PostgreSQL SQL Masterplan

Audience: Intermediate-level Data Engineer in Finance

Duration: 30 Days (June 24 to July 23, 2025)

Goal: Master PostgreSQL from Basics to Advanced + Real-time Scenarios

---

▧ 1. Core SQL Foundations

1.1 SQL Basics

- SELECT, FROM, WHERE
- ORDER BY, LIMIT, OFFSET
- DISTINCT, BETWEEN, IN, LIKE
- Logical operators (AND, OR, NOT)
- Aliases

1.2 SQL Data Types

- Numeric: INT, BIGINT, NUMERIC, DECIMAL
- Character: CHAR, VARCHAR, TEXT
- Date/Time: DATE, TIMESTAMP
- Boolean, JSON/JSONB, ARRAY

1.3 Data Manipulation Language (DML)

- INSERT INTO
- UPDATE SET
- DELETE FROM
- RETURNING clause
- Transactions: BEGIN, COMMIT, ROLLBACK

1.4 Aggregate Functions & Grouping

- COUNT, SUM, AVG, MIN, MAX
- GROUP BY, HAVING
- Filtering aggregates

1.5 Joins

- INNER JOIN
- LEFT JOIN, RIGHT JOIN
- FULL OUTER JOIN
- CROSS JOIN
- SELF JOIN

1.6 Subqueries

- In SELECT, FROM, WHERE
- Correlated subqueries
- EXISTS, NOT EXISTS

🧱 2. Database Design

2.1 Table Design

- CREATE TABLE
- Column types, defaults
- Constraints: PRIMARY KEY, NOT NULL, UNIQUE
- AUTO_INCREMENT (SERIAL, IDENTITY)

2.2 Relationships & Integrity

- FOREIGN KEY constraints
- ON DELETE / ON UPDATE rules (CASCADE, RESTRICT, SET NULL)

2.3 Normalization

- 1NF, 2NF, 3NF
- Identify anomalies
- Denormalization trade-offs

2.4 Schema Planning

- Logical vs Physical design
- ERD modeling tools (drawSQL, dbdiagram.io)
- Schema versioning

🔧 3. Intermediate SQL Techniques

3.1 Views

- CREATE VIEW, REPLACE VIEW
- Updatable views
- Materialized views

3.2 Common Table Expressions (CTEs)

- WITH clause
- Recursive CTEs

3.3 Window Functions

- OVER(), PARTITION BY, ORDER BY
- ROW_NUMBER(), RANK(), DENSE_RANK()

- LAG(), LEAD(), FIRST_VALUE(), LAST_VALUE()
- Running totals, moving averages

3.4 Set Operations

- UNION vs UNION ALL
- INTERSECT
- EXCEPT

3.5 CASE Expressions

- Conditional logic inside SELECT
- CASE WHEN THEN ELSE END

⚙ 4. PostgreSQL-Specific Features

4.1 PostgreSQL Functions

- CREATE FUNCTION (SQL & PL/pgSQL)
- RETURNS VOID/INT/TABLE
- VOLATILE vs STABLE vs IMMUTABLE

4.2 Stored Procedures

- CREATE PROCEDURE (Postgres 11+)
- CALL vs SELECT usage

4.3 Triggers

- BEFORE/AFTER INSERT/UPDATE/DELETE
- Row vs statement level
- CREATE TRIGGER syntax

4.4 Advanced Data Types

- ENUM
- JSONB operations
- ARRAY and unnest()
- HSTORE

4.5 Extensions

- PostGIS (geospatial)
- pg_stat_statements (performance insights)

🚀 5. Optimization & Performance

5.1 Indexing

- B-tree, Hash, GIN, GiST
- CREATE INDEX, UNIQUE INDEX
- Partial Indexes
- Expression Indexes

5.2 Query Tuning

- EXPLAIN / EXPLAIN ANALYZE
- Query plan interpretation
- Optimizer hints

5.3 Vacuuming & Analyze

- VACUUM, VACUUM FULL
- ANALYZE
- Autovacuum tuning

5.4 Partitioning

- Range Partitioning
- List Partitioning
- Declarative Partitioning

🔒 6. Security & Access Control

6.1 Role Management

- CREATE ROLE / USER
- GRANT / REVOKE permissions
- Role attributes: LOGIN, SUPERUSER, etc.

6.2 Row-Level Security (RLS)

- CREATE POLICY
- ENABLE ROW LEVEL SECURITY
- Security barrier views

🧪 7. Real-time Finance Scenarios

7.1 Use Case: Credit Risk Reporting

- Table design: customers, accounts, loans, repayments
- Queries: late payment identification, NPA calculation

7.2 Use Case: Investment Portfolio

- Schema for holdings, stocks, transactions
- Queries: portfolio valuation, gain/loss, top performers

7.3 Use Case: Transaction Monitoring

- Trigger on suspicious transactions
- Daily/monthly aggregation reports
- Time-window functions for anomaly detection

🗓️ 30-Day Learning Plan

Day Range	Focus Area
Days 1-3	Core SQL (SELECT, WHERE, JOINs)
Days 4-6	DML, Aggregate Functions, Subqueries
Days 7-9	Database Design & Constraints
Days 10-11	Normalization + Schema Planning
Days 12-13	Views, CTEs, CASE, Set Ops
Days 14-15	Window Functions + Hands-on Exercises
Days 16-18	PostgreSQL Functions, Triggers
Days 19-21	Indexing + Performance Tuning
Days 22-23	Security & Access Management
Days 24-27	Real-world Use Case Projects
Days 28-30	Final Project, Optimization, Presentation

Let me know if you want this split into separate PDF files, add images/ERDs, or include guided exercises with solutions for each topic.