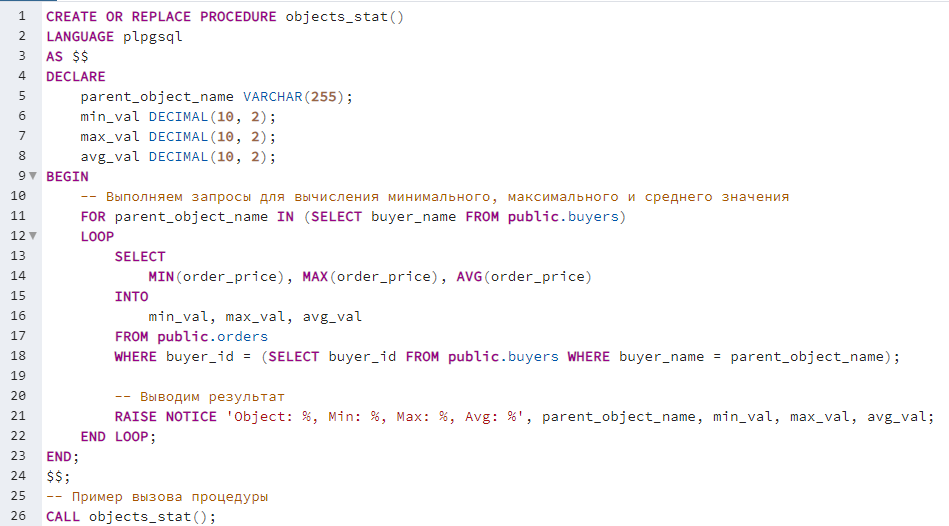
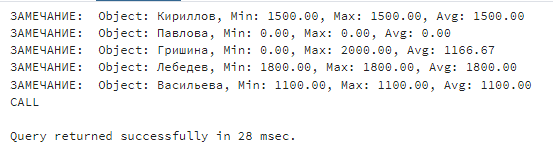
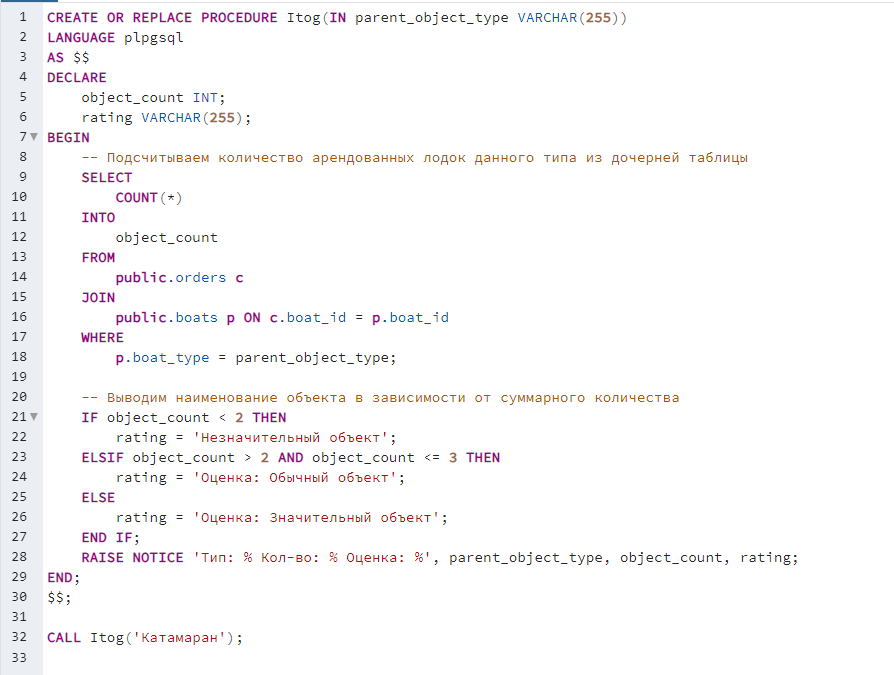
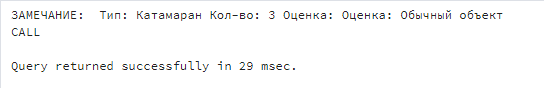
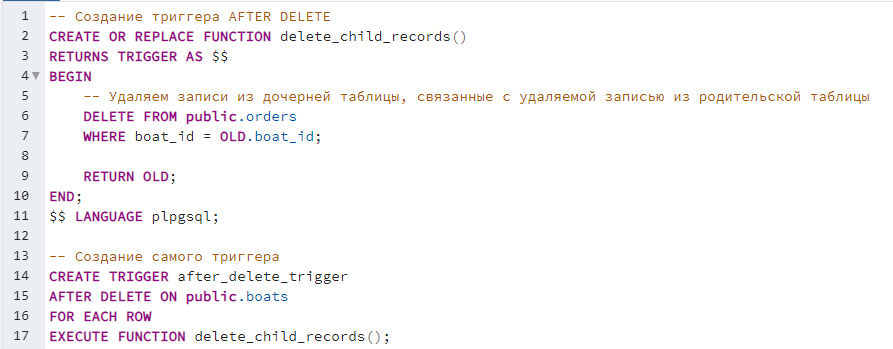
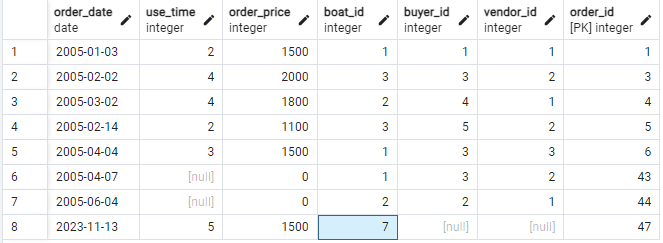
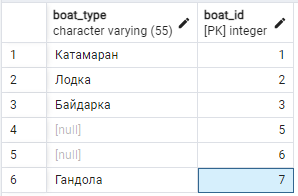
Отчет по лабораторной работе 6

