1. Table Definitions

```
CREATE TABLE BANK_BRANCH (
    BCODE VARCHAR(10) PRIMARY KEY,
    BNAME VARCHAR(100),
    CITY VARCHAR(50)
);

CREATE TABLE ACCOUNT_TYPE (
    TYPE_ID INT PRIMARY KEY,
    TYPE_NAME VARCHAR(50)
);

CREATE TABLE ACCOUNT (
    ACC_NO INT PRIMARY KEY,
    CUST_NAME VARCHAR(100),
    BALANCE DECIMAL(10, 2),
    BCODE VARCHAR(10),
    TYPE_ID INT,
    FOREIGN KEY (BCODE) REFERENCES BANK_BRANCH(BCODE),
    FOREIGN KEY (TYPE_ID) REFERENCES ACCOUNT_TYPE(TYPE_ID)
);
```

2. Sample Data

```
-- Branches
INSERT INTO BANK_BRANCH VALUES ('B001', 'MG Road Branch', 'Mumbai');
INSERT INTO BANK_BRANCH VALUES ('B002', 'Park Street Branch', 'Kolkata');
INSERT INTO BANK_BRANCH VALUES ('B003', 'Brigade Road Branch', 'Bangalore');

-- Account Types
INSERT INTO ACCOUNT_TYPE VALUES (1, 'Savings');
INSERT INTO ACCOUNT_TYPE VALUES (2, 'Current');
INSERT INTO ACCOUNT_TYPE VALUES (3, 'Fixed Deposit');

-- Accounts
INSERT INTO ACCOUNT VALUES (1001, 'Alice', 12000.00, 'B001', 1);
INSERT INTO ACCOUNT VALUES (1002, 'Bob', 5000.00, 'B001', 2);
INSERT INTO ACCOUNT VALUES (1003, 'Charlie', 30000.00, 'B002', 1);
INSERT INTO ACCOUNT VALUES (1004, 'David', 100000.00, 'B003', 3);
INSERT INTO ACCOUNT VALUES (1005, 'Eve', 15000.00, 'B001', 1);
```

3. SQL Query Examples

```
-- Number of Accounts per Branch
SELECT BB.BNAME, COUNT(A.ACC_NO) AS TOTAL_ACCOUNTS
FROM BANK_BRANCH BB
LEFT JOIN ACCOUNT A ON BB.BCODE = A.BCODE
GROUP BY BB.BNAME;

-- Total Balance per Branch
```

```
SELECT BB.BNAME, SUM(A.BALANCE) AS TOTAL_BALANCE
FROM BANK_BRANCH BB
JOIN ACCOUNT A ON BB.BCODE = A.BCODE
GROUP BY BB.BNAME;

-- Number of Accounts per Branch and Account Type
SELECT BB.BNAME, AT.TYPE_NAME, COUNT(*) AS NUM_ACCOUNTS
FROM ACCOUNT A
JOIN BANK_BRANCH BB ON A.BCODE = BB.BCODE
JOIN ACCOUNT_TYPE AT ON A.TYPE_ID = AT.TYPE_ID
GROUP BY BB.BNAME, AT.TYPE_NAME;

-- Branches with No Accounts
SELECT BB.BNAME
FROM BANK_BRANCH BB
LEFT JOIN ACCOUNT A ON BB.BCODE = A.BCODE
WHERE A.ACC_NO IS NULL;

-- Average Balance per Account Type
SELECT AT.TYPE_NAME, AVG(A.BALANCE) AS AVG_BALANCE
FROM ACCOUNT A
JOIN ACCOUNT_TYPE AT ON A.TYPE_ID = AT.TYPE_ID
GROUP BY AT.TYPE_NAME;

-- Highest Balance per Branch
SELECT BB.BNAME, MAX(A.BALANCE) AS MAX_BALANCE
FROM ACCOUNT A
JOIN BANK_BRANCH BB ON A.BCODE = BB.BCODE
GROUP BY BB.BNAME;

-- Customer Details by Account Type and Branch
SELECT A.CUST_NAME, BB.BNAME, AT.TYPE_NAME, A.BALANCE
FROM ACCOUNT A
JOIN BANK_BRANCH BB ON A.BCODE = BB.BCODE
JOIN ACCOUNT_TYPE AT ON A.TYPE_ID = AT.TYPE_ID
ORDER BY BB.BNAME, AT.TYPE_NAME, A.BALANCE DESC;
```

