

PL/SQL MANDATORY HANDS-ON

Exercise 1: Control Structures

1. Create Customers table

```
CREATE TABLE Customers (  
    CustomerID    NUMBER PRIMARY KEY,  
    Name          VARCHAR2(100),  
    Age           NUMBER,  
    Balance       NUMBER(10, 2),  
    LoanInterestRate NUMBER(5, 2),  
    IsVIP         VARCHAR2(5) DEFAULT 'FALSE'  
);
```

2. CREATE TABLE Loans (

```
    LoanID    NUMBER PRIMARY KEY,  
    CustomerID NUMBER,  
    DueDate   DATE,  
    FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)  
);
```

3. INSERT INTO Customers VALUES (1, 'Ravindar Singh', 65, 15000.00, 10.5, 'FALSE');

INSERT INTO Customers VALUES (2, 'Manjula Vodnala', 45, 9000.00, 12.0, 'FALSE');

INSERT INTO Customers VALUES (3, 'Shravya Verma', 70, 25000.00, 9.0, 'FALSE');

INSERT INTO Customers VALUES (4, 'Kiran Das', 58, 11000.00, 11.0, 'FALSE');

INSERT INTO Customers VALUES (5, 'Meena Iyer', 62, 8000.00, 10.0, 'FALSE');

COMMIT;

4. INSERT INTO Loans VALUES (101, 1, SYSDATE + 10);

INSERT INTO Loans VALUES (102, 2, SYSDATE + 40);

INSERT INTO Loans VALUES (103, 3, SYSDATE + 5);

INSERT INTO Loans VALUES (104, 4, SYSDATE - 1);

INSERT INTO Loans VALUES (105, 5, SYSDATE + 25);

COMMIT;

The bank wants to apply a discount to loan interest rates for customers above 60 years old.

The screenshot shows the Oracle Live SQL web interface. At the top, there's a browser tab for 'Oracle Live SQL' and a URL bar showing 'livesql.oracle.com/next/'. The interface includes a 'Live SQL' header with 'Worksheet' and 'Library' tabs. Below this, a SQL query is entered in a text area: `SELECT * FROM Customers;`. The query is executed, and the results are displayed in a table below. The table has columns: CUSTOMERID, NAME, AGE, BALANCE, LOANINTERESTRATE, and ISVIP. The results show 5 rows of customer data. The 'LOANINTERESTRATE' column is highlighted in yellow in the original image.

	CUSTOMERID	NAME	AGE	BALANCE	LOANINTERESTRATE	ISVIP
1	1	Ravinder Singh	65	15000	9.5	TRUE
2	2	Manjula Vodnala	45	9000	12	FALSE
3	3	Shravya Verma	70	25000	8	TRUE
4	4	Kiran Das	58	11000	11	TRUE
5	5	Meena Iyer	62	8000	9	FALSE

SCENARIO-2:

A customer can be promoted to VIP status based on their balance.

The screenshot shows the Oracle Live SQL interface. The SQL Worksheet contains a PL/SQL procedure named `FOR cust_rec IN` that iterates over the `CUSTOMERS` table, updating the `ISVIP` status to `TRUE` for customers with a balance greater than 10,000. The procedure is executed, and the Script output tab shows the message: `VIP STATUS CHANGED FOR CUSTOMERS WITH BALANCE > 10000`. The DBMS output tab shows the message: `PL/SQL procedure successfully completed.`

```
SQL> BEGIN
FOR cust_rec IN(
  SELECT CUSTOMERID, BALANCE
  FROM CUSTOMERS
  WHERE BALANCE > 10000
) LOOP
  UPDATE CUSTOMERS
  SET ISVIP = 'TRUE'
  WHERE CUSTOMERID = cust_rec.CUSTOMERID;
END LOOP;
DBMS_OUTPUT.PUT_LINE('VIP STATUS CHANGED FOR CUSTOMERS WITH BALANCE > 10000');
END;
```

Query result | Script output | DBMS output | Explain Plan | SQL history

VIP STATUS CHANGED FOR CUSTOMERS WITH BALANCE > 10000

PL/SQL procedure successfully completed.

Elapsed: 00:00:00.005

The screenshot shows the Oracle Live SQL interface. The SQL Worksheet contains a simple `SELECT * FROM CUSTOMERS;` query. The Query result tab displays the results of the query, showing a table with columns: `CUSTOMERID`, `NAME`, `AGE`, `BALANCE`, `LOANINTERESTRAT`, and `ISVIP`. The results show five customers, with the `ISVIP` status highlighted in yellow for each row.

```
1 SELECT * FROM CUSTOMERS;
2
3
4
5
6
7
8
9
10
11
12
13
14
15
```

Query result | Script output | DBMS output | Explain Plan | SQL history

Download Execution time: 0.002 seconds

	CUSTOMERID	NAME	AGE	BALANCE	LOANINTERESTRAT	ISVIP
1	1	Ravindar Singh	65	15000	9.5	TRUE
2	2	Manjula Vodnala	45	9000	12	FALSE
3	3	Shravya Verma	70	25000	8	TRUE
4	4	Kiran Das	58	11000	11	TRUE
5	5	Meena Iyer	62	8000	9	FALSE

SCENARIO-3:

The bank wants to send reminders to customers whose loans are due within the next 30 days.

The screenshot shows the Oracle Live SQL interface. The top bar includes the 'Live SQL' logo, a 'Worksheet' tab, and a 'Library' icon. The main area contains a SQL worksheet with the following PL/SQL code:

```
1 BEGIN
2   FOR loan_rec IN (
3     SELECT l.LOANID, c.NAME,
4            l.DUEDATE
5     FROM LOANS l
6     JOIN CUSTOMERS c ON
7           c.CUSTOMERID=l.CUSTOMERID
8     WHERE l.DUEDATE BETWEEN SYSDATE AND SYSDATE+30
9   ) LOOP
10    DBMS_OUTPUT.PUT_LINE('!!!!REMINDER : LoanID '|| loan_rec.LOANID || ' for '|| loan_rec.NAME || ' is due on '|| TO_CHAR(loan_rec.DUEDATE, 'DD-MON-YY'));
11  END LOOP;
12 END;
```

Below the code editor, the 'Script output' tab is active, displaying the results of the PL/SQL procedure:

```
SELECT l.LOANID, c.NAME,
       l.DUEDATE...
Show more...
```

!!!!REMINDER : LoanID 101 for Ravindar Singh is due on 09-JUL-25
!!!!REMINDER : LoanID 103 for Shrayya Verma is due on 04-JUL-25
!!!!REMINDER : LoanID 105 for Meena Iyer is due on 24-JUL-25

PL/SQL procedure successfully completed.
Elapsed: 00:00:00.017

The bottom of the interface features a footer with links: 'About Oracle', 'Contact Us', 'Legal Notices', 'Terms and Conditions', 'Your Privacy Rights', 'Delete Your Live SQL Account', and 'Cookie Preferences'. The version number 'r31.1' is also displayed.