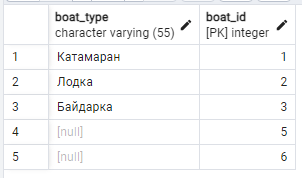
ERD:  


1. Создать функцию Period(…, …) c двумя входными параметрами типа date, которая выберет строки из дочерней таблицы в диапазоне дат, указанных первым и вторым аргументами при вызове функции Period(…, …)  
   Запрос: Вывести данные об аренде с 2005-01-03по 2005-04-04.  
   SQL: Данные:  
   
2. Создать функцию Sum\_object(…)(с одним параметром), которая возвращает список имен объектов из родительской таблицы на основании данных дочерней таблицы. Список объектов определяется значением параметра, исходя из условия, что суммарное количество объектов должно быть больше, чем заданное значение в параметре.  
   Запрос: вывести список имен объектов из таблицы buyer, для которых суммарное количество покупок в таблице orders больше 1500.  
   SQL: Данные:  
   
3. Создать функцию row\_count(…), которая подсчитывает количество строк дочерней таблицы, даты которых находятся между параметрами date\_from и date\_to.  
   Запрос: отображает количество заказов(строк), которые были сделаны между 2005-01-01 и 2005-04-04.  
   SQL: Данные:  
   
4. Создать хранимую процедуру object\_stat(…), которая подсчитывает минимальное, максимальное и среднее значение объектов в дочерней таблице, входным параметром является имя объекта.   
   Запрос: Вывести минимальную, максимальную, среднюю цену потраченную покупателем «Гришина».  
   SQL:  
     
   Данные:  
   
5. Создать хранимую процедуру objects\_stat(…), которая подсчитывает минимальное, максимальное и среднее значение каждого объекта в дочерней таблице и выводит имя объекта, входным параметром является имя объекта из родительской таблицы.  
   Запрос: подсчитать минимальное, максимальное и среднее значение каждого объекта в дочерней таблице и вывести имя покупателя.  
   SQL:  
     
