1. **Create a PL/SQL block which displays the list of customers (all the informations):**

```
DECLAIRE

    CURSOR cur IS SELECT * FROM Customer

BEGIN

    FOR rec IN cur LOOP

      dbms_output.put_line("id " || rec.customer_id "first
      name: " || rec.customer_Name || " Tel: " ||
      rec.customer_Tel);

    END  LOOP;

END;
```

2. **Create a Procedure PS_Customer_Prodcuts which displays the list of product names of a given customer (customer_id). If no result returned (No_Data_Found exception raised), display the following message "No products returned or customer not found"**

```
CREATE OR REPLACE PROCEDURE PS_Customer_Products
(v_cust_id Customer.customer_id %type) IS

    CURSOR cur IS SELECT Product_id FROM Orders WHERE
    Customer_id=v_cust_id;

    name_prod Varchar2(20);

    BEGIN

        FOR rec IN cur LOOP

            SELECT product_name into name_prod FROM
            Products WHERE Product_id=rec;

            dbms_output.put_line(name_prod);

        END LOOP;

    EXCEPTION

        WHEN no_data_found THEN
```

```
        dbms_output.put_line("No products returned or
        customer not found");

    END;
```

3. **Create a Function FN_Customer_Orders which returns the number of orders of a given customer (customer_id):**

```
CREATE OR REPLACE FUNCTION FN_Customer_Orders (v_cust_id
Customer.customer_id %type) RETURN number IS

    nb_orders number;

    BEGIN

        SELECT count(*) into nb_orders FROM ORDERS WHERE
        Customer_id=v_cust_id;

        RETURN nb_orders;

    END;
```

4. **Create a trigger TRIG_INS_ORDERS which starts before each INSERT on Orders tables and test if the OrderDate >= SYSDATE. If not the following message is displayed "Order Date must be greater than or equal to today's date":**

```
CREATE OR REPLACE TRIGGER TRIG_INS_ORDERS

BEFORE INSERT ON Orders

BEGIN

    IF (OrderDate<SYSDATE) THEN

        dbms_output.put_line("Order Date must be greater
        than or equal to today's date");

    END IF;

END;
```