WEEK 2-MANDATORY HANDS ON EXERCISES

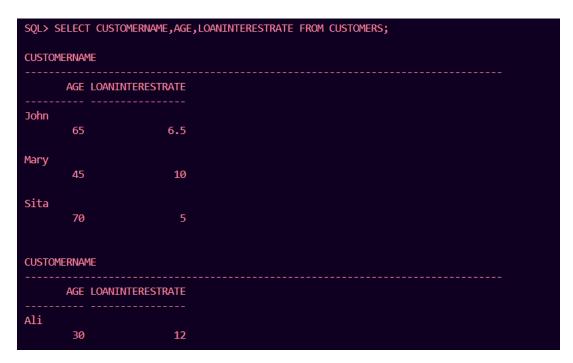
Module 3 - PL/SQL Programming

Exercise 1: Control Structures

Scenario 1: The bank wants to apply a discount to loan interest rates for customers above 60 years old.

 Question: Write a PL/SQL block that loops through all customers, checks their age, and if they are above 60, apply a 1% discount to their current loan interest rates.

Output ->



Scenario 2: A customer can be promoted to VIP status based on their balance.

 Question: Write a PL/SQL block that iterates through all customers and sets a flag IsVIP to TRUE for those with a balance over \$10,000.

Output ->

```
SQL> SELECT CUSTOMERNAME, AGE, LOANINTERESTRATE, BALANCE, ISVIP FROM CUSTOMERS;

CUSTOMERNAME

AGE LOANINTERESTRATE BALANCE ISVIP

John
65 6.5 12000 TRUE

Mary
45 10 8000 FALSE

Sita
70 5 15000 TRUE

CUSTOMERNAME

AGE LOANINTERESTRATE BALANCE ISVIP

AGE LOANINTERESTRATE BALANCE ISVIP

AGE LOANINTERESTRATE BALANCE ISVIP

ALI
30 12 2000 FALSE
```

Scenario 3: The bank wants to send reminders to customers whose loans are due within the next 30 days.

 Question: Write a PL/SQL block that fetches all loans due in the next 30 days and prints a reminder message for each customer.

Output->

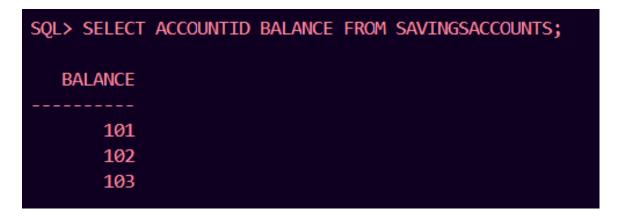
```
Reminder: Customer John (ID: 1) has Loan ID 201 due on 29-JUN-2025. 
PL/SQL procedure successfully completed.
```

Exercise 3: Stored Procedures

Scenario 1: The bank needs to process monthly interest for all savings accounts.

 Question: Write a stored procedure ProcessMonthlyInterest that calculates and updates the balance of all savings accounts by applying an interest rate of 1% to the current balance.

Output ->



Scenario 2: The bank wants to implement a bonus scheme for employees based on their performance.

 Question: Write a stored procedure UpdateEmployeeBonus that updates the salary of employees in a given department by adding a bonus percentage passed as a parameter.

Output->

SQL> SELECT EMPID, DEPARTMENT, SALARY FROM EMPLOYEES;	
EMPID DEPARTMENT	SALARY
1 HR	30000
2 IT	48400
3 IT	54450
4 Finance	38000

Scenario 3: Customers should be able to transfer funds between their accounts.

 Question: Write a stored procedure TransferFunds that transfers a specified amount from one account to another, checking that the source account has sufficient balance before making the transfer.

Output->

```
SQL> SELECT * FROM ACCOUNTS;

ACCOUNTID CUSTOMERNAME BALANCE

201 John 6000
202 Mary 8000
```

Module 4 – Test driven development and Logging framework

Output of all the testing files->

```
PS D:\JAVA FSE\Week 2\junitexample> mvn clean test
[INFO] Scanning for projects...
[INFO]
                 ----- com.example: junitexample >-----
INFO -----
[INFO] Building junitexample 1.0-SNAPSHOT
[INFO] from pom.xml
INFO -----[ jar ]------
[INFO]
[INFO] --- clean:3.1.0:clean (default-clean) @ junitexample ---
[INFO] Deleting D:\JAVA FSE\Week 2\junitexample\target
INFO
[INFO] --- resources:3.0.2:resources (default-resources) @ junitexample ---
Using 'UTF 8' encoding to copy filtered resources.
[INFO] skip non existing resourceDirectory D:\JAVA FSE\Week 2\junitexample\src\main\resources
[INFO] --- compiler:3.8.0:compile (default-compile) @ junitexample ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 2 source files to D:\JAVA FSE\Week 2\junitexample\target\classes
INFO
[INFO] --- resources:3.0.2:testResources (default-testResources) @ junitexample ---
[INFO] Using 'UTF-8' encoding to copy filtered resources.
[DNFO] skip non existing resourceDirectory D:\JAVA FSE\Week 2\junitexample\src\test\resources
INFO] --- compiler:3.8.0:testCompile (default-testCompile) @ junitexample ---
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 7 source files to D:\JAVA FSE\Week 2\junitexample\target\test-classes
[INFO] --- surefire:2.22.1:test (default-test) @ junitexample ---
INFO
INFO
```

```
[INFO] Running com.example.AAATest
Setup before test
Teardown after test
Setup before test
Teardown after test
[INFO] Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.009 s - in com.example.AAATest
[INFO] Running com.example.AppTest
[IMFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 s - in com.example.AppTest
[INFO] Running con.example.AssertionsTest
[INFO] Tests run: 1, Failures: 0, Frrors: 0, Skipped: 0, Time elapsed: 0 s - in con.example.AssertionsTest
[INFO] Running com.example.MyServiceTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.6/ s - in com.example.MyServiceTest
[INFO] Running MyFirstTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.001 s - in MyFirstTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 6, Failures: 0, Errors: 0, Skipped: 0
INFO
INFO --
[INFO] Total time: 5.821 s
[INFO] Finished at: 2025 06 28118:23:49:05:30
PS D:\JAWA FSE\Week 2\junitexample>
```

```
S D:\JAVA FSE\Week 2\junitexample> mvn exec:java
>>
[INFO] Scanning for projects...
INFO
INFO] ------
com.example:junitexample >-----
[INFO] Building junitexample 1.0-SNAPSHOT
[INFO] from pom.xml
[INFO] ------[ jar ]-----[
[INFO]
[INFO] --- exec:3.1.0:java (default-cli) @ junitexample ---
18:38:08.211 [com.example.LoggingExample.main()] ERROR com.example.LoggingExample - This is an error message 18:38:08.214 [com.example.LoggingExample.main()] WARN com.example.LoggingExample - This is a warning message
[INFO] ------
[INFO] BUILD SUCCESS
INFO] -----
INFO] Total time: 0.742 s
[INFO] Finished at: 2025-06-28T18:38:08+05:30
[INFO] ------
PS D:\JAVA FSE\Week 2\junitexample>
```