   Данные:  
   
6. Создать хранимую процедуру Itog(…) с одним входным параметром, которая выводит наименование объекта по суммарному количеству объектов:   
   а) Оценка «Незначительный объект», если число объектов меньше 2   
   б) Оценка «Обычный объект», если число объектов больше 2 и меньше или равно 3   
   с) Оценка «Значительный объект», если число объектов больше 3  
   Запрос: Выводим для типа лодки «Катамаран» оценку по количеству уникальных покупок.  
   SQL:   
     
   Данные:  
   
7. Создать триггер After\_Delete, который при удалении записи из родительской таблицы удалял бы все связанные записи из дочерней таблицы. Показать результат работы триггера.  
   Запрос: Создает тригер, который реагирует на удаление записи из таблицы заказов путем удаления лодки из таблицы Лодок.  
   SQL:  
     
   Данные:  
   Исходные:



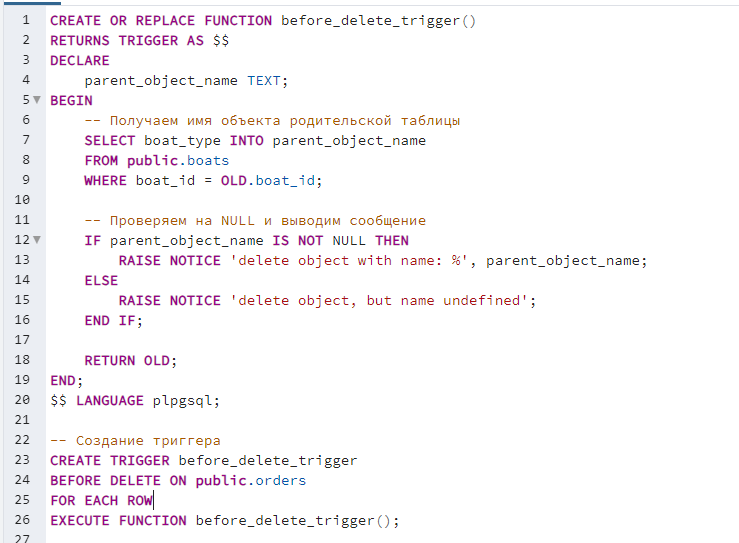
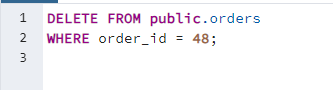


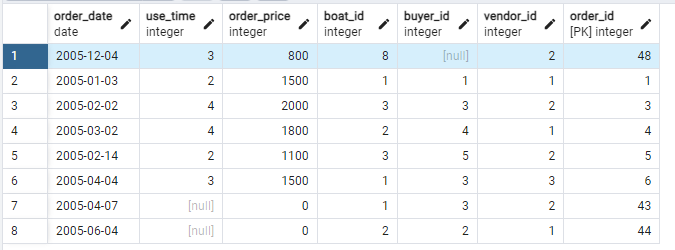
**Итоговые:**



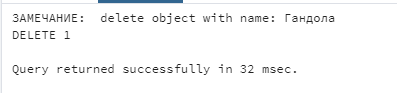


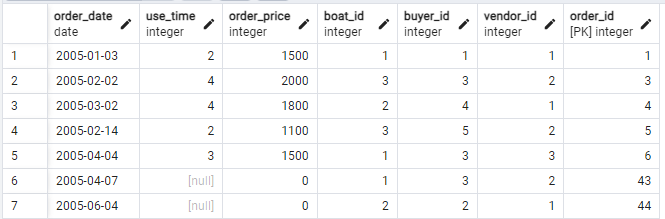
1. Создать триггер Before\_Delete, который при удалении записи из дочерней таблицы выводил бы имя объекта родительской таблицы. Показать результат работы триггера.  
   Запрос: Создает Тригер, который реагирует на удаление записи из таблицы Покупок  
   SQL:

  
  
Данные:

