1. **List customers with a missing middle name.**

SELECT customer\_id, first\_name, last\_name

FROM Customers

WHERE middle\_name IS NULL;

**Output**:

| **customer\_id** | **first\_name** | **last\_name** |
| --- | --- | --- |
| 2 | Priya | Patel |
| 4 | Sneha | Singh |
| 7 | Ravi | Joshi |
| 9 | Sanjay | Reddy |
| 11 | Arun | Iyer |

1. **Find accounts with a balance greater than ₹1,00,000.**

SELECT account\_number, balance

FROM Accounts

WHERE balance > 100000;

**Output**:

| **account\_number** | **balance** |
| --- | --- |
| KOTAK123457 | 120000 |
| KOTAK123458 | 200000 |
| KOTAK123464 | 180000 |
| KOTAK123466 | 250000 |
| KOTAK123470 | 300000 |

1. **Show employees with an undefined position (NULL).**

SELECT employee\_id, first\_name, last\_name

FROM Employees

WHERE position IS NULL;

**Output**:

| **employee\_id** | **first\_name** | **last\_name** |
| --- | --- | --- |
| 3 | Rohan | Mehta |
| 5 | Vijay | Kumar |
| 9 | Rahul | Verma |
| 13 | Amit | Sharma |
| 16 | Anjali | Dubey |

1. **Count the number of approved loans.**

SELECT COUNT(\*) AS approved\_loans

FROM Loans

WHERE status = 'Approved';

**Output**:

| **approved\_loans** |
| --- |
| 10 |

1. **List transactions of type 'Deposit'.**

SELECT transaction\_id, account\_number, amount

FROM Transactions

WHERE type = 'Deposit';

**Output**:

| **transaction\_id** | **account\_number** | **amount** |
| --- | --- | --- |
|  |  |  |
| 1 | KOTAK123456 | 10000 |
| 4 | KOTAK123459 | 15000 |
| 7 | KOTAK123462 | 50000 |
| 10 | KOTAK123465 | 7000 |
| 13 | KOTAK123468 | 9000 |

1. **Find branches in Mumbai.**

SELECT \* FROM Branches

WHERE location LIKE '%Mumbai%';

**Output**:

| **branch\_id** | **branch\_name** | **location** |
| --- | --- | --- |
| 1 | Kotak Mumbai Main | Mumbai |

1. **Calculate the average salary of employees.**

SELECT AVG(salary) AS avg\_salary

FROM Employees;

**Output**:

| **avg\_salary** |
| --- |
| 47500.00 |

1. **List customers with alternate phone numbers (non-NULL).**

SELECT customer\_id, first\_name, phone, alt\_phone

FROM Customers

WHERE alt\_phone IS NOT NULL;

**Output**:

| **customer\_id** | **first\_name** | **phone** | **alt\_phone** |
| --- | --- | --- | --- |
| 2 | Priya | 9876543211 | 9876543212 |
| 5 | Vikram | 9876543215 | 9876543216 |
| 7 | Ravi | 9876543218 | 9876543219 |
| 10 | Pooja | 9876543222 | 9876543223 |
| 12 | Kavita | 9876543225 | 9876543226 |

1. **Show FD accounts opened after 2022-01-01.**

SELECT account\_number, opened\_date

FROM Accounts

WHERE account\_type = 'FD' AND opened\_date > '2022-01-01';

**Output**:

| **account\_number** | **opened\_date** |
| --- | --- |
| KOTAK123458 | 2022-03-10 |
| KOTAK123462 | 2022-02-05 |
| KOTAK123466 | 2022-06-15 |
| KOTAK123470 | 2022-04-20 |
| KOTAK123474 | 2022-07-05 |

1. **Find loans with no end date (NULL).**

SELECT loan\_id, customer\_id, amount

FROM Loans

WHERE end\_date IS NULL;

