

Name: Mah Rukh

Roll No: 22L-6702

Section: BCS-4J

Assignment 1

Instructor: Ms. Aleena Ahmad

Course: Database Systems (CS-2005)

Submission Deadline: 18-02-2024

**Part 1:**

Comparative analysis of Date (MySQL, Oracle, SQL Server) data types, related functions and usage in different forms in queries.

**Date:**

Storing and manipulating date data is crucial in various database applications. This analysis compares the date data types, related functions, and usage in queries across three popular database systems: MySQL, Oracle, and SQL Server.

**Data Types:**

|  |  |  |  |
| --- | --- | --- | --- |
| *Features* | *MySQL* | *Oracle* | *SQL Server* |
| Supported data types | Date  Datetime  Timestamp  Year  Time | Date  Timestamp  Timestamp with time zone  Timestamp with local time zone  Interval day to second  Interval year to month | Date  Time  Datetime  Datetime2  Datetimeoffset  Smalldatetime |
| Precision | Date: year and day  Datetime/Timestamp: year, month, day, hour, minute, second | Date: year and day Timestamp: year, month, day, hour, minute, second, fractions of seconds  Timestamp with time zone/ local time zone: year, month, day, hour, minute, second, fractions of seconds, time zone | Date: year and day Time: hour, minute, second  Datetime: year, month, day, hour, minute, second DateTime2: year, month, day, hour, minute, second, fractions of seconds Datetimeoffset: year, month, day, hour, minute, second, fractions of seconds, time zone |
| Storage Size(Bytes) | 3 | 7(variable) | 8 |

**Functions:**

|  |  |  |  |
| --- | --- | --- | --- |
| *Feature* | *MySQL* | *Oracle* | *SQL Server* |
| Date Arithmetic | Date\_add (date, interval days Day)  Date\_sub (date, interval days Day) | Add\_months, To\_date,  Date+INTERVAL days DAY, Date-INTERVAL days DAY | Dateadd(day, days, date),  Datediff(day, date, other\_date) |
| Date Formatting | Date\_format,  Str\_to\_Date | To\_char,  To\_date | Format,  Cast |
| Extract Components | Year(date), Month(date), Day(date) | Extract (day/month/year from date) ,  Trunc | Year(date), Month(date), Day(date) |
| Time zone Conversion | No built-in function | Timestamp with time zone, utldate | Convert,  At time zone |
| Get Current Date | Now (),  CurDate () | Sysdate,  Current\_date | GetDate () |
| Get Current Time | Now (),  Curtime () | Sysdate,  Current\_date,  Current\_TimeStamp | GetUtcDate,  GetDate () |
| Parse String | Str\_to\_date | To\_date | Convert,  Parse |
| Format Date | Date\_format | To\_char | Format,  Convert |
| Add/Subtract Time | Date\_add,  Date\_sub | Add\_months  Add\_days | DateAdd,  DatePart |

**Usage in Queries:**

|  |  |  |  |
| --- | --- | --- | --- |
| *Feature* | *MySQL* | *Oracle* | *SQL Server* |
| Date Filtering | WHERE order\_date BETWEEN '2023-12-01' AND '2023-12-31' | WHERE order\_date >=TO\_DATE('2023-12-01', 'YYYY-MM-DD') AND order\_date < TO\_DATE('2024-01-01', 'YYYY-MM-DD'); | WHERE order\_date >= '2023-12-01' AND order\_date < '2024-01-01'; |
| Date Calculations | SELECT YEAR(order\_date) AS order\_year FROM orders | SELECT EXTRACT(YEAR FROM order\_date) AS order\_year FROM orders; | SELECT YEAR(order\_date) AS order\_year FROM orders |
| Grouping by Date | SELECT date(order\_date), COUNT(\*) AS orders\_count FROM orders GROUP BY date(order\_date) | SELECT DATETRUNC('day', order\_date), COUNT(\*) AS orders\_count FROM orders GROUP BY DATETRUNC('day', order\_date); | SELECT CONVERT(DATE, order\_date), COUNT(\*) AS orders\_count FROM orders GROUP BY CONVERT(DATE, order\_date); |
| Select all records from an year | SELECT \* FROM your\_table WHERE YEAR(date\_column) = year | SELECT \* FROM your\_table WHERE EXTRACT(YEAR FROM date\_column) = year; | SELECT \* FROM your\_table WHERE YEAR(date\_column) = year; |
| Add 30 days to a date | UPDATE your\_table SET date\_column = DATE\_ADD(date\_column, INTERVAL 30 DAY) | UPDATE your\_table SET date\_column = date\_column + INTERVAL 30 DAY; | UPDATE your\_table SET date\_column = DATEADD(day, 30, date\_column); |
| Select records from last week | SELECT \* FROM your\_table WHERE date\_column >= CURDATE() - INTERVAL 7 DAY; | SELECT \* FROM your\_table WHERE date\_column >= SYSDATE - INTERVAL 7 DAY; | SELECT \* FROM your\_table WHERE date\_column >= GETDATE() - INTERVAL 7 DAY; |

All three systems allow using date/time data in *WHERE* clauses for filtering data based on date/time ranges. *JOIN* operations can be performed on date/time columns to relate tables based on date/time relationships. Functions like *GROUP BY* and *ORDER BY* can be used with date/time columns for data aggregation and sorting.