

Exercise : DATE Functions

Table1: Employees

emp_id	name	hire_date
1	Alice	2020-01-15
2	Bob	2021-06-10
3	Charlie	2023-03-22

Q1 → Add 6 Months to each employee's hire date using DATEADD()

* SELECT emp_id, name, hire_date, DATEADD(MONTH, 6, hire_date)
AS hire_plus_6_months
FROM Employees;

emp_id	name	hire_date	hireplus 6 months
1	Alice	2020-01-15	2020-07-15
2	Bob	2021-06-10	2021-12-10
3	Charlie	2023-03-22	2023-09-22

Table 2 : Students

student_id	name	dob
101	Maya	2005-08-10
102	Ethan	2004-12-01
103	Senna	2006-03-15

Q2 → use DateDiff() to find age in days from dob to day.

* SELECT student_id, name, DateDiff(day, dob, CURRENT_DATE()) AS age_in_days
FROM students;

student_id	name	age_in_days
101	Maya	7,248
102	Ethan	7,503
103	Senna	7,008

Table 3 : Events

event_id	event_name	event_date
1	Seminar	2024-06-15
2	Workshop	2025-09-01
3	Hackathon	2025-01-20

Q3 → find how many days are left until each event using DATEDIFF()

4 SELECT event_id, event_name, DATEDIFF(DAY, CURRENT_DATE(), event_date) AS days_remaining
FROM events;

event_id	event_name	days remaining
1	Seminar	-376
2	Workshop	67
3	Hackathon	-157

Table 4 : Invoices

Invoice_Id	Issue_date	Due_date
I01	2025-03-10	2025-03-25
I02	2025-04-01	2025-04-15
I03	2025-04-10	2025-04-20

Q4 → calculate the number of days between issue_date and due_date

SELECT invoice_id, issue_date, due_date, DATEDIFF(DAY, issue_date, due_date) AS days_between
FROM invoices;

Invoice_Id	Issue_date	Due_date	days between
I01	2025-03-10	2025-03-25	15
I02	2025-04-01	2025-04-15	14
I03	2025-04-10	2025-04-20	10

Table 5 : Courses

Course_id	name	start_date
201	SQL Basics	2025-05-01
202	Python	2025-06-01

Q5 → format start_date as 'Month YYYY' using To_char()

+ SELECT course_id, name, TO_CHAR(start_date, 'MONTH YYYY') AS formatted_date
FROM Courses;

course_id	name	formatted_date
201	SQL Basics	May 2025
202	Python	JUNE 2025

Table 6 : Memberships

member_id	start_year	start_month	start_day
1	2023	05	10
2	2022	11	25

Q6 → Create full date from parts using DATE_FROM_PARTS()

+ SELECT member_id, DATEFROMPARTS(start_year, start_month, start_day)
AS full_start_date
FROM Members

member_id	full_start_date
1	2023-05-10
2	2022-11-25

Table 7 : Subscriptions

Sub_id	plan	renewal_date
11	Basic	2025-01-01
12	Premium	2025-03-15

Q7 → Extend each renew_date by 1 year using DATEADD()

* SELECT Sub_id, plan, DATEADD(YEAR, 1, renewal_date) AS extended_renewal_date
FROM Subscriptions;

Sub_id	Plan	Extended_renewal_date
11	BASIC	2026-01-01
12	Premium	2026-03-15

Table 8: orders

order_id	order_date
1001	2025-04-15
1002	2025-04-10

Q8 → Show current date and difference ~~from~~ order_date
use CURRENT_DATE and DATEDIFF().

* SELECT order_id, order_date, CURRENT_DATE AS today_date,
DATEDIFF(CURRENTDATE(), order_date) AS days_since_order
FROM orders;

order_id	order_date	today_date	days_since_order
1001	2025-04-15	2025-06-26	72
1002	2025-04-10	2025-06-26	77

Table 9: Trainers

training_id	topic	training_date
1	Safety	2025-01-10
2	Compliance	2025-02-20

Q9 → Extract year from training_date using DATE_PART() or EXTRACT()

* SELECT training_id, topic, DATEPART(YEAR, training_date) AS training_year
FROM Trainings

training_id	topic	training_year
1	safety	2025
2	compliance	2025

Table 10 : Blog_Posts

post_id	title	published_on
1	SQL Tips	2025-04-10 10:15:00
2	Date Cleaning	2025-04-12 16:45:00

Q10 → Extract hour and minute from published_on

* SELECT post_id, title, DATEPART(HOUR, published_on) AS hour_published, DATEPART(MINUTE, published_on) AS minute_published
FROM Blog_Posts;

post_id	title	hour_published	minute_published
1	SQL tips	10	15
2	Date cleaning	16	45

Q11 : Drivers

driver_id	license_expiry
301	2025-08-10
302	2023-12-31

Q11 → calculate days left until license expiry using DATEDIFF() and today's date

* SELECT driver_id, license_expiry, DATEDIFF(DAY, GETDATE(), license_expiry) AS days_left
FROM Drivers;

driver_id	license_expiry	days_left
301	2025-08-15	45
302	2023-12-31	-543

Table 12 : Messages

message_id	sent_timestamp
1	2025-04-19 09:32:45
2	2025-04-18 23:59:59

Q12 → Display the current timestamp and calculate seconds since the message was sent.

* SELECT message_id, sent_timestamp, GETDATE() AS current_timestamp,
DATEDIFF(SECOND, sent_timestamp, GETDATE()) AS seconds_since_sent
FROM messages;

message_id	sent_timestamp	current_timestamp	seconds_since_sent
1	2025-04-19 09:32:45	2025-06-26 10:00:00	5695800
2	2025-04-18 23:59:59	2025-06-26 10:00:00	5718001

Q13 : Returns

return_id	return_date
901	2025-04-05
902	2025-04-01

Q13 → Add 15 days to return_date using DATEADD() to show restock_date

* SELECT return_id, return_date, DATEADD(DAY, 15, return_date)
AS restock_date
FROM Returns;

return_id	return_date	restock_date
901	2025-04-05	2025-04-20
902	2025-04-01	2025-04-16

Table 14 : Assignments

assign_id	assigned_on
1	2025-03-01
2	2025-03-05

assign_id	assigned_on_date
1	2025-03-01
2	2025-03-05

Q14 → convert assigned_on to date using TO_DATE() if it's stored as string()

* SELECT assign_id, TO_DATE(assigned_on, 'YYYY-MM-DD') AS assigned_on_date
FROM Assignments

Q15 : Meetings

meeting_id	scheduled_time
1	2025-04-19 14:00:00
2	2025-04-19 09:30:00

Meeting_id	formatted_meeting_time
1	April 19, 2025 at 2:00 PM
2	April 19, 2025 at 9:30 AM

Q15 → convert scheduled_time to formatted string like 'April 19, 2025' at 2:00 PM using TO_CHAR()

* SELECT meeting_id, TO_CHAR(scheduled_time, 'MONTH DD, YYYY "at" HH:MI AM') AS formatted_meeting_time
FROM Meetings;