```
/*
	============================
	== Date & Time Functions ==
	============================
	- Date Format: 2025-08-20
	- Time Format: 18:55:45
	- DateTime: 2025-08-20  18:55:45

	- GETDATE(): returns the current system date and time of the SQL server

	| Format | Meaning      |
	| ------ | ------------ |
	| `ddd`  | Wed          |
	| `MMM`  | Jan          |
	| `yyyy` | 2025         |
	| `hh`   | 12-hour time |
	| `mm`   | minutes      |
	| `ss`   | seconds      |
	| `tt`   | AM / PM      |

*/

USE SalesDB;

SELECT
	OrderID,
	OrderDate,
	ShipDate,
	CreationTime
FROM Sales.Orders;


-- get current date and time
SELECT GETDATE() AS 'current DateTime';

/*
	-- insert the current date and time
	USE MyDatabase;
	INSERT INTO orders (order_date)
	VALUES (GETDATE());
*/

-- get the only date
SELECT CAST(GETDATE() AS DATE);


-- get the only time
SELECT CAST(GETDATE() AS TIME);


/*
	======================
	== Date Extraction ==
	======================
	DAY(): returns the day from a date
	MONTH(): returns the month from a date
	YEAR(): returns the year from a date
*/
```

```sql
USE SalesDB;
SELECT
	CreationTime,
	YEAR(CreationTime) AS 'Year',
	MONTH(CreationTime) AS 'Month',
	DAY(CreationTime) AS 'Day'
FROM Sales.Orders;


/*
	DATEPART(part, date): returns a specific part of a date as a number
	part: refers to the which part you need to extract like year, month, day, ...
	date: refers to the date or the column that contains the date
*/
SELECT
	OrderDate,
	DATEPART(YEAR, OrderDate) AS 'Year',
	DATEPART(MONTH, OrderDate) AS 'Month',
	DATEPART(DAY, OrderDate) AS 'Day',
	DATEPART(WEEK, OrderDate) AS 'Week'
FROM Sales.Orders;


-- also can use the abbreviation of part
SELECT
	CreationTime,
	DATEPART(YY, CreationTime) AS 'Year',
	DATEPART(MM, CreationTime) AS 'Month',
	DATEPART(DD, CreationTime) AS 'Day',
	DATEPART(HOUR, CreationTime) AS 'Hour',
	DATEPART(MINUTE, CreationTime) AS 'Minute',
	DATEPART(SECOND, CreationTime) AS 'Seconds',
	DATEPART(MILLISECOND, CreationTime) AS 'Milli Seconds',
	DATEPART(MICROSECOND, CreationTime) AS 'Micro Seconds'
FROM Sales.Orders;


-- Get the Quarter
SELECT
	CreationTime,
	DATEPART(QUARTER, CreationTime) AS 'Quarter'
FROM Sales.Orders;


-- get the weekday
SELECT
	CreationTime,
	DATEPART(WEEKDAY, CreationTime) AS 'Weekday'
FROM Sales.Orders;


/*
	DATENAME(part, date): returns the name of a specific part of a date
	the data type of the output is string
*/
SELECT
	CreationTime,
	DATENAME(MONTH, CreationTime) AS 'Month_DateName',
```

```sql
	DATENAME(WEEKDAY, CreationTime) AS 'Weekday_DateName'
FROM Sales.Orders;


/*
	DATETRUNC(part, date): truncate the date to the specific part, will keep the
part then reset the others values
	part: when define the part on DATRTRUNC() function, anything after that will
be reset
	if determine the part as MONTH, will reset the DAY as '01', because of there
is no DAY like 00
	if determine the part as YEAR, will reset the MONTH as '01', because of there
is no MONTH like 00
*/
SELECT
	CreationTime,
	DATETRUNC(YEAR, CreationTime) AS 'Hour'
FROM Sales.Orders;


/*
	count the number of orders based on the 'CreationTime'
