Air Pollution in the USA From 2016 to 2021

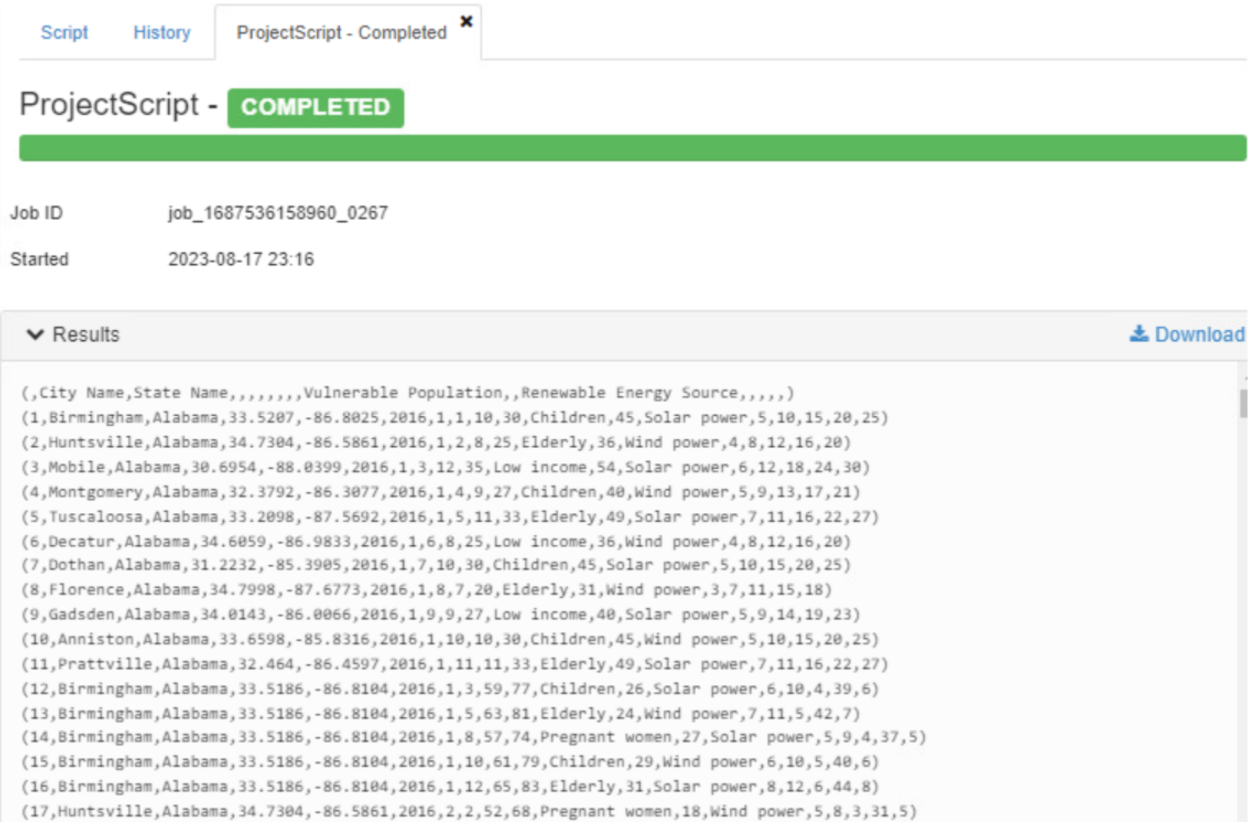
Janki Patel(N01533282)
Vrushali Ponkia(N01530336)

Using Pig

1) **Load csv file and store in pig using pigstorage function**

```
air_pollution = LOAD '/user/maria_dev/air_pollution.csv'
USING PigStorage(',')
AS (
  Id:int,
  CityName:chararray,
  StateName:chararray,
  Latitude:float,
  Longitude:float,
  Year:int,
  Month:int,
  Day:int,
  PollutionLevel:int,
  AQILevel:int,
  Vulnerable_Pollution:chararray,
  HospitalAdmissions:int,
  RenewableEnergySource:chararray,
  ParticulateMatter:int,
  Nitrogen:int,
  SulfurDioxide:int,
  ozone:int,
  CarbonMonoxide:int
);

--dump air_pollution;
```



