🗄️ ORACLE DATABASE: COMPLETE TECHNICAL GUIDE

Prepared on January 30, 2026 | Version 26ai Compatible

📌 TABLE OF CONTENTS

---

1. DATABASE FUNDAMENTALS

What is a Database?

An **organized collection of interrelated data** designed for efficient storage, retrieval, management, and security.

Industry Domain	Core Entities (Tables)	Relationships
Banking	Customers, Accounts, Transactions, Loans, Employees	Customer → Accounts → Transactions
University	Students, Courses, Faculty, Departments, Enrollments	Student ↔ Courses via Enrollments
E-Commerce	Products, Orders, Customers, Payments, Inventory	Order links Customer + Products + Payments

Database Types Comparison

Type	Full Form	Primary Use Case	Characteristics	Example Operations
OLTP	Online Transaction Processing	Real-time operational systems	• High concurrency • Short transactions • ACID compliance • Row-level operations	ATM withdrawal, Order placement
OLAP	Online Analytical Processing	Business intelligence & reporting	• Complex queries • Historical analysis • Columnar storage	Sales trend analysis, Forecasting

Type	Full Form	Primary Use Case	Characteristics	Example Operations
			• Aggregation-heavy	
HTAP	Hybrid Transactional/Analytical Processing	Modern real-time analytics	• Unified OLTP + OLAP • In-memory processing • Zero-latency analytics	Real-time fraud detection

💡 **2026 Insight**: Oracle 26ai introduces **Autonomous HTAP** – automatically routes queries to optimized execution engines without schema changes.

---

2. DBMS EVOLUTION TIMELINE

Evolution of Database Management Systems

Era	Technology	Key Characteristics	Limitations
1960s	File Management System (FMS)	Flat files, sequential access	• Data redundancy • No relationships • Application-dependent
1970s	Hierarchical DBMS (HDBMS)	Tree structure (parent-child)	• Rigid structure • Complex navigation • IBM IMS example

Era	Technology	Key Characteristics	Limitations
1970s	Network DBMS (NDBMS)	Graph structure (many-to-many)	• Complex schema design • CODASYL standard
1980s	Relational DBMS (RDBMS)	Tables with rows/columns	• E.F. Codd's 12 rules • SQL standard • Set-based operations
1990s	Object-Relational DBMS (ORDBMS)	RDBMS + OOP features	• Type extensibility • Inheritance • Collections

E.F. Codd's 12 Rules (RDBMS Foundation)

Rule #	Rule Name	Description
0	Foundation Rule	System must qualify as relational as a database and as a management system
1	Information Rule	All information represented as values in tables
2	Guaranteed Access	Every value accessible via table name + primary key + column name
3	Systematic Null	Supports NULL for missing/unknown data
4	Active Catalog	Database description stored in tables (data dictionary)
5	Comprehensive Sublanguage	SQL must support DDL, DML, DCL, views

Rule #	Rule Name	Description
6	View Updating	Theoretically updatable views must be updatable
7	High-Level Insert/Update/Delete	Set-level operations, not record-at-a-time
8	Physical Data Independence	Apps unaffected by storage changes
9	Logical Data Independence	Apps unaffected by logical schema changes
10	Integrity Independence	Integrity constraints in catalog, not apps
11	Distribution Independence	Apps unaware of data distribution
12	Non-Subversion Rule	Low-level interfaces can't bypass integrity rules

---

3. RDBMS CORE CONCEPTS

Relational Model Structure

Key Constraints

Constraint	Purpose	Example	Enforcement
Primary Key	Uniquely identify rows	empid, aadharno, panno	NOT NULL + UNIQUE
Foreign Key	Enforce referential integrity	emp.projid → projects.projid	CASCADE options
Unique	Prevent duplicate values	email, phone	Allows NULLs
Check	Domain validation	age > 18, gender IN ('M','F')	Application-level

Constraint	Purpose	Example	Enforcement
Not Null	Mandatory values	empname, doj	Column-level

