**ДЗ 1**

* Найти OrderID и количество всех продуктов в ордере (Quantity) из таблицы OrderDetails, где количество всех продуктов больше 100

**SELECT OrderID, SUM(Quantity) AS amount FROM [OrderDetails] GROUP BY OrderID HAVING amount > 100**

* Предыдущий пункт отсортировать по количеству от большего к меньшему

**сортировка по OrderID**

**SELECT OrderID, SUM(Quantity) AS amount FROM [OrderDetails] GROUP BY OrderID HAVING amount > 100 ORDER BY OrderID DESC**

**сортировка по amoun**

**SELECT OrderID, SUM(Quantity) AS amount FROM [OrderDetails] GROUP BY OrderID HAVING amount > 100 ORDER BY amount DESC**

* Найти Employees.LastName employees которые имеют более 10 ордеров. Таблица Orders содержит EmployeeID, нужно использовать INNER JOIN чтобы соединить с таблицей Employees и сгруппировать по Orders.EmployeeID.

**SELECT LastName, count(OrderID) AS amount FROM [Employees] AS E INNER JOIN Orders ON E.EmployeeID == Orders.EmployeeID GROUP BY LastName HAVING**

**amount > 10 ORDER BY amount DESC**

**ДЗ 2**

Сделать практические задания:

* [SELECT базовый](https://sqlzoo.net/wiki/SELECT_basics)
* [SUM и COUNT](https://sqlzoo.net/wiki/SUM_and_COUNT)
* [JOIN](https://sqlzoo.net/wiki/The_JOIN_operation)

Установить и настроить MySQL - <https://dev.mysql.com/downloads/installer/>

Подробная инструкция - <https://info-comp.ru/install-mysql-on-windows-10>

**ДЗ 3**

* Создать базу данных university\_db с помощью SQL запроса
* Создать сущности:
  + Student
    - student\_id - первичный ключ, автоинкремент.
    - first\_name - имя, varchar(50), Not Null
    - last\_name - фамилия, varchar(50), Not Null
    - data\_of\_birth - дата рождения
  + Module
    - module\_id - первичный ключ, автоинкремент.
    - module\_title - название модуля varchar(50), Not Null
    - level - уровень, число, Not Null
    - credits - число, по умолчанию значение 20, Not Null
* Каждый студент изучает множество “модулей”. Каждый модуль проходят множество студентов:
  + registration - таблица содержит id студента, id модуля и результат
    - Student\_id - внешний ключ таблицы student
    - module\_id - внешний ключ таблицы module
    - result - число с одним символом после запятой
* Добавить данные для таблиц:
  + student - [Daniel Radcliffe, 1995-04-23], [Emma Watson, 1994-07-15], [Rupert Grint, 1995-10-24]
  + module - [Math, 1, 30], [Physics, 1], [Chemistry, 1, 15]
  + registration
    - Daniel Radcliffe - Math, Daniel Radcliffe - Physics, Daniel Radcliffe - Chemistry
    - Emma Watson - Chemistry, Emma Watson - Math
    - Rupert Grint - Physics, Rupert Grint - Math