Job ID job_1687536158960_0267

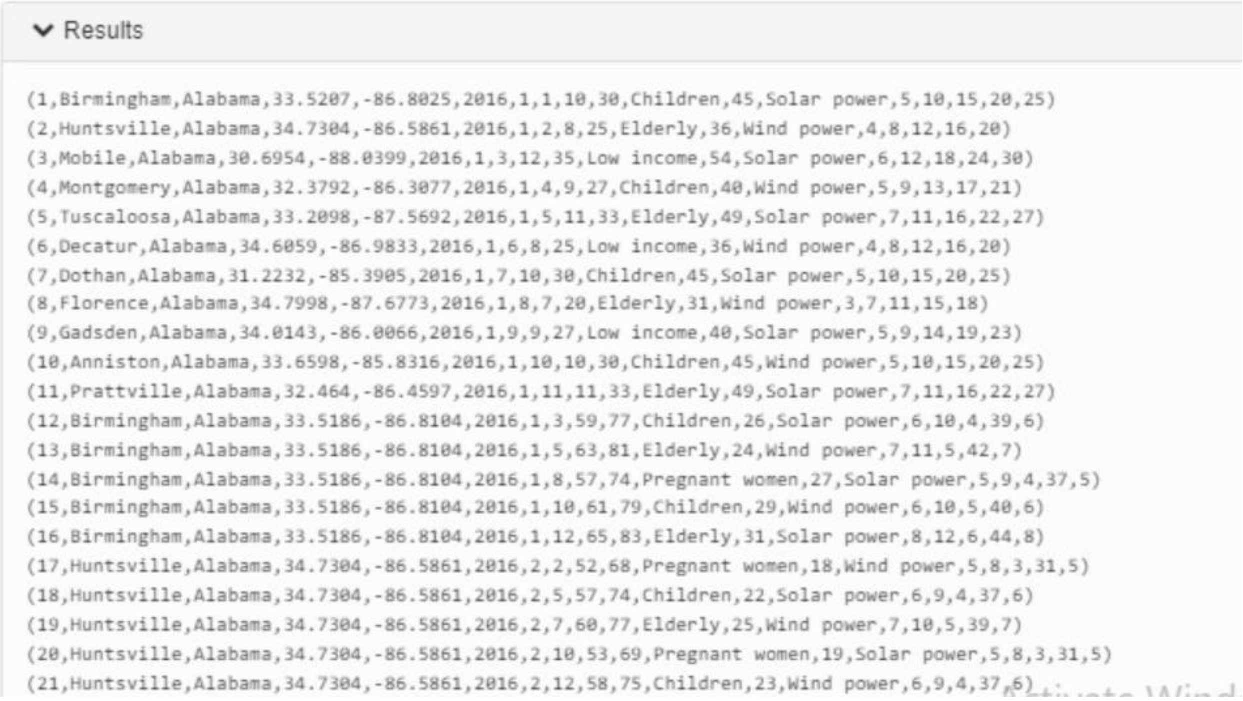
Started 2023-08-17 23:16

Results

```
(,City Name,State Name,,,,,,,,Vulnerable Population,,Renewable Energy Source,,,,,)
(1,Birmingham,Alabama,33.5207,-86.8025,2016,1,1,10,30,Children,45,Solar power,5,10,15,20,25)
(2,Huntsville,Alabama,34.7304,-86.5861,2016,1,2,8,25,Elderly,36,Wind power,4,8,12,16,20)
(3,Mobile,Alabama,30.6954,-88.0399,2016,1,3,12,35,Low income,54,Solar power,6,12,18,24,30)
(4,Montgomery,Alabama,32.3792,-86.3077,2016,1,4,9,27,Children,40,Wind power,5,9,13,17,21)
(5,Tuscaloosa,Alabama,33.2098,-87.5692,2016,1,5,11,33,Elderly,49,Solar power,7,11,16,22,27)
(6,Decatur,Alabama,34.6059,-86.9833,2016,1,6,8,25,Low income,36,Wind power,4,8,12,16,20)
(7,Dothan,Alabama,31.2232,-85.3905,2016,1,7,10,30,Children,45,Solar power,5,10,15,20,25)
(8,Florence,Alabama,34.7998,-87.6773,2016,1,8,7,20,Elderly,31,Wind power,3,7,11,15,18)
(9,Gadsden,Alabama,34.0143,-86.0066,2016,1,9,9,27,Low income,40,Solar power,5,9,14,19,23)
(10,Anniston,Alabama,33.6598,-85.8316,2016,1,10,10,30,Children,45,Wind power,5,10,15,20,25)
(11,Prattville,Alabama,32.464,-86.4597,2016,1,11,11,33,Elderly,49,Solar power,7,11,16,22,27)
(12,Birmingham,Alabama,33.5186,-86.8104,2016,1,3,59,77,Children,26,Solar power,6,10,4,39,6)
(13,Birmingham,Alabama,33.5186,-86.8104,2016,1,5,63,81,Elderly,24,Wind power,7,11,5,42,7)
(14,Birmingham,Alabama,33.5186,-86.8104,2016,1,8,57,74,Pregnant women,27,Solar power,5,9,4,37,5)
(15,Birmingham,Alabama,33.5186,-86.8104,2016,1,10,61,79,Children,29,Wind power,6,10,5,40,6)
(16,Birmingham,Alabama,33.5186,-86.8104,2016,1,12,65,83,Elderly,31,Solar power,8,12,6,44,8)
(17,Huntsville,Alabama,34.7304,-86.5861,2016,2,2,52,68,Pregnant women,18,Wind power,5,8,3,31,5)
```

2) Assuming you have already loaded and filtered the data as shown in your code

```
filter_data = FILTER air_pollution BY Id IS NOT NULL AND CityName IS NOT NULL;
--dump filter_data
```