Relationship Types (with Examples)

Relationship	Description	Example Tables	Implementation
One-to-Many	One parent → Many children	DEPT → EMP	FK in child table
Many-to-Many	Many parents ↔ Many children	STUDENTS ↔ COURSES	Junction table (ENROLLMENTS)
One-to-One	One parent ↔ One child	EMP ↔ EMP_DETAILS	PK = FK or unique FK

---

4. ORDBMS ARCHITECTURE

RDBMS vs ORDBMS: Address Reusability Example

Approach	Schema Design	Drawbacks
Pure RDBMS	CUST(cid, cname, hno, street, city, state, pin) EMP(empid, ename, hno, street, city, state, pin)	• Data redundancy • Schema changes require multiple table modifications • No code reuse
ORDBMS	ADDR_TYPE(hno, street, city, state, pin) CUST(cid, cname, addr ADDR_TYPE)	• Reusable object type • Single change propagates everywhere • Supports inheritance

Approach	Schema Design	Drawbacks
	EMP(empid, ename, addr ADDR_TYPE)	

ORDBMS Features Supported by Oracle

- ✅ Abstract Data Types (ADTs)
- ✅ Type inheritance & method overriding
- ✅ Collections (VARRAY, Nested Tables)
- ✅ Large Objects (BLOB, CLOB, BFILE)
- ✅ SQL/MM Spatial & Multimedia types

---

5. DATABASE DEVELOPMENT LIFECYCLE

Six-Phase Lifecycle with Roles & Deliverables

Phase	Key Activities	Responsible Roles	Deliverables
Analysis	Requirements gathering, business rules	Business Analysts	BRD, Use Cases
Design	ER Modeling, Normalization, Schema design	DB Architects	ER Diagrams, DFDs
Development	DDL creation, PL/SQL coding	DB Developers	Tables, Views, Procedures

Phase	Key Activities	Responsible Roles	Deliverables
Testing	Unit testing, performance validation	QA Engineers	Test cases, Reports
Deployment	Migration to production environment	DBAs	Deployment scripts
Maintenance	Monitoring, tuning, backups	DBAs	Performance reports

Normalization Levels (Design Best Practice)

Normal Form	Rule	Eliminates
1NF	Atomic values, no repeating groups	Repeating groups
2NF	1NF + no partial dependencies	Partial dependencies
3NF	2NF + no transitive dependencies	Transitive dependencies
BCNF	Every determinant is candidate key	Remaining anomalies
4NF	No multi-valued dependencies	MVD anomalies
5NF	No join dependencies	Join anomalies

⚠️ **Practical Tip**: Most production systems normalize to **3NF**, denormalize selectively for performance.

---

6. ORACLE ECOSYSTEM

Oracle Database Versions Timeline

Version	Year	Key Innovation	Target Workload
Oracle 2-7	1979-1992	Foundational RDBMS	OLTP
Oracle 8i	1999	Internet-ready, Java in DB	Web apps
Oracle 9i	2001	RAC (Real Application Clusters)	High availability
Oracle 10g	2003	Grid computing, ASM	Scalability
Oracle 11g	2007	Total Recall, Compression	Enterprise
Oracle 12c	2013	Multitenant architecture	Cloud consolidation
Oracle 18c/19c	2018-2019	Autonomous features	Self-driving DB
Oracle 21c	2021	Blockchain tables, JSON Relational Duality	Modern apps
Oracle 23ai	2023	AI Vector Search, Property Graph	AI/ML workloads
Oracle 26ai	2026	**Autonomous HTAP**, GenAI SQL Assistant	Real-time analytics + AI

Deployment Models (12c+)

Model	Description	Management Responsibility	Use Case
On-Premises	Installed on client-owned servers	Client IT team	Strict compliance requirements

Model	Description	Management Responsibility	Use Case
Cloud (DBaaS)	Hosted on Oracle Cloud Infrastructure	Oracle + Client shared	Rapid scalability
Autonomous DB	Self-driving, self-securing, self-repairing	Fully Oracle-managed	Minimize DBA overhead

