**HIVE**

CREATE TABLE weather\_report

(location STRING, position FLOAT, localTime TIMESTAMP, conditions STRING , temperature FLOAT , pressure FLOAT , humidity INT)

ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde'

WITH SERDEPROPERTIES (

   "separatorChar" = "\t",

   "quoteChar"     = "'",

   "escapeChar"    = "\\"

)

STORED AS TEXTFILE;

LOAD DATA LOCAL INPATH ‘C:\\Users\\Dell\\Desktop\\WeatherReport.psv’ OVERWRITE INTO TABLE weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, MAX(temperature) over (partition by location) FROM weather\_report WHERE temperature != 9999 AND quality IN (0, 1, 4, 5, 9);

SELECT location, position, localTime, conditions, temperature, pressure, humidity, MIN(temperature) over (partition by location) FROM weather\_report WHERE temperature != 9999 AND quality IN (0, 1, 4, 5, 9);

SELECT location, position, localTime, conditions, temperature, pressure, humidity, AVG(temperature) over (partition by location) FROM weather\_report WHERE temperature != 9999 AND quality IN (0, 1, 4, 5, 9)

SELECT location, position, localTime, conditions, temperature, pressure, humidity, MAX(pressure) over (partition by location) FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, MIN(pressure) over (partition by location) FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, AVG(pressure) over (partition by location) FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, MAX(humidity) over (partition by location) FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, MIN(humidity) over (partition by location) FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, AVG(humidity) over (partition by location) FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, RANK() over(partition by location order by temperature) Rank From weather\_report;

SELECT wd.location, wd.position, wd.localTime, wd.conditions, wd.temperature, wd.pressure, wd.humidity, FROM (SELECT location, position, localTime, conditions, temperature, pressure, humidity, RANK() over(partition by location order by temperature desc) Rank FROM weather\_report) wd where wd.rank =1 ;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, DENSE\_RANK() over(partition by location order by temperature) Rank FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, ROW\_NUMBER() over(partition by location order by temperature) Rank FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, RANK() over(partition by location order by temperature), DENSE\_RANK() over(partition by location order by temperature), ROW\_NUMBER() over(order by temperature desc) ROWNUM FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, LEAD(temperature) over (partition by location order by temperature) FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, LEAD(temperature,2) over (partition by location order by temperature) FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, LAG(pressure) over (partition by location order by pressure), LAG(pressure,2) OVER (PARTITION BY location order by pressure) FROM weather\_report;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, FIRST\_VALUE(conditions) over (partition by location order by temperature ) as firstValue\_temp FROM student;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, LAST\_VALUE(conditions) over (partition by location order by temperature ) as lastvalue\_temp FROM student;

SELECT location, position, localTime, conditions, temperature, pressure, humidity, COUNT(conditions) OVER(partition by location order by location) as count\_conditions FROM weather\_report where conitions=Snow;