1.提高性能  
DELIMITER $$

CREATE PROCEDURE GetTopProducts(

IN count INT

)

BEGIN

SELECT product\_name, SUM(quantity) AS total\_quantity

FROM order\_details od

JOIN products p ON od.product\_id = p.product\_id

GROUP BY product\_name

ORDER BY total\_quantity DESC

LIMIT count;

END$$

DELIMITER ;

2. 增強安全性:  
DELIMITER $$

CREATE PROCEDURE CreateUser(

IN username VARCHAR(50),

IN password VARCHAR(100),

IN email VARCHAR(100)

)

BEGIN

INSERT INTO users (username, password\_hash, email)

VALUES (username, SHA2(password, 512), email);

END$$

DELIMITER ;

3.封裝業務邏輯

DELIMITER $$

CREATE PROCEDURE PlaceOrder(

IN customerID INT,

IN productID INT,

IN quantity INT

)

BEGIN

DECLARE productPrice DECIMAL(10,2);

DECLARE availableStock INT;

SELECT price INTO productPrice FROM products WHERE product\_id = productID;

SELECT quantity INTO availableStock FROM inventory WHERE product\_id = productID;

IF quantity > availableStock THEN

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Insufficient stock.';

END IF;

INSERT INTO orders (customer\_id, total\_amount)

VALUES (customerID, quantity \* productPrice);

INSERT INTO order\_details (order\_id, product\_id, quantity, price)

VALUES (LAST\_INSERT\_ID(), productID, quantity, productPrice);

UPDATE inventory SET quantity = quantity - availableStock

WHERE product\_id = productID;

END$$

DELIMITER ;

4.事務  
DELIMITER $$

CREATE PROCEDURE TransferFunds(

IN fromAccount INT,

IN toAccount INT,

IN amount DECIMAL(10,2)

)

BEGIN

DECLARE availableBalance DECIMAL(10,2);

START TRANSACTION;

SELECT balance INTO availableBalance

FROM accounts

WHERE account\_id = fromAccount;

IF availableBalance < amount THEN

ROLLBACK;

SIGNAL SQLSTATE '45000' SET MESSAGE\_TEXT = 'Insufficient funds.';

END IF;

UPDATE accounts

SET balance = balance - amount

WHERE account\_id = fromAccount;

UPDATE accounts

SET balance = balance + amount

WHERE account\_id = toAccount;

COMMIT;

END$$

DELIMITER ;