---

7. CLIENT-SERVER ARCHITECTURE

Oracle Server Components

Client Tools Comparison

Tool	Type	Features	Best For
SQL*Plus	CLI	Lightweight, scriptable	DBAs, automation
SQL Developer	GUI	Visual query builder, ER diagrams	Developers
Oracle Enterprise Manager	Web GUI	Performance monitoring, tuning	Production DBAs
APEX	Low-code	App development without coding	Citizen developers

---

8. SQL LANGUAGE STRUCTURE

SQL Sublanguages & Commands

Sublanguage	Purpose	Key Commands	Transaction Scope
DDL (Data Definition Language)	Define/modify schema	CREATE, ALTER, DROP, TRUNCATE, RENAME	Auto-commit
DML (Data Manipulation Language)	Manipulate data	INSERT, UPDATE, DELETE, MERGE	Requires COMMIT
DQL/DRL (Data Query/Retrieval Language)	Query data	SELECT	Read-only
TCL (Transaction Control Language)	Manage transactions	COMMIT, ROLLBACK, SAVEPOINT	Transaction boundary
DCL (Data Control Language)	Security management	GRANT, REVOKE	Auto-commit

Transaction Behavior Comparison

Command	Undoable?	Releases Locks?	Frees Space?	DDL Equivalent?
DELETE	✅ Yes (until COMMIT)	❌ No	❌ No (high water mark unchanged)	❌ No
TRUNCATE	❌ No	✅ Yes	✅ Yes (resets HWM)	✅ Yes (DDL)
DROP	❌ No	✅ Yes	✅ Yes (removes structure)	✅ Yes (DDL)

---

9. ORACLE DATATYPES REFERENCE

Character Datatypes

Datatype	Max Size	Storage Behavior	Use Case	Unicode Support
CHAR(n)	2,000 bytes	Fixed-length (pads with spaces)	Fixed codes (gender, state)	ASCII only
VARCHAR2(n)	4,000 bytes	Variable-length (no padding)	Names, addresses, descriptions	ASCII only
NCHAR(n)	2,000 chars	Fixed-length Unicode	Multilingual fixed codes	✅ UTF-16
NVARCHAR2(n)	4,000 chars	Variable-length Unicode	Multilingual text	✅ UTF-16
CLOB	128 TB	Character large object	Documents, JSON	ASCII
NCLOB	128 TB	Unicode large object	Multilingual documents	✅ UTF-16
LONG	2 GB	Legacy (avoid)	Deprecated	❌ Avoid

⚠️ **Best Practice**: Prefer VARCHAR2 over CHAR for variable-length data to save space. Use CLOB instead of LONG.

Numeric Datatypes

Datatype	Precision	Scale	Use Case	Example Values
NUMBER(p,s)	1-38 digits	-84 to 127	Flexible numeric storage	NUMBER(10,2) → 12345678.99
NUMBER(p)	1-38 digits	0 (integer)	Whole numbers	NUMBER(5) → 12345
NUMBER	38 digits	Floating	Unconstrained	1.2345678901234567890123456789012345 6789
BINARY_FLOAT	32-bit	IEEE 754	Scientific calculations	Fast but approximate
BINARY_DOUBLE	64-bit	IEEE 754	High-precision science	More precise than FLOAT

Date/Time Datatypes

Datatype	Precision	Storage	Timezone Support	Use Case
DATE	Seconds	7 bytes	❌ No	Birthdays, hire dates
TIMESTAMP	Nanoseconds	7-11 bytes	❌ No	Event logging
TIMESTAMP WITH TIME ZONE	Nanoseconds	13 bytes	✅ Yes	Global applications

Datatype	Precision	Storage	Timezone Support	Use Case
TIMESTAMP WITH LOCAL TIME ZONE	Nanoseconds	7-11 bytes	✅ Session-based	User-local timestamps
INTERVAL YEAR TO MONTH	Year-month	5 bytes	N/A	Duration (e.g., employment period)
INTERVAL DAY TO SECOND	Day-second	8-11 bytes	N/A	Precise durations

