Table of Contents

MySQL implements the Union query but not the Intersect operation not the Minus operation directly. You can use subqueries to implement these.

1. Implementing an Intersect

The Intersect operator returns rows which are in both of the select result sets.

Suppose we want to return HW items purchased both in Nov and in Dec of any year

This would be the INTERSECT Query

```
      Select prod_id, catg_id, prod_name
      From prd_products
      Join oe_order_details  using (prod_id)
      Join oe_order_headers  using (ord_id)
      Where catg_id = 'HW'
      And extract(month from ord_date) = 11
   INTERSECT
      Select prod_id, catg_id, prod_name
      From prd_products
      Join oe_order_details  using (prod_id)
      Join oe_order_headers  using (ord_id)
      Where catg_id = 'HW'
      And extract(month from ord_date) = 12
   ;
```

Demo 01: Implementing an intersection

```
    Select prod_id, catg_id, prod_name
    From a_prd.products
    Where prod_id in (
       Select prod_id
       From  a_oe.order_details
       Join  a_oe.order_headers  using (ord_id)
       Where catg_id = 'HW'
       And extract(month from ord_date) = 11 )
    And  prod_id in (
       Select prod_id
       From  a_oe.order_details
       Join  a_oe.order_headers  using (ord_id)
       Where catg_id = 'HW'
       And extract(month from ord_date) = 12 )
     ;
+---------+---------+-----------------+
| prod_id | catg_id | prod_name       |
+---------+---------+-----------------+
|    1080 | HW      | Cornpopper      |
|    1100 | HW      | Blender         |
|    1110 | HW      | Pancake griddle |
+---------+---------+-----------------+
```

2. Implementing a Minus

The Minus operator returns rows which are in one of the select result sets but not the other. The Minus operator is not commutative.

This shows HW items that were purchased in Nov but not in Dec. But MySQL does not support the MINUS syntax

```
    Select prod_id, catg_id, prod_name
    From a_prd.products
    Join a_oe.order_details  using (prod_id)
    Join a_oe.order_headers  using (ord_id)
    Where catg_id = 'HW'
    And extract(month from ord_date) = 11
MINUS
    Select prod_id, catg_id, prod_name
    From a_prd.products
    Join a_oe.order_details  using (prod_id)
    Join a_oe.order_headers  using (ord_id)
    Where catg_id = 'HW'
    And extract(month from ord_date) = 12
    ;
```

Demo 02: Implementing a Minus: This shows HW items that were purchased in Nov but not in Dec

```
    Select prod_id, catg_id, prod_name
    From a_prd.products
    Where prod_id in (
             Select prod_id
             From  a_oe.order_details
             Join  a_oe.order_headers  using (ord_id)
             Where catg_id = 'HW'
             And extract(month from ord_date) = 11 )
    And  prod_id NOT in (
             Select prod_id
             From  a_oe.order_details
             Join  a_oe.order_headers  using (ord_id)
             Where catg_id = 'HW'
             And extract(month from ord_date) = 12 )
    ;
+---------+---------+------------+
| prod_id | catg_id | prod_name  |
+---------+---------+------------+
|    1000 | HW      | Hand Mixer |
|    1070 | HW      | Iron       |
|    1071 | HW      | Iron       |
+---------+---------+------------+
```

Demo 03: This shows HW items that were purchased in Dec but not in Nov

```
    Select prod_id, catg_id, prod_name
    From a_prd.products
    Where prod_id NOT in (
             Select prod_id
             From  a_oe.order_details
             Join  a_oe.order_headers  using (ord_id)
             Where catg_id = 'HW'
```

```
            And extract(month from ord_date) = 11 )
   And  prod_id in (
            Select prod_id
            From  a_oe.order_details
            Join  a_oe.order_headers  using (ord_id)
            Where catg_id = 'HW'
            And extract(month from ord_date) = 12 )
;
+---------+---------+-----------+
| prod_id | catg_id | prod_name |
+---------+---------+-----------+
|    1090 | HW      | Gas grill |
+---------+---------+-----------+
```