

SQL Revision

Working with Date Time Function :

1. **Dayname** : select dayname (column_name)
2. **Monthname** : select monthname (column_name)
3. **Year** : select year (column_name)
4. **Day of week** : select dayofweek (column_name)
5. **Day of month** : select dayofmonth (column_name)
6. **Day of year** : select dayofyear (column_name)
7. **Datediff** (To subtract the 2 dates) : select datediff (column1,column2)
8. **Date_add/Date_sub** : select date_add (column_name, interval 10 years/month/hour/minute etc)
9. **For Today's Date** : curdate() or current_date ()

Working with String Function :

1. **Concating** : concat (column1 , " ", column2)
2. **Upper_case** : ucase (column_name)
3. **Replace** : replace (column_name, "what to replace" , "with what")
4. **Substring** : substring (column_name , from what postion, till which position)
5. **Sub string index** :

To extraxt the string before and after specific symbol

```
select substring_index(column_name,'symbol',-1/1) as "s"
from user_profile;
# here if you want before the symbol then use 1 and after the symbol then use -1
```

Control Function :

1. **Case - when** :

```
select case
    when column_name > 20000 then "Excellent"
    when column_name > 10000 then "good"
    when column_name < 10000 then "Ok"
    else "na"
end as salary_cat
```

1. **If** : if (column_name > 1000, "if true", "if false")

Windows Function :

1. **Rank/dense_rank/row_number** :

```
select rank/dense_rank/row_number() over ( order by column_name asc/desc )
```

1. **Aggregation windows Function (with partition by and order by** rows betn)
 - o sum()
 - o max()
 - o min()
 - o avg()
 - o count()

```
select sum ( column_name ) over (partition by ___ order by ___ rows between unounded/no preceding and current row / unbounded following /
no following
```

Current row, preceding, following

1. **Nth, previous and next value** :

- o First_value () : first_value (column) over (partition by __ order by __)
- o Nth_value () : nth_value (column, position) over (partition by __ order by __)
- o lead / lag : lead/lag (column name , how much lead , "n/a) over (partition by __ order by __)

Other :

1. **Common Table Expression** :

With table_name as (entire SQL query)

select *

from table_name

1. **ifnull**(column_name, "__")
2. where column is null / is not null

DDL Commands :

```
Create table_name (  
column1 varchar (250),  
column2 int  
)
```

```
Insert into table_name (column1,column2)  
values  
("value",__),  
("value", __);
```

```
Update table_name  
set column_name = "__"  
where column_name = "__" ;
```

```
Alter table table_name  
rename column old_name to new_name
```

```
alter table table name  
add column_name varchar (),int ;
```