4. Advanced SQL

```
-- CTE: Top Customer by Balance per Branch
WITH RankedCustomers AS (
  SELECT A.CUST_NAME, BB.BNAME, A.BALANCE,
         RANK() OVER (PARTITION BY BB.BNAME ORDER BY A.BALANCE DESC) AS rnk
  FROM ACCOUNT A
  JOIN BANK_BRANCH BB ON A.BCODE = BB.BCODE
)
SELECT * FROM RankedCustomers WHERE rnk = 1;

-- Window Function: Running Total of Balance per Branch
SELECT A.CUST_NAME, BB.BNAME, A.BALANCE,
            SUM(A.BALANCE) OVER (PARTITION BY BB.BNAME ORDER BY A.BALANCE DESC) AS
RunningTotal
FROM ACCOUNT A
```

```
JOIN BANK_BRANCH BB ON A.BCODE = BB.BCODE;

-- Procedure: Deposit Amount
DELIMITER //
CREATE PROCEDURE DepositAmount(IN accNo INT, IN depositAmt DECIMAL(10,2))
BEGIN
  UPDATE ACCOUNT SET BALANCE = BALANCE + depositAmt WHERE ACC_NO = accNo;
END //
DELIMITER ;

-- Procedure: Withdraw Amount with Balance Check
DELIMITER //
CREATE PROCEDURE WithdrawAmount(IN accNo INT, IN withdrawAmt DECIMAL(10,2))
BEGIN
  DECLARE currentBal DECIMAL(10,2);
  SELECT BALANCE INTO currentBal FROM ACCOUNT WHERE ACC_NO = accNo;
  IF currentBal >= withdrawAmt THEN
    UPDATE ACCOUNT SET BALANCE = BALANCE - withdrawAmt WHERE ACC_NO = accNo;
  ELSE
    SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Insufficient balance';
  END IF;
END //
DELIMITER ;

-- Simulate Transfer Transaction
START TRANSACTION;
UPDATE ACCOUNT SET BALANCE = BALANCE - 1000 WHERE ACC_NO = 1001;
UPDATE ACCOUNT SET BALANCE = BALANCE + 1000 WHERE ACC_NO = 1002;
COMMIT;
```

5. Bank Manager Dashboard

```
-- Total Branches
SELECT COUNT(*) AS TOTAL_BRANCHES FROM BANK_BRANCH;

-- Total Customers
SELECT COUNT(*) AS TOTAL_CUSTOMERS FROM ACCOUNT;

-- Total Deposits
SELECT SUM(BALANCE) AS TOTAL_DEPOSITS FROM ACCOUNT;

-- Top 3 Customers by Balance
SELECT CUST_NAME, ACC_NO, BALANCE FROM ACCOUNT ORDER BY BALANCE DESC LIMIT 3;

-- Average Balance per Branch
SELECT BB.BNAME, ROUND(AVG(A.BALANCE), 2) AS AVG_BALANCE
FROM ACCOUNT A JOIN BANK_BRANCH BB ON A.BCODE = BB.BCODE GROUP BY BB.BNAME;

-- Account Distribution by Type
SELECT AT.TYPE_NAME, COUNT(*) AS NUM_ACCOUNTS
FROM ACCOUNT A JOIN ACCOUNT_TYPE AT ON A.TYPE_ID = AT.TYPE_ID
GROUP BY AT.TYPE_NAME;
```

```sql
-- Top Branch in Each City by Total Balance
WITH CityBranchSums AS (
  SELECT BB.CITY, BB.BNAME, SUM(A.BALANCE) AS TOTAL_BALANCE,
         RANK() OVER (PARTITION BY BB.CITY ORDER BY SUM(A.BALANCE) DESC) AS rnk
  FROM ACCOUNT A JOIN BANK_BRANCH BB ON A.BCODE = BB.BCODE
  GROUP BY BB.CITY, BB.BNAME
)
SELECT CITY, BNAME, TOTAL_BALANCE FROM CityBranchSums WHERE rnk = 1;
```