

# Cursors

## Implicit Cursor Example

```
mysql> show tables;
```

```
+-----+  
| Tables_in_miniproject_23203 |  
+-----+  
| cust_order_view      |  
| customer              |  
| jewellery             |  
| monthly_sales        |  
| order_items          |  
| orders                |  
| payment               |  
| unit                  |  
+-----+
```

8 rows in set (0.00 sec)

```
mysql> select * from payment;
```

```
+-----+-----+-----+-----+-----+  
| paymentid | orderid | mode      | payment_date | amount |  
+-----+-----+-----+-----+-----+  
| PAY0000001 | OR0001 | Credit Card | 2025-03-20 | 75580 |  
| PAY0000002 | OR0002 | Debit Card | 2025-04-10 | 25000 |  
| PAY0000003 | OR0003 | Cash       | 2025-05-18 | 84700 |  
| PAY0000004 | OR0004 | Debit Card | 2025-07-08 | 74490 |  
| PAY0000005 | OR0005 | Credit Card | 2025-09-25 | 110000 |  
| PAY0000006 | OR0006 | Debit Card | 2025-11-12 | 108400 |  
| PAY0000007 | OR0007 | Debit Card | 2026-03-22 | 103130 |  
| PAY0000008 | OR0008 | Cash       | 2026-06-05 | 105380 |  
| PAY0000010 | OR0010 | Credit Card | 2026-12-31 | 57800 |  
| PAY0000011 | OR0011 | Debit Card | 2025-04-09 | 30000 |  
+-----+-----+-----+-----+-----+
```

10 rows in set (0.00 sec)

```
mysql> -- if payment is above 100000 then update payment mode to credit card
```

```
mysql> delimiter $$
```

```
mysql>
```

```
create procedure UpdateHighPaymentsToCreditCard()
```

```
→ begin
```

```
→   update payment
```

```
→   set mode = 'Credit Card'
```

```
→   where amount >= 100000;
```

```
→
```

```
→   select
```

```
ROW_COUNT() as 'Records Updated to Credit Card';
```

```
→ end$$
```

```
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> delimiter ;
```

```
mysql>
```

```
call UpdateHighPaymentsToCreditCard();
```

```
+-----+
```

```
| Records Updated to Credit Card |
```

```
+-----+
```

```
|                3 |
```

```
+-----+
```

```
1 row in set (0.04 sec)
```

```
Query OK, 0 rows affected (0.05 sec)
```

```
mysql> select * from payment
```

```
→ ;
```

```
+-----+-----+-----+-----+-----+
```

```
| paymentid | orderid | mode      | payment_date | amount |
```

```
+-----+-----+-----+-----+-----+
```

```
| PAY0000001 | OR0001 | Credit Card | 2025-03-20 | 75580 |
```

```
| PAY0000002 | OR0002 | Debit Card | 2025-04-10 | 25000 |
```

```
| PAY0000003 | OR0003 | Cash       | 2025-05-18 | 84700 |
```

```
| PAY0000004 | OR0004 | Debit Card | 2025-07-08 | 74490 |
```

```
| PAY0000005 | OR0005 | Credit Card | 2025-09-25 | 110000 |
```

```
| PAY0000006 | OR0006 | Credit Card | 2025-11-12 | 108400 |
```

```
| PAY0000007 | OR0007 | Credit Card | 2026-03-22 | 103130 |
```

PAY00000008	OR0008	Credit Card	2026-06-05	105380
PAY00000010	OR0010	Credit Card	2026-12-31	57800
PAY00000011	OR0011	Debit Card	2025-04-09	30000

+-----+-----+-----+-----+-----+

10 rows in set (0.00 sec)

mysql>

## Explicit Cursor

mysql> -- show all orders with total price > 50,000

mysql> delimiter \$\$

mysql> create

**procedure ListHighValueOrders()**

→ begin

→ declare done int default 0;

→ declare v\_orderid char(6);

→ declare v\_custid char(5);

→ declare v\_total\_price decimal(10,2);

→

→

**declare cur\_high\_orders cursor for**

→ select orderid,custid, total\_price

→ from orders

→ where total\_price > 50000;

→

→ declare continue handler for not found set done = 1;

→

→ open cur\_high\_orders;

→

→

**read\_loop : LOOP**

→ fetch cur\_high\_orders into v\_orderid, v\_custid,

→ v\_total\_price;

→ if done then

```

→          leave read_loop;
→          end if;
→
→ select concat('Order ID: ' , v_orderid , ' , Customer : ' , v_custid , ' , Amount : ' ,
v_total_price) as High_Value_Order;
→ end loop;
→ close cur_high_orders;
→ end$$

```

Query OK, 0 rows affected (0.05 sec)

mysql>

mysql> CALL ListHighValueOrders();

→ ^C

mysql> delimiter ;

mysql> CALL ListHighValueOrders();

```

+-----+
| High_Value_Order          |
+-----+
| Order ID: OR0001 , Customer : C0001 , Amount : 75580.00 |
+-----+

```

1 row in set (0.04 sec)

```

+-----+
| High_Value_Order          |
+-----+
| Order ID: OR0003 , Customer : C0003 , Amount : 84700.00 |
+-----+

```

1 row in set (0.05 sec)

```

+-----+
| High_Value_Order          |
+-----+
| Order ID: OR0004 , Customer : C0001 , Amount : 74490.00 |
+-----+

```

1 row in set (0.06 sec)

```
+-----+
| High_Value_Order          |
+-----+
| Order ID: OR0005 , Customer : C0004 , Amount : 110000.00 |
+-----+
1 row in set (0.06 sec)
```

```
+-----+
| High_Value_Order          |
+-----+
| Order ID: OR0006 , Customer : C0005 , Amount : 108400.00 |
+-----+
1 row in set (0.06 sec)
```

```
+-----+
| High_Value_Order          |
+-----+
| Order ID: OR0007 , Customer : C0002 , Amount : 103130.00 |
+-----+
1 row in set (0.07 sec)
```

```
+-----+
| High_Value_Order          |
+-----+
| Order ID: OR0008 , Customer : C0003 , Amount : 105380.00 |
+-----+
1 row in set (0.07 sec)
```

```
+-----+
| High_Value_Order          |
+-----+
| Order ID: OR0010 , Customer : C0004 , Amount : 57800.00 |
+-----+
1 row in set (0.08 sec)
```

Query OK, 0 rows affected (0.09 sec)

---

---

