🧮1️⃣ Numeric Functions

Function	Description	Example
`ABS(n)`	Absolute value	`ABS(-5)` → `5`
`CEIL(n)`	Smallest integer ≥ n	`CEIL(4.2)` → `5`
`FLOOR(n)`	Largest integer ≤ n	`FLOOR(4.8)` → `4`
`MOD(m,n)`	Remainder of m/n	`MOD(10,3)` → `1`
`POWER(m,n)`	m raised to n	`POWER(2,3)` → `8`
`ROUND(n[,dec])`	Round to dec places	`ROUND(123.456,2)` → `123.46`
`TRUNC(n[,dec])`	Truncate to dec places	`TRUNC(123.456,2)` → `123.45`
`SIGN(n)`	Sign of n (-1,0,1)	`SIGN(-12)` → `-1`
`SQRT(n)`	Square root	`SQRT(16)` → `4`
`EXP(n)`	e^n	`EXP(1)` → `2.71828`
`LN(n)`	Natural log	`LN(10)`
`LOG(b,n)`	Log base b of n	`LOG(10,100)` → `2`
`REMAINDER(m,n)`	IEEE remainder	`REMAINDER(10,3)` → `1`

🔤2️⃣ Character / String Functions

Function	Description	Example
`INITCAP(str)`	Capitalize first letter of each word	`INITCAP('hello world')` → `Hello World`
`LOWER(str)`	Convert to lowercase	`LOWER('SQL')` → `sql`

`UPPER(str)`	Convert to uppercase	`UPPER('sql')` → `SQL`
`LENGTH(str)`	String length	`LENGTH('Oracle')` → `6`
`INSTR(str, sub [,start [,occurrence]])`	Find position	`INSTR('DATABASE','A')` → `2`
`SUBSTR(str, start [,length])`	Extract substring	`SUBSTR('DATABASE',1,4)` → `DATA`
`LPAD(str, len, pad)`	Left pad	`LPAD('45',5,'0')` → `00045`
`RPAD(str, len, pad)`	Right pad	`RPAD('45',5,'0')` → `45000`
`LTRIM(str [,set])`	Trim left chars	`LTRIM('$$DATA','$')` → `DATA`
`RTRIM(str [,set])`	Trim right chars	`RTRIM('DATA$$','$')` → `DATA`
`TRIM([LEADING | TRAILING | BOTH] chr FROM str)`		
`REPLACE(str, search [,replace])`	Replace substring	`REPLACE('abc','a','z')` → `zbc`
`TRANSLATE(str, from, to)`	Translate chars	`TRANSLATE('12345','15','AB')` → `A234B`
`CONCAT(a,b)`	Concatenate strings	`CONCAT('Hello','World')` → `HelloWorld`
`		`

📅 3 Date and Time Functions

Function	Description	Example
`SYSDATE`	Current date and time	`SELECT SYSDATE FROM DUAL;`
`CURRENT_DATE`	Current date (session TZ)	`SELECT CURRENT_DATE FROM DUAL;`

`CURRENT_TIMESTAMP`	Timestamp with time zone	`SELECT CURRENT_TIMESTAMP FROM DUAL;`
`ADD_MONTHS(date, n)`	Add n months	`ADD_MONTHS(SYSDATE,2)`
`MONTHS_BETWEEN(d1,d2)`	Months difference	`MONTHS_BETWEEN(SYSDATE, '2025-01-01')`
`NEXT_DAY(date, 'DAY')`	Next given weekday	`NEXT_DAY(SYSDATE,'MONDAY')`
`LAST_DAY(date)`	Last day of month	`LAST_DAY(SYSDATE)`
`ROUND(date [,fmt])`	Round date	`ROUND(SYSDATE,'MONTH')`
`TRUNC(date [,fmt])`	Truncate date	`TRUNC(SYSDATE,'YEAR')`
`EXTRACT(YEAR	MONTH	DAY FROM date)`
`TO_CHAR(date, fmt)`	Convert date to string	`TO_CHAR(SYSDATE,'DD-MON-YYYY')`
`TO_DATE(str, fmt)`	Convert string to date	`TO_DATE('12-11-2025','DD-MM-YYYY')`
`TO_TIMESTAMP(str, fmt)`	Convert to timestamp	`TO_TIMESTAMP('2025-11-12 10:00:00','YYYY-MM-DD HH24:MI:SS')`