**Output**:

| **loan\_id** | **customer\_id** | **amount** |
| --- | --- | --- |
| 2 | 2 | 200000 |
| 4 | 4 | 1000000 |
| 5 | 5 | 150000 |
| 7 | 7 | 250000 |
| 9 | 9 | 100000 |

1. **Count transactions involving transfers.**

SELECT COUNT(\*) AS total\_transfers

FROM Transactions

WHERE type = 'Transfer';

**Output**:

| **total\_transfers** |
| --- |
| 8 |

1. **List employees hired in 2022.**

SELECT employee\_id, first\_name, hire\_date

FROM Employees

WHERE YEAR(hire\_date) = 2022;

**Output**:

| **employee\_id** | **first\_name** | **hire\_date** |
| --- | --- | --- |
| 5 | Vijay | 2022-01-30 |
| 12 | Mira | 2022-02-15 |
| 20 | Kiran | 2022-05-12 |

1. **Find the highest account balance.**

SELECT MAX(balance) AS max\_balance

FROM Accounts;

**Output**:

| **max\_balance** |
| --- |
| 300000.00 |

1. **List customers from Delhi or Mumbai.**

SELECT customer\_id, first\_name, address

FROM Customers

WHERE address LIKE '%Delhi%' OR address LIKE '%Mumbai%';

**Output**:

| **customer\_id** | **first\_name** | **address** |
| --- | --- | --- |
| 1 | Rahul | Mumbai |
| 2 | Priya | Delhi |

1. **Show loans with amounts between ₹2,00,000 and ₹5,00,000.**

SELECT loan\_id, customer\_id, amount

FROM Loans

WHERE amount BETWEEN 200000 AND 500000;

**Output**:

| **loan\_id** | **customer\_id** | **amount** |
| --- | --- | --- |
| 2 | 2 | 200000 |
| 3 | 3 | 300000 |
| 6 | 6 | 400000 |
| 10 | 10 | 350000 |
| 16 | 16 | 450000 |

**Join Questions (16-30)**

1. **List customers with their account numbers and types.**

SELECT c.first\_name, a.account\_number, a.account\_type

FROM Customers c

JOIN Accounts a ON c.customer\_id = a.customer\_id;

**Output**:

| **first\_name** | **account\_number** | **account\_type** |
| --- | --- | --- |
| Rahul | KOTAK123456 | Savings |
| Priya | KOTAK123457 | Current |
| Amit | KOTAK123458 | FD |

1. **Show transactions with customer names.**

SELECT t.transaction\_id, c.first\_name, t.amount

FROM Transactions t

JOIN Accounts a ON t.account\_number = a.account\_number

JOIN Customers c ON a.customer\_id = c.customer\_id;

**Output**:

| **transaction\_id** | **first\_name** | **amount** |
| --- | --- | --- |
| 1 | Rahul | 10000 |
| 2 | Priya | 5000 |
| 3 | Amit | 20000 |

1. **Find employees and their branch locations.**

SELECT e.first\_name, b.branch\_name

FROM Employees e

LEFT JOIN Branches b ON e.branch\_id = b.branch\_id;

**Output**:

| **first\_name** | **branch\_name** |
| --- | --- |
| Aarav | Kotak Mumbai Main |
| Isha | Kotak Delhi Central |

1. **List approved loans with customer details.**

SELECT c.first\_name, l.amount, l.type

FROM Loans l

JOIN Customers c ON l.customer\_id = c.customer\_id

WHERE l.status = 'Approved';

**Output**:

| **first\_name** | **amount** | **type** |
| --- | --- | --- |
| Rahul | 500000 | Home |
| Amit | 300000 | Car |
| Vikram | 150000 | Personal |

1. **Calculate total deposits per account.**

SELECT a.account\_number, SUM(t.amount) AS total\_deposits

FROM Transactions t

JOIN Accounts a ON t.account\_number = a.account\_number

WHERE t.type = 'Deposit'

GROUP BY a.account\_number;

**Output**:

| **account\_number** | **total\_deposits** |
| --- | --- |
| KOTAK123456 | 10000 |
| KOTAK123459 | 15000 |
| KOTAK123462 | 50000 |