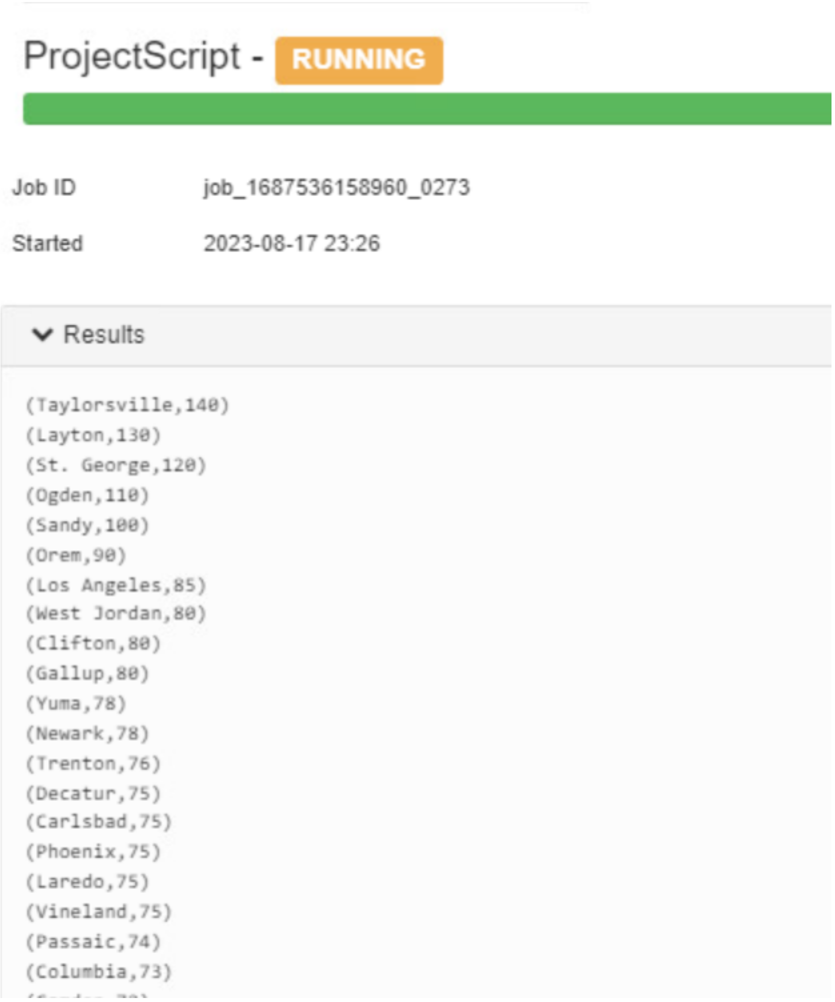
```
(1,Birmingham,Alabama,33.5207,-86.8025,2016,1,1,10,30,Children,45,Solar power,5,10,15,20,25)
(2,Huntsville,Alabama,34.7304,-86.5861,2016,1,2,8,25,Elderly,36,Wind power,4,8,12,16,20)
(3,Mobile,Alabama,30.6954,-88.0399,2016,1,3,12,35,Low income,54,Solar power,6,12,18,24,30)
(4,Montgomery,Alabama,32.3792,-86.3077,2016,1,4,9,27,Children,40,Wind power,5,9,13,17,21)
(5,Tuscaloosa,Alabama,33.2098,-87.5692,2016,1,5,11,33,Elderly,49,Solar power,7,11,16,22,27)
(6,Decatur,Alabama,34.6059,-86.9833,2016,1,6,8,25,Low income,36,Wind power,4,8,12,16,20)
(7,Dothan,Alabama,31.2232,-85.3905,2016,1,7,10,30,Children,45,Solar power,5,10,15,20,25)
(8,Florence,Alabama,34.7998,-87.6773,2016,1,8,7,20,Elderly,31,Wind power,3,7,11,15,18)
(9,Gadsden,Alabama,34.0143,-86.0066,2016,1,9,9,27,Low income,40,Solar power,5,9,14,19,23)
(10,Anniston,Alabama,33.6598,-85.8316,2016,1,10,10,30,Children,45,Wind power,5,10,15,20,25)
(11,Prattville,Alabama,32.464,-86.4597,2016,1,11,11,33,Elderly,49,Solar power,7,11,16,22,27)
(12,Birmingham,Alabama,33.5186,-86.8104,2016,1,3,59,77,Children,26,Solar power,6,10,4,39,6)
(13,Birmingham,Alabama,33.5186,-86.8104,2016,1,5,63,81,Elderly,24,Wind power,7,11,5,42,7)
