**SQL. Level 1**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1**. Написать запрос, котрый из таблиц **A** и **B** (столбец **id – integer identity primary key**) вернет все строки таблицы **A**, отсутствующие в таблице **B** и все строки таблицы **B**, отсутствующие в таблице **A**. Сформировать столбец **remark**, как показано в примере.   |  |  |  | | --- | --- | --- | | Имеется |  | Должно быть |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **A** | |  | **B** | |  | **Result** | | | | **id** | **name** |  | **id** | **name** |  | **id** | **name** | **remark** | | 1 | AAA |  | 4 | DDD |  | 1 | AAA | нет в B | | 2 | BBB |  | 5 | EEE |  | 2 | BBB | нет в B | | 3 | CCC |  | 6 | FFF |  | 3 | CCC | нет в B | | 4 | DDD |  | 7 | GGG |  | 7 | GGG | нет в A | | 5 | EEE |  | 8 | HHH |  | 8 | HHH | нет в A | | 6 | FFF |  | 9 | III |  | 9 | III | нет в A | |
| **2**. Написать запрос, который из таблицы **A** вернет неуникальные значения столбца **name**.   |  |  |  | | --- | --- | --- | | Имеется |  | Должно быть |  |  |  |  | | --- | --- | --- | | **A** |  | **Result** | | **name** |  | **name** | | AAAA |  | AAAA | | BBBB |  | BBBB | | AAAA |  |  | | CCCC |  |  | | BBBB |  |  | | DDDD |  |  | |
| **3**. Написать запрос, который из таблицы **A** (столбец **id – integer identity primary key**) удалит все более поздние неуникальные по столбцу **name** строки.   |  |  |  | | --- | --- | --- | | Имеется |  | Должно быть |  |  |  |  |  |  | | --- | --- | --- | --- | --- | | **A** | |  | **A** | | | **id** | **name** |  | **id** | **name** | | 1 | AAAA |  | 1 | AAAA | | 2 | BBBB |  | 2 | BBBB | | 3 | AAAA |  | 4 | CCCC | | 4 | CCCC |  | 6 | DDDD | | 5 | BBBB |  |  |  | | 6 | DDDD |  |  |  | |
| **4**. Написать запрос, который на основе данных таблицы **A** вернет следующие значения:   * минимальное и максимальной значение, сумму по столбцу **amount**; * сумму всех отрицательных и сумму всех положительных значений **amount**; * число уникальных значений в столбце **amount**.  |  |  |  | | --- | --- | --- | | Имеется |  | Должно быть |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **A** | |  | **Result** | | | | | | | **id** | **amount** |  | **min** | **max** | **sum** | **-** | **+** | **unique** | | 1 | 100 |  | -100 | 300 | 900 | -200 | 1100 | 4 | | 2 | 200 |  |  |  | | | | | | 3 | 300 |  |  |  | | | | | | 4 | -100 |  |  |  | | | | | | 5 | 200 |  |  |  | | | | | | 6 | 300 |  |  |  | | | | | | 7 | -100 |  |  |  | | | | | |
| **5**. Написать запрос, который из двух таблиц **A** и **B** вернет все строки, в столбце **name** которых есть подстрока **'**иван**'**.   |  |  |  | | --- | --- | --- | | Имеется |  | Должно быть |  |  |  |  |  |  | | --- | --- | --- | --- | --- | | **A** |  | **B** |  | **Result** | | **name** |  | **name** |  | **name** | | ОАО "Мосэнерго" |  | Смирнов Иван |  | ЗАО "Ивановохлеб" | | ЗАО "Ивановохлеб" |  | Иванов Петр |  | Смирнов Иван | | ЗАО "Туламебель" |  | Сидорова Анна |  | Иванов Петр | |

**SQL. Level 2**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1**. Написать запрос, который из таблицы **A** возвращает все строки, для которых значение столбца **name** отсутсвует в таблице **B**.   |  |  |  | | --- | --- | --- | | Имеется |  | Должно быть |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **A** | |  | **B** | |  | **Result** | | | **id** | **name** |  | **id** | **name** |  | **id** | **name** | | 1 | AAA |  | 1 | null |  | 1 | AAA | | 2 | BBB |  | 2 | BBB |  | 5 | EEE | | 3 | CCC |  | 3 | CCC |  |  |  | | 4 | DDD |  | 4 | CCC |  |  |  | | 5 | EEE |  | 5 | null |  |  |  | |  |  |  | 6 | DDD |  |  |  | |
| **2**. Таблица **A** содержит следующие данные:   |  |  | | --- | --- | | **A** | | | **id** | **name** | | 1 | AAAA | | 2 | BBBB | | 2 | CCCC |   Какой результат вернет каждый из шести следующих запросов? Опишите, почему сервер вернет такой результат.  -- вариант A  declare @name; select @name = 'XXXX'  select @name = name from A where id = 0  print @name  go  -- вариант B  declare @name; select @name = 'XXXX'  select @name = name from A where id = 1  print @name  go  -- вариант C  declare @name; select @name = 'XXXX'  select @name = name from A where id = 2  print @name  go  -- вариант D  declare @name; select @name = 'XXXX'  set @name = (select name from A where id = 0)  print @name  go  -- вариант E  declare @name; select @name = 'XXXX'  set @name = (select name from A where id = 1)  print @name  go  -- вариант F  declare @name; select @name = 'XXXX'  set @name = (select name from A where id = 2)  print @name  go |
| **3**. Опишите отличия и особенности следующих команд **delete from A** и **truncate table A**? |
| **4**. (T-SQL) В таблице **A** (столбец **id – integer identity primary key**) было три строки. Мы выполнили следующий скрипт:  insert A(name) values('DDD')  print @@identity  insert A(name) values('EEE')  print @@identity  go  Результат, выводимый скриптом:  4  2  Объясните, чем может быть вызван этот неожиданный результат? Как добиться правильного результата (=5) во втором случае?  Состояние таблицы:   |  |  |  | | --- | --- | --- | | до вставки |  | после вставки |  |  |  |  |  |  | | --- | --- | --- | --- | --- | | **A** | |  | **A** | | | **id** | **name** |  | **id** | **name** | | 1 | AAA |  | 1 | AAA | | 2 | BBB |  | 2 | BBB | | 3 | CCC |  | 3 | CCC | |  |  |  | 4 | DDD | |  |  |  | 5 | EEE | |
| **5**. (T-SQL) Найдите ошибку в следующем скрипте:  begin transaction A  insert A(id,name) values(1,'AAAA')  begin transaction  save transaction B  insert B(id,name) values(1,'AAAA')  if @@error <> 0 rollback transaction B else commit transaction B  commit transaction A |