1. **Show branches with no employees.**

SELECT b.branch\_name

FROM Branches b

LEFT JOIN Employees e ON b.branch\_id = e.branch\_id

WHERE e.employee\_id IS NULL;

**Output**:  
*(None in sample data)*

1. **List customers with their total account balances.**

SELECT c.customer\_id, c.first\_name, SUM(a.balance) AS total\_balance

FROM Customers c

JOIN Accounts a ON c.customer\_id = a.customer\_id

GROUP BY c.customer\_id;

**Output**:

| **customer\_id** | **first\_name** | **total\_balance** |
| --- | --- | --- |
| 1 | Rahul | 50000 |
| 2 | Priya | 120000 |
| 3 | Amit | 200000 |

1. **Find transactions with recipient account details (for transfers).**

SELECT t.transaction\_id, a.account\_type AS recipient\_type

FROM Transactions t

LEFT JOIN Accounts a ON t.related\_account = a.account\_number

WHERE t.type = 'Transfer';

**Output**:

| **transaction\_id** | **recipient\_type** |
| --- | --- |
| 3 | Savings |
| 5 | Current |
| 8 | Current |

1. **Show employees earning more than their branch’s average salary.**

SELECT e1.first\_name, e1.salary

FROM Employees e1

JOIN (

SELECT branch\_id, AVG(salary) AS avg\_salary

FROM Employees

GROUP BY branch\_id

) e2 ON e1.branch\_id = e2.branch\_id

WHERE e1.salary > e2.avg\_salary;

**Output**:

| **first\_name** | **salary** |
| --- | --- |
| Aarav | 75000 |
| Karan | 78000 |

1. **List customers with active loans and their loan types.**

SELECT c.first\_name, l.type

FROM Customers c

JOIN Loans l ON c.customer\_id = l.customer\_id

WHERE l.status = 'Approved';

**Output**:

| **first\_name** | **type** |
| --- | --- |
| Rahul | Home |
| Amit | Car |
| Vikram | Personal |

1. **Find accounts with no transactions.**

SELECT a.account\_number

FROM Accounts a

LEFT JOIN Transactions t ON a.account\_number = t.account\_number

WHERE t.transaction\_id IS NULL;

**Output**:  
*(None in sample data)*

1. **Show the latest transaction for each account.**

SELECT a.account\_number, MAX(t.transaction\_date) AS latest\_transaction

FROM Accounts a

LEFT JOIN Transactions t ON a.account\_number = t.account\_number

GROUP BY a.account\_number;

**Output**:

| **account\_number** | **latest\_transaction** |
| --- | --- |
| KOTAK123456 | 2023-10-05 10:00:00 |
| KOTAK123457 | * + 1. 1:00:00 |

1. **List branches with total employee salaries.**

SELECT b.branch\_name, SUM(e.salary) AS total\_salary

FROM Branches b

LEFT JOIN Employees e ON b.branch\_id = e.branch\_id

GROUP BY b.branch\_id;

**Output**:

| **branch\_name** | **total\_salary** |
| --- | --- |
| Kotak Mumbai Main | 75000 |
| Kotak Delhi Central | 35000 |

1. **Find customers with both Savings and FD accounts.**

SELECT c.customer\_id, c.first\_name

FROM Customers c

JOIN Accounts a1 ON c.customer\_id = a1.customer\_id AND a1.account\_type = 'Savings'

JOIN Accounts a2 ON c.customer\_id = a2.customer\_id AND a2.account\_type = 'FD';

**Output**:

| **customer\_id** | **first\_name** |
| --- | --- |
| 1 | Rahul |
| 3 | Amit |

1. **List employees and their managers (if manager\_id exists).**  
   *(Add a manager\_id column to the Employees table first.)*

tSELECT e1.first\_name AS employee, e2.first\_name AS manager

FROM Employees e1

LEFT JOIN Employees e2 ON e1.manager\_id = e2.employee\_id;