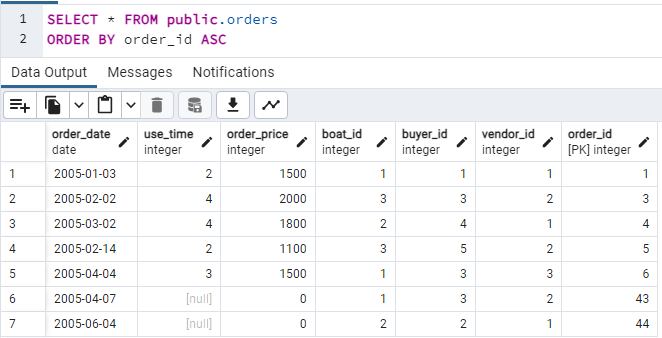
Исходные:  


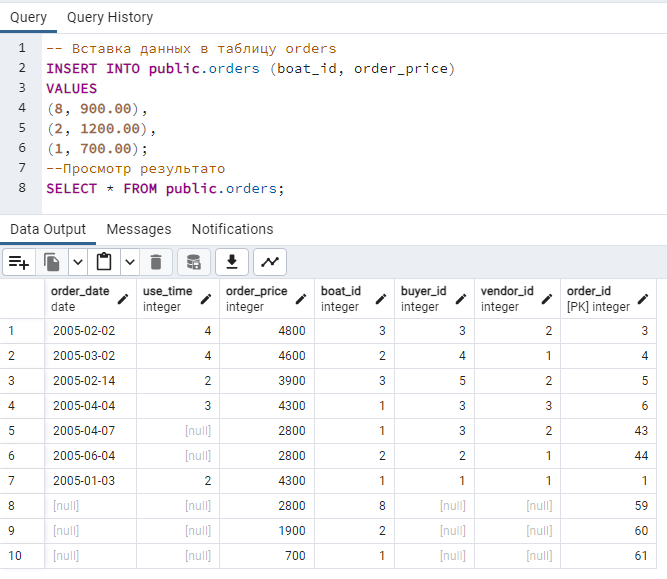
**Итоговые:**

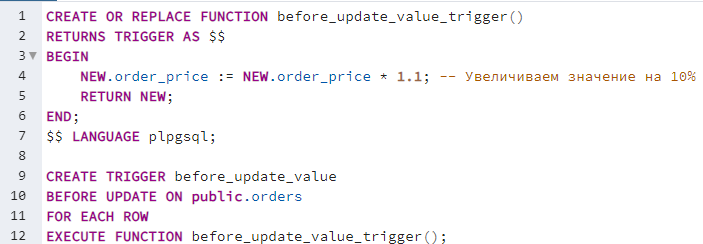
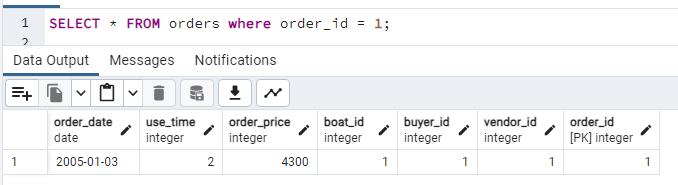




1. Создать триггер ins\_sum, который связывает триггер с таблицей для инструкций INSERT. Это действует как сумматор, чтобы суммировать значения, вставленные в один из столбцов дочерней таблицы. Триггер должен активироваться перед каждой строкой, вставленной в таблицу. Показать результат работы триггера.  
   Запрос: Этот триггер обновляет столбец цены в каждой вставленной строке, прибавляя новое значение к существующему значению перед вставкой в таблицу Покупки.  
   SQL:   
     
   Данные:



**Итоговые:**  


1. Создать триггер Before\_Update\_Value на событие UPDATE, который увеличивает значение числового поля дочерней таблицы на 10%. Показать результат работы триггера.  
   Запрос: создать триггер на обновление поля цены аренды в дочерней таблице покупки, так, чтобы цена увеличилась на 10%.  
   SQL:  
     
   Данные:  
   Исходные:  
     
   **Итоговые:**  
   