```
PROMPT PART A: Subquery-Based SQL Queries

PROMPT '1. Find out the name, phone_no and cust_no of customer having
Account_no "A0004"';
SELECT NAME, PHONE_NO, CUSTOMER.CUST_NO
FROM CUSTOMER
WHERE CUST_NO = (SELECT CUST_NO FROM DEPOSITOR WHERE ACCOUNT_NO =
'A0004');

PROMPT '2. Find out the name of the customer who has not taken any
loan';
SELECT NAME
FROM CUSTOMER
WHERE CUST_NO NOT IN (SELECT CUST_NO FROM LOAN);

PROMPT '3. Find out the account_no and Balance of customer with cust_no
= "C0010"';
SELECT ACCOUNT_NO, BALANCE
FROM ACCOUNT
WHERE ACCOUNT_NO IN (SELECT ACCOUNT_NO FROM DEPOSITOR WHERE CUST_NO =
'C0010');

PROMPT '4. Find out the branch_city where "ASLESHA TIWARI" has taken a
loan';
SELECT BRANCH_CITY
FROM BRANCH
WHERE BRANCH_CODE = (SELECT BRANCH_CODE FROM LOAN WHERE CUST_NO =
(SELECT CUST_NO FROM CUSTOMER WHERE NAME = 'ASLESHA TIWARI'));

PROMPT '5. Find out the installment details of customer named "ANKITA
SINGH"';
SELECT *
FROM INSTALLMENT
WHERE LOAN_NO IN (
    SELECT LOAN_NO FROM LOAN
    WHERE CUST_NO = (SELECT CUST_NO FROM CUSTOMER WHERE NAME = 'ANKITA
SINGH')
);

PROMPT '6. Find out the branch name and branch city, in which "ABHIJIT
MISHRA" has an account';
SELECT BRANCH_NAME, BRANCH_CITY
```

```sql
FROM BRANCH
WHERE BRANCH_CODE IN (
    SELECT BRANCH_CODE FROM ACCOUNT
    WHERE ACCOUNT_NO IN (
        SELECT ACCOUNT_NO FROM DEPOSITOR
        WHERE CUST_NO = (SELECT CUST_NO FROM CUSTOMER WHERE NAME =
'ABHIJIT MISHRA')
    )
);

PROMPT '7. Create a table named ACCOUNT_TYPE with ACCOUNT_NO and TYPE
only';
CREATE TABLE ACCOUNT_TYPE AS
SELECT ACCOUNT_NO, TYPE
FROM ACCOUNT
WHERE 1=0;

PROMPT '8. Insert into ACCOUNT_TYPE the accounts with balance < 50000';
INSERT INTO ACCOUNT_TYPE (ACCOUNT_NO, TYPE)
SELECT ACCOUNT_NO, TYPE FROM ACCOUNT WHERE BALANCE < 50000;

PROMPT '9. Update type to FD in ACCOUNT_TYPE for customer with CUST_NO
= C0007';
UPDATE ACCOUNT_TYPE
SET TYPE = 'FD'
WHERE ACCOUNT_NO IN (
    SELECT ACCOUNT_NO FROM DEPOSITOR WHERE CUST_NO = 'C0007'
);

PROMPT '10. Delete from ACCOUNT_TYPE the accounts with balance <
20000';
DELETE FROM ACCOUNT_TYPE
WHERE ACCOUNT_NO IN (
    SELECT ACCOUNT_NO FROM ACCOUNT WHERE BALANCE < 20000
);

PROMPT '11. Find account_no with balance greater than some FD
accounts';
SELECT ACCOUNT_NO
FROM ACCOUNT
WHERE BALANCE > SOME (SELECT BALANCE FROM ACCOUNT WHERE TYPE = 'FD');

PROMPT '12. Find account_no with balance greater than all FD accounts';
```

```
SELECT ACCOUNT_NO
FROM ACCOUNT
WHERE BALANCE > ALL (SELECT BALANCE FROM ACCOUNT WHERE TYPE = 'FD');


PROMPT '13. Branches where some loans are taken (EXISTS)';
SELECT * FROM BRANCH B
WHERE EXISTS (
    SELECT 1 FROM LOAN L WHERE L.BRANCH_CODE = B.BRANCH_CODE
);


PROMPT '14. Loans for which no installment is paid (NOT EXISTS)';
SELECT * FROM LOAN L