Binary/Large Object Datatypes

Datatype	Max Size	Storage Location	Use Case
BLOB	128 TB	Inside database	Images, videos, PDFs
BFILE	OS file size	Outside database (OS file)	Large read-only files
RAW	2,000 bytes	Binary data	Encrypted values
LONG RAW	2 GB	Legacy (avoid)	Deprecated

---

10. USER & SCHEMA MANAGEMENT

Schema Conceptual Model

User Management Commands

Operation	Syntax	Required Privilege	Notes
Create User	CREATE USER username IDENTIFIED BY password DEFAULT TABLESPACE users QUOTA UNLIMITED ON users;	DBA	Always set tablespace quota
Grant Basic Privs	GRANT CONNECT, RESOURCE TO username;	DBA	CONNECT = login, RESOURCE = create objects
Grant Full DBA	GRANT DBA TO username;	DBA	Use cautiously in production
Change Own Pwd	PASSWORD (in SQL*Plus)	Self	Interactive prompt
Change Other Pwd	ALTER USER username IDENTIFIED BY new_password;	DBA	No old password required
Reset SYS Pwd	ALTER USER SYS IDENTIFIED BY new_password; (as SYSDBA)	SYSDBA	Emergency recovery only
Drop User	DROP USER username [CASCADE];	DBA	CASCADE required if schema has objects

Critical Security Best Practices (2026)

- 🔒 **Never use default passwords** (TIGER, CHANGE_ON_INSTALL)
- 🔒 **Apply Principle of Least Privilege** – grant only required privileges
- 🔒 **Use profiles** to enforce password complexity & account locking

- 🔒 **Enable auditing** for privileged operations (AUDIT CREATE USER)
- 🔒 **Rotate passwords** quarterly for production accounts
- 🔒 **Use Oracle Key Vault** for credential management in cloud deployments

---

11. 2026 BEST PRACTICES & INNOVATIONS

Oracle 26ai New Features

Feature	Benefit	Use Case
GenAI SQL Assistant	Natural language → SQL conversion	Business users writing queries
Autonomous HTAP	Real-time analytics on transactional data	Fraud detection without ETL
Vector Search	AI-powered similarity search	Product recommendations
JSON Relational Duality	Single view for JSON + relational	Modern app development
In-Memory Auto-Optimization	Self-tuning column store	Analytics acceleration

Performance Checklist

- ✅ Use bind variables to avoid hard parsing
- ✅ Create indexes on FK columns & frequent WHERE clauses
- ✅ Partition large tables (>10M rows) by date/range
- ✅ Use EXPLAIN PLAN before deploying complex queries

- ✅ Avoid SELECT * – fetch only required columns
- ✅ Use AUTOTRACE for execution statistics

Migration Path Recommendations

Current Version	Recommended Target	Rationale
11g or earlier	19c (Long Term Support)	Extended support until 2027
12c/18c/19c	23ai	AI/ML capabilities, JSON enhancements
21c/23ai	26ai	GenAI integration, Autonomous HTAP

📚 APPENDIX: QUICK REFERENCE CHEAT SHEET

Common SQL*Plus Commands

Command	Description
CONNECT username/password	Login to database
SHOW USER	Display current user
DESC table_name	Describe table structure
SET LINESIZE 200	Wider output display
SET PAGESIZE 100	More rows per page
SPOOL filename.log	Save session output
EXIT	Disconnect and exit

Default Oracle Accounts (Lock Unused!)

Account	Purpose	Status Recommendation
SYS	Data dictionary owner	Keep locked except maintenance
SYSTEM	Administrative objects	Keep locked except maintenance
SCOTT	Sample schema (Tiger)	**LOCK** in production
HR	Human Resources demo	**LOCK** in production
OE	Order Entry demo	**LOCK** in production