```sql
SET SERVEROUTPUT ON;

-- find_customer (customer_id IN NUMBER, found OUT NUMBER)
CREATE OR REPLACE PROCEDURE find_customer(customerId IN NUMBER, found OUT NUMBER) AS
BEGIN
    SELECT COUNT(*)
    INTO found
    FROM customers
    WHERE customer_id = customerId;

EXCEPTION
    WHEN NO_DATA_FOUND THEN
        found := 0;
END;
/

-- find_product
CREATE OR REPLACE PROCEDURE find_product(productId IN NUMBER, price OUT products.list_price%TYPE) AS
BEGIN
    SELECT  list_price
    INTO    price
    FROM    products
    WHERE   product_id = productId;

    EXCEPTION
    WHEN NO_DATA_FOUND THEN
        price := 0;
END;
/

-- add_order_item
CREATE OR REPLACE PROCEDURE add_order_item(orderId IN order_items.order_id%type,

                                           itemId IN order_items.item_id%type,

                                           productId IN order_items.product_id%type,

                                           inQuantity IN order_items.quantity%type,

                                           price IN order_items.unit_price%type) AS
BEGIN

    INSERT INTO order_items
    (order_id, item_id, product_id, quantity, unit_price)

    VALUES

    (orderId, itemId, productId, inQuantity, price);

END;
/

-- add_order
CREATE OR REPLACE PROCEDURE add_order(customerId IN NUMBER, new_order_id OUT NUMBER) AS

BEGIN
    SELECT  MAX(order_id)
    INTO    new_order_id
    FROM    orders;
    new_order_id := new_order_id + 1;

    INSERT INTO orders

    (order_id, customer_id, status, salesman_id, order_date)

    VALUES
    (new_order_id, customerId, 'Shipped', 56, SYSDATE);
END;
/

-- check the new order added through c++:
SELECT * FROM ORDERS ORDER BY ORDER_DATE DESC;
```