1. GETDATE()

Purpose: Retrieves the current system date and time.

Syntax:

```sql
GETDATE()
```

Example:

```sql
SELECT GETDATE() AS CurrentDateTime; -- Output: 2024-12-09 14:25:30.567 (Varies based on current date and time)
```

---

2. DATEADD()

Purpose: Adds or subtracts a specific number of time units (days, months, years, etc.) to/from a given date.

Syntax:

```sql
DATEADD(datepart, number, date)
```

- `datepart` : The part of the date (e.g., year, month, day).
- `number` : The value to add or subtract (negative for subtraction).

Example:

```sql
SELECT DATEADD(DAY, 7, GETDATE()) AS SevenDaysLater; -- Output: 2024-12-16 14:25:30.567 (Adds 7 days to the current date) SELECT DATEADD(MONTH, -2, GETDATE()) AS TwoMonthsAgo; -- Output: 2024-10-09 14:25:30.567 (Subtracts 2 months from the current date)
```

---

3. DATEDIFF()

Purpose: Calculates the difference between two dates based on the specified date part.

Syntax:

```sql
DATEDIFF(datepart, startdate, enddate)
```

- `datepart` : The unit of difference (e.g., day, month, year).

Example:

```sql
SELECT DATEDIFF(DAY, '2024-12-01', GETDATE()) AS DaysDifference; -- Output: 8 (Difference in days between Dec 1, 2024, and today) SELECT DATEDIFF(YEAR, '2000-01-01', GETDATE()) AS YearsDifference; -- Output: 24 (Difference in years from Jan 1, 2000, to today)
```

---

4. DAY()

Purpose: Extracts the day of the month from a given date.

Syntax:

```sql
DAY(date)
```

Example:

```sql
SELECT DAY(GETDATE()) AS DayOfMonth; -- Output: 9 (Today's day of the month is 9)
```

---

5. MONTH()

Purpose: Extracts the month from a given date.

Syntax:

```sql
MONTH(date)
```

Example:

```sql
SELECT MONTH(GETDATE()) AS Month; -- Output: 12 (Current month is December)
```

---

6. YEAR()

Purpose: Extracts the year from a given date.

Syntax:

```sql
YEAR(date)
```

Example:

```sql
SELECT YEAR(GETDATE()) AS Year; -- Output: 2024 (Current year is 2024)
```

---

7. DATEPART()

Purpose: Extracts a specified part of a date (e.g., year, month, day, week, etc.).

Syntax:

```sql
DATEPART(datepart, date)
```

Example:

```sql
```

```sql
SELECT DATEPART(WEEKDAY, GETDATE()) AS DayOfWeek; -- Output: 2 (Monday is represented as 2, depending on the SQL server configuration) SELECT DATEPART(HOUR, GETDATE()) AS Hour; -- Output: 14 (Current hour of the day is 14, i.e., 2 PM)
```

---

8. CONVERT()

Purpose: Converts a date into a specified format or datatype.

Syntax:

```sql
CONVERT(datatype, expression, style)
```

- `style` : Specifies the output format (e.g., 10 for `MM-DD-YY` , 110 for `MM-DD-YYYY`).

Examples:

```sql
SELECT CONVERT(VARCHAR(19), GETDATE()) AS DefaultFormat; -- Output: Dec 9 2024 14:25PM (Default style without specifying format) SELECT CONVERT(VARCHAR(19), GETDATE(), 10) AS MMDDYY; -- Output: 12-09-24 (Style 10: MM-DD-YY) SELECT CONVERT(VARCHAR(19), GETDATE(), 110) AS MMDDYYYY; -- Output: 12-09-2024 (Style 110: MM-DD-YYYY)
```

---

9. FORMAT()

Purpose: Formats a date value as a string in a specified format.

Syntax:

```sql
FORMAT(expression, format)
```

Examples:

```sql
SELECT FORMAT(GETDATE(), 'D') AS LongDate; -- Output: Monday, December 9, 2024 (Formatted in a long date format) SELECT FORMAT(GETDATE(), 'dd-MM-yyyy HH:mm') AS
```

```
CustomFormat; -- Output: 09-12-2024 14:25 (Custom date-time format)
```

Use Case Example

Consider a table `Orders` with a column `OrderDate`:

OrderID	OrderDate
101	2024-11-20 10:30:00
102	2024-12-01 14:45:00
103	2024-12-07 09:20:00

Query:

sql Copy code

```sql
SELECT OrderID, DAY(OrderDate) AS DayOfMonth, MONTH(OrderDate) AS Month,
YEAR(OrderDate) AS Year, FORMAT(OrderDate, 'dddd') AS DayName, DATEDIFF(DAY,
OrderDate, GETDATE()) AS DaysSinceOrder FROM Orders;
```

Output:

OrderID	DayOfMonth	Month	Year	DayName	DaysSinceOrder
101	20	11	2024	Wednesday	19
102	1	12	2024	Sunday	8
103	7	12	2024	Saturday	2

This example shows how to extract various date parts and calculate the difference between dates.