🧰4️⃣ Conversion Functions

Function	Description	Example
`TO_CHAR(expr [,fmt])`	Convert to string	`TO_CHAR(1234,'$9,999')`
`TO_NUMBER(str [,fmt])`	Convert to number	`TO_NUMBER('1234')`
`TO_DATE(str [,fmt])`	Convert to date	`TO_DATE('2025-11-12','YYYY-MM-DD')`

`TO_TIMESTAMP(str[,fmt])`	Convert to timestamp	`TO_TIMESTAMP('12-NOV-2025 10:00','DD-MON-YYYY HH24:MI')`
`CAST(expr AS datatype)`	Type conversion	`CAST('123' AS NUMBER)`

🧩5️⃣ Aggregate Functions

Function	Description	Example
`COUNT(expr)`	Number of rows	`COUNT(*)`
`SUM(expr)`	Sum of values	`SUM(salary)`
`AVG(expr)`	Average value	`AVG(salary)`
`MIN(expr)`	Minimum value	`MIN(salary)`
`MAX(expr)`	Maximum value	`MAX(salary)`
`STDDEV(expr)`	Standard deviation	`STDDEV(salary)`
`VARIANCE(expr)`	Variance	`VARIANCE(salary)`
`LISTAGG(expr, delimiter)`	Concatenate within group	`LISTAGG(name, ',') WITHIN GROUP (ORDER BY name)`

⚖️6️⃣ Conditional / Comparison Functions

Function	Description	Example
`NVL(expr1, expr2)`	Replace NULL	`NVL(comm,0)`
`NVL2(expr1, expr2, expr3)`	If expr1 NOT NULL → expr2 else expr3	`NVL2(comm,sal+comm,sal)`
`NULLIF(expr1, expr2)`	Return NULL if equal	`NULLIF(sal,0)`

`COALESCE(expr1, expr2, …)`	First non-NULL value	`COALESCE(comm,bonus,0)`
`DECODE(expr, val1, res1, …, default)`	Conditional logic	`DECODE(deptno,10,'Sales',20,'IT','Other')`
`CASE WHEN cond THEN val ELSE val END`	Conditional branching	`CASE WHEN sal>5000 THEN 'High' ELSE 'Low' END`

🔥7 Analytical / Window Functions

Function	Description	Example
`ROW_NUMBER()`	Row sequence	`ROW_NUMBER() OVER (ORDER BY sal DESC)`
`RANK()`	Rank with gaps	`RANK() OVER (ORDER BY sal DESC)`
`DENSE_RANK()`	Rank without gaps	`DENSE_RANK() OVER (ORDER BY sal DESC)`
`NTILE(n)`	Divide rows into n buckets	`NTILE(4) OVER (ORDER BY sal)`
`LEAD(expr [,offset,default])`	Next row value	`LEAD(sal,1,0) OVER (ORDER BY sal)`
`LAG(expr [,offset,default])`	Previous row value	`LAG(sal,1,0) OVER (ORDER BY sal)`
`FIRST_VALUE(expr)`	First in window	`FIRST_VALUE(sal) OVER (ORDER BY sal)`
`LAST_VALUE(expr)`	Last in window	`LAST_VALUE(sal) OVER (ORDER BY sal ROWS BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING)`
`SUM(expr) OVER (...)`	Running total	`SUM(sal) OVER (ORDER BY sal)`

`AVG(expr) OVER (...)`	Moving average	`AVG(sal) OVER (PARTITION BY deptno)`

🧮8️⃣ Miscellaneous Functions

Function	Description	Example
`USER`	Current DB user	`SELECT USER FROM DUAL;`
`UID`	Current user ID	`SELECT UID FROM DUAL;`
`SYS_CONTEXT(namespace, param)`	Session/env info	`SYS_CONTEXT('USERENV','IP_ADDRESS')`
`DUMP(expr)`	Data type and internal info	`DUMP('A')`
`VSIZE(expr)`	Bytes used	`VSIZE('Oracle')`

🧾9️⃣ Dual Table Usage

Oracle provides a dummy table `DUAL` for evaluating functions:

```
SELECT SYSDATE, USER, ABS(-5), UPPER('oracle') FROM DUAL;
```