	- will get one everywhere, because the level of details 'CreationTime' is
different
*/
SELECT
	CreationTime,
	COUNT(*) 'Number Of Orders'
FROM Sales.Orders
GROUP BY CreationTime;


-- aggregate the data based on the month level
SELECT
	DATETRUNC(MONTH, CreationTime) AS 'Month Level',
	COUNT(*) AS 'Number Of Orders'
FROM Sales.Orders
GROUP BY DATETRUNC(MONTH, CreationTime);


/*
	EOMONTH(date): return the last day of a month
	date: refers to the date or the column that contains the date
*/
SELECT
	CreationTime,
	EOMONTH(CreationTime) AS 'LastDay'
FROM Sales.Orders;


-- get the last day of the month
SELECT
	OrderDate,
	EOMONTH(OrderDate) AS 'LastDay'
FROM Sales.Orders;


-- get the start of month and end of month
```

```sql
SELECT
	CreationTime,
	DATETRUNC(MONTH, CreationTime) AS 'StartOfMonth',
	EOMONTH(CreationTime) AS 'EndOfMonth'
FROM Sales.Orders;


/*
	CAST(): used to convert the data types
*/
SELECT
	CreationTime,
	DATETRUNC(MONTH, CreationTime) AS 'StartOfDate',
	CAST(DATETRUNC(MONTH, CreationTime) AS DATE) AS 'StartOfData'
FROM Sales.Orders;

-- how many orders were placed each month
SELECT
	DATENAME(MONTH, OrderDate) AS OrderDate,
	COUNT(*) 'Number Of Orders'
FROM Sales.Orders
GROUP BY DATENAME(MONTH, OrderDate);


-- This query calculates how many orders were placed each month by grouping
SELECT
    YEAR(OrderDate) AS OrderYear,
    MONTH(OrderDate) AS OrderMonth,
    DATENAME(MONTH, OrderDate) AS MonthName,
    COUNT(*) AS NumberOfOrders
FROM Sales.Orders
GROUP BY
    YEAR(OrderDate),
    MONTH(OrderDate),
    DATENAME(MONTH, OrderDate)
ORDER BY
    OrderYear,
    OrderMonth;

-- show all orders that were placed during the month of february
SELECT *
FROM Sales.Orders
WHERE MONTH(OrderDate) = 2;


/*
	==============================
	== Date Formatting & Casting ==
	==============================
	- formatting: changing the format of a value from one to another (how the
data looks like)
	the formats can be as the following
		MONTH-DAY-YEAR
		DAY-MONTH-YEAR
	------------
	-- Syntax --
	------------
	FORMAT(value, format)
```

```sql
	- casting: convert the data type from one to another (string -> integer)
*/
SELECT
	OrderDate,

	-- change the arrange of the Date
	FORMAT(OrderDate, 'dd/MM/yy') AS 'Order Date',

	-- get the short name of the day
	FORMAT(OrderDate, 'ddd') AS 'Day',

	-- get the full name of the day
	FORMAT(OrderDate, 'dddd') AS 'Day',

	-- get the month as number
	FORMAT(OrderDate, 'MM') AS 'Month',

	-- get the short name of the month
	FORMAT(OrderDate, 'MMM') AS 'Month',

	-- get the full name of the month
	FORMAT(OrderDate, 'MMMM') AS 'Month',

	-- get the year from the date as number
	FORMAT(OrderDate, 'yy') AS 'Year',

	-- get the short name of the year
	FORMAT(OrderDate, 'yyy') AS 'Year'
FROM Sales.Orders;

/*
	show the CreationTime using the following format:
	Day Wed Jan Q1 2025 12:34:56 PM
*/
SELECT
    CreationTime,
    CONCAT(
        'Day ',
        FORMAT(CreationTime, 'ddd MMM '),
        'Q', DATEPART(QUARTER, CreationTime), ' ',
        FORMAT(CreationTime, 'yyyy hh:mm:ss tt')
    ) AS [Creation Time]
FROM Sales.Orders;
```