**1.**

Дано две таблицы tab1 и tab2:

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Дата | Город | Сумма |
| 1 | 01.01.2019 | Москва | 100 |
| 2 | 01.01.2019 | Санк-Петербург | 200 |
| 3 | 01.01.2019 | Самара | 300 |

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Дата | Город | Сумма |
| 1 | 01.01.2019 | Москва | 100 |
| 2 | 01.01.2019 | Санк-Петербург | 200 |

Используя их, необходимо получить выборку (4 разными запросами):

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Дата | Город | Сумма |
| 3 | 01.01.2019 | Самара | 300 |

**1 Вариант**

SELECT \* FROM tab1

EXCEPT

SELECT \* FROM tab2

**2 вариант**

SELECT \* FROM tab1

LEFT JOIN tab2 on tab2.ID = tab1.ID

WHERE tab2.ID is NULL

**3 вариант**

SELECT \* FROM tab1 t

WHERE t.ID not in (SELECT t2.ID FROM tab2 t2)

**4 вариант**

SELECT \* FROM tab1 t

WHERE NOT EXISTS (SELECT \* FROM tab2 WHERE t.ID = tab2.ID)

**2.**

Дана таблица tab1:

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Дата | Город | Сумма |
| 1 | 01.01.2019 | Москва | 100 |
| 2 | 02.01.2019 | Москва | 200 |
| 3 | 03.01.2019 | Москва | 300 |
| 4 | 01.01.2019 | Санк-Петербург | 200 |
| 5 | 02.01.2019 | Санк-Петербург | 300 |
| 6 | 03.01.2019 | Санк-Петербург | 400 |
| 7 | 01.01.2019 | Самара | 300 |
| 8 | 02.01.2019 | Самара | 400 |
| 9 | 03.01.2019 | Самара | 500 |

Используя ее, необходимо получить выборку (2 разными запросами):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Дата | Город | Сумма | ИТОГО по городу |
| 1 | 01.01.2019 | Москва | 100 | 600 |
| 2 | 02.01.2019 | Москва | 200 | 600 |
| 3 | 03.01.2019 | Москва | 300 | 600 |
| 4 | 01.01.2019 | Санк-Петербург | 200 | 900 |
| 5 | 02.01.2019 | Санк-Петербург | 300 | 900 |
| 6 | 03.01.2019 | Санк-Петербург | 400 | 900 |
| 7 | 01.01.2019 | Самара | 300 | 1200 |
| 8 | 02.01.2019 | Самара | 400 | 1200 |
| 9 | 03.01.2019 | Самара | 500 | 1200 |

**Вариант 1**

SELECT \*, SUM(Сумма) OVER(PARTITION BY Город) as 'ИТОГО по городу' FROM tab1

**Вариант 2**

SELECT \* FROM tab1 t

JOIN (SELECT Город, SUM(Сумма) as 'ИТОГО по городу' FROM tab1

GROUP BY Город) t1 on t.Город = t1.Город

**3.**

Дана таблица tab1:

|  |  |  |
| --- | --- | --- |
| Дата | Регион | Сумма |
| 01.01.2019 | Москва | 100 |
| 01.01.2019 | Москва | 100 |
| 01.01.2019 | Санк-Петербург | 200 |
| 01.01.2019 | Санк-Петербург | 200 |
| 01.01.2019 | Самара | 300 |
| 01.01.2019 | Самара | 300 |

Необходимо удалить из нее дублирующиеся записи по полям Дата, Регион, Сумма:

|  |  |  |
| --- | --- | --- |
| Дата | Регион | Сумма |
| 01.01.2019 | Москва | 100 |
| 01.01.2019 | Санк-Петербург | 200 |
| 01.01.2019 | Самара | 300 |

WITH CTE (Дата, Регион, Сумма, Номер) AS

(SELECT Дата, Регион, Сумма, ROW\_NUMBER ()

OVER (PARTITION BY Дата, Регион, Сумма ORDER BY Регион) FROM tab1)

DELETE FROM CTE

WHERE Номер > 1

**4.**

Используя иерархический запрос к таблице dual, необходимо получить первые числа каждого месяца текущего года:

|  |
| --- |
| 01.01.2019 |
| 01.02.2019 |
| 01.03.2019 |
| 01.04.2019 |
| 01.05.2019 |
| 01.06.2019 |
| 01.07.2019 |
| 01.08.2019 |
| 01.09.2019 |
| 01.10.2019 |
| 01.11.2019 |
| 01.12.2019 |

SELECT DATEADD(DAY, 1 - day(Дата), Дата) as 'первое число каждого месяца' FROM dual