WHERE NOT EXISTS (
    SELECT 1 FROM INSTALLMENT I WHERE I.LOAN_NO = L.LOAN_NO
);


PROMPT '15. Increase balance with CASE';
UPDATE ACCOUNT
SET BALANCE = BALANCE +
    CASE
        WHEN BALANCE > 80000 THEN BALANCE * 0.06
        ELSE BALANCE * 0.05
    END;


PROMPT PART B: Join-Based SQL Queries

PROMPT '1. Loan_nos for loans from branches in MUMBAI';
SELECT LOAN_NO FROM LOAN
JOIN BRANCH ON LOAN.BRANCH_CODE = BRANCH.BRANCH_CODE
WHERE BRANCH_CITY = 'MUMBAI';


PROMPT '2. Account types in branches in DELHI';
SELECT DISTINCT TYPE FROM ACCOUNT
JOIN BRANCH ON ACCOUNT.BRANCH_CODE = BRANCH.BRANCH_CODE
WHERE BRANCH_CITY = 'DELHI';


PROMPT '3. Name and Ph_no of customers with account balance > 100000';
SELECT DISTINCT CUSTOMER.NAME, CUSTOMER.PHONE_NO
FROM CUSTOMER
JOIN DEPOSITOR ON CUSTOMER.CUST_NO = DEPOSITOR.CUST_NO
JOIN ACCOUNT ON DEPOSITOR.ACCOUNT_NO = ACCOUNT.ACCOUNT_NO
WHERE BALANCE > 100000;
```

```sql
PROMPT '4. Installments of customer RAJ ANAND SINGH';
SELECT INST_NO, INST_AMOUNT
FROM INSTALLMENT
JOIN LOAN ON INSTALLMENT.LOAN_NO = LOAN.LOAN_NO
JOIN CUSTOMER ON LOAN.CUST_NO = CUSTOMER.CUST_NO
WHERE NAME = 'RAJ ANAND SINGH';

PROMPT '5. Customers without SB type account';
SELECT DISTINCT NAME
FROM CUSTOMER
WHERE CUST_NO NOT IN (
    SELECT CUST_NO FROM DEPOSITOR D JOIN ACCOUNT A ON D.ACCOUNT_NO =
A.ACCOUNT_NO WHERE TYPE = 'SB'
);

PROMPT '6. Customers who paid 50000 as installment';
SELECT DISTINCT NAME
FROM CUSTOMER
JOIN LOAN ON CUSTOMER.CUST_NO = LOAN.CUST_NO
JOIN INSTALLMENT ON LOAN.LOAN_NO = INSTALLMENT.LOAN_NO
WHERE INST_AMOUNT = 50000;

PROMPT '7. Phone numbers of customers having account at SALTLAKE
branch';
SELECT PHONE_NO FROM CUSTOMER
JOIN DEPOSITOR ON CUSTOMER.CUST_NO = DEPOSITOR.CUST_NO
JOIN ACCOUNT ON DEPOSITOR.ACCOUNT_NO = ACCOUNT.ACCOUNT_NO
JOIN BRANCH ON ACCOUNT.BRANCH_CODE = BRANCH.BRANCH_CODE
WHERE BRANCH_NAME = 'SALTLAKE BRANCH';

PROMPT '8. Branch name and city where ABHIJIT MISHRA has an account';
SELECT BRANCH_NAME, BRANCH_CITY
FROM BRANCH
JOIN ACCOUNT ON BRANCH.BRANCH_CODE = ACCOUNT.BRANCH_CODE
JOIN DEPOSITOR ON ACCOUNT.ACCOUNT_NO = DEPOSITOR.ACCOUNT_NO
JOIN CUSTOMER ON DEPOSITOR.CUST_NO = CUSTOMER.CUST_NO
WHERE CUSTOMER.NAME = 'ABHIJIT MISHRA';

PROMPT '9. Account types and balance for SWAROOP RAY';
SELECT TYPE, BALANCE
FROM ACCOUNT
JOIN DEPOSITOR ON ACCOUNT.ACCOUNT_NO = DEPOSITOR.ACCOUNT_NO
JOIN CUSTOMER ON DEPOSITOR.CUST_NO = CUSTOMER.CUST_NO
```

```sql
WHERE CUSTOMER.NAME = 'SWAROOP RAY';


PROMPT '10. Branch codes with total balance > average total balance';
WITH BRANCH_BALANCE AS (
    SELECT BRANCH_CODE, SUM(BALANCE) AS TOTAL_BAL
    FROM ACCOUNT
    GROUP BY BRANCH_CODE
),
AVG_BALANCE AS (
    SELECT AVG(TOTAL_BAL) AS AVG_BAL FROM BRANCH_BALANCE
)
SELECT BRANCH_CODE
FROM BRANCH_BALANCE, AVG_BALANCE
WHERE TOTAL_BAL > AVG_BAL;
```