BRIGHTLIGHT TUTORIALS

DATA ANALYTICS

EXERCISE 5: DATE FUNCTIONS

Table 1 : Employees

Q1 : Add 6 months to each employee's hire date using DATEADD ()

Syntax :

```
 SELECT  emp_id,
         name,
         hire_date,
DATEADD ( MONTH, 6, hire_date) AS hire_date_
 plus_6_moltths
FROM Employees;
```

Output :

```
emp_id   name     hire_date     hire_date_plus_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

Q2 : Use DATEDIFF() to find age in days from dob to today

Syntax :

```
 SELECT  student_id,
            name,
DATEDIFF ( DAY, dob, CURRENT_DATE ) AS
 age_in_days
 FROM student;
```

Table 3 : Events

Q3 : Find how many days are left until each event using DATEDIFF ()

Syntax :

```
 SELECT  event_id,
         event_name,
DATEDIFF (DAY, CURRENT_DATE, event_date)
 As days_remaining
FROM  Events;
```

Output :

```
event_id   event_name   days_remaining
   1       Seminar       -493
   2       Workshop      ~~348~~ -50
   3       Hackathon     -293
```

Table 4 : Invoices

Q4 : Calculate the number of days between Issue_date and due_date

Syntax :

```
 SELECT  invoice_id,
         issue_date,
         due_date,
DATEDIFF (DAY, Issue_date, due_date) AS
days_between
FROM  Invoices;
```

Output :

```
Invoice_id  Issue_date   due_date     days_between
   501      2025-03-10   2025-03-25      15
   502      2025-04-01   2025-04-15      14
   503      2025-04-10   2025-04-20      10
```

Table 5: courses

Q5: Format start_date as 'Month YYYY' using TO_CHAR()

Syntax:

```
 SELECT  course_id,
          name,
TO_CHAR (start_date, 'Month YYYY') AS
 formatted_date
 FROM  courses;
```

Output:

```
course_id    name         formatted_date
   201       SQL Basics     May 2025
   202       Python         June 2025
```

Table 6: Memberships

Q6: Create full date from parts using DATE_FROM_PARTS ()

Syntax:

```
 SELECT   member_id,

DATE_FROM_PARTS (start_year, start_month,
start_day) AS  full_start_date
FROM  Memberships;
```

Output:

```
member_id       Full_start_date
    1             2023-05-10
    2             2022-11-25
```

Table 7: Subscriptions

Q7: Extend each renewal_date by 1 year using DATEADD()

Syntax :

```
SELECT  sub_id,
          plan,
DATEADD (YEAR, 1, renewal_date ) AS  extended_
renewal_date
FROM  subscriptions;
```

Output :

```
        sub_id   plan     extended_renewal_date
          11     Basic         2026-01-01
          12     Premium       2026-03-15
```

Table 8 : Orders

Q8. : Show current date and difference from order_date

Syntax :

```
SELECT  order_id,
          order_date,
 CURRENT_DATE  AS  today_date,
DATEDIFF (DAY, order_date, CURRENT_DATE)
 AS   days_since_order
FROM  orders;
```

Output ;

```
order_id   order_date   today_date   days_since_order
  1001     2025-04-15   2025-10-20        188
  1002     2025-04-10   2025-10-20        193
```

Table 9 : Trainings

Q9 : Extract the year from training_date using DATE_PART () or EXTRACT ()

Syntax :

```
SELECT training_id,
       topic,
EXTRACT (YEAR FROM training_date) AS
training_year
FROM Trainings;
```

Output :

```
training_id   topic        training_year
    1         Safety          2025
    2         Compliance      2025
```

Table 10 : Blog_Posts

Q10 : Extract hour and minute from published_on

Syntax :

```
SELECT post_id,
       title,
EXTRACT (HOUR FROM published_on) AS hour_published
EXTRACT (MINUTE FROM published_on) AS minute_published
FROM Blog_Posts;
```

Output :

```
post_id   title           hour_published   Minute_published
  1       SQL Tips             10                15
  2       Data Cleaning        16                45
```

Table 11 : Drivers

Q11 : Calculate days left until license expires

Syntax :

```
SELECT driver_id,
        license-expiry,
DATEDIFF (DAY, CURRENT_DATE, license-expiry)
 AS days-left
FROM Drivers;
```

```
Output: driver-id  license_expiry   days_left
          301       2025-08-10        -71
          302       2023-12-31        -659
```

Table 12: Messages

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

Syntax:

```
SELECT message-id,
        sent_timestamp,
CURRENT_TIMESTAMP AS current_timestamp,
DATEDIFF (SECOND, sent_timestamp, CURRENT_TIMESTAMP)
AS seconds_since_sent
FROM Messages;
```

Output:

```
message_id  sent_timestamp       current_timestamp     seconds_since
                                                       - sent
    1       2025-04-19 09:32:45  2025-10-20 02:53:04   15724819
    2       2025-04-18 23:59:59  2025-10-20 02:53:04   15728425
```

Table 13: Returns

Q13: Add 15 day to return_date using DATEADD() to show restock_date

Syntax:

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

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

Table 14 : Assignments

Q14 : Convert assigned_on to date using TO_DATE(

Syntax :

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

```
Output :   assign_id        assigned_on_date
              1                2025-03-01
              2                2025-03-05
```

Table 15 : Meetings

Q15 : Convert scheduled_time to formatted string !

Syntax :

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

Output :

```
meeting_id      formatted_meeting_time
    1           April 19, 2025 at 02.00 PM
    2           April 19, 2025 at 09:30 AM
```