(14,Birmingham,Alabama,33.5186,-86.8104,2016,1,8,57,74,Pregnant women,27,Solar power,5,9,4,37,5)
(15,Birmingham,Alabama,33.5186,-86.8104,2016,1,10,61,79,Children,29,Wind power,6,10,5,40,6)
(16,Birmingham,Alabama,33.5186,-86.8104,2016,1,12,65,83,Elderly,31,Solar power,8,12,6,44,8)
(17,Huntsville,Alabama,34.7304,-86.5861,2016,2,2,52,68,Pregnant women,18,Wind power,5,8,3,31,5)
(18,Huntsville,Alabama,34.7304,-86.5861,2016,2,5,57,74,Children,22,Solar power,6,9,4,37,6)
(19,Huntsville,Alabama,34.7304,-86.5861,2016,2,7,60,77,Elderly,25,Wind power,7,10,5,39,7)
(20,Huntsville,Alabama,34.7304,-86.5861,2016,2,10,53,69,Pregnant women,19,Solar power,5,8,3,31,5)
(21,Huntsville,Alabama,34.7304,-86.5861,2016,2,12,58,75,Children,23,Wind power,6,9,4,37,6)
```

3) Find city with the highest pollution level

```
max_pollution = FOREACH (GROUP filter_data BY CityName) {
  sorted = ORDER filter_data BY PollutionLevel DESC;
  top_city = LIMIT sorted 1;
  GENERATE FLATTEN(top_city.(CityName, PollutionLevel));
}

-- Display the result in descending order of pollution level
max_pollution_ordered = ORDER max_pollution BY PollutionLevel DESC;

-- Display the result
--DUMP max_pollution_ordered;
```



ProjectScript -

Job ID job_1687536158960_0273

Started 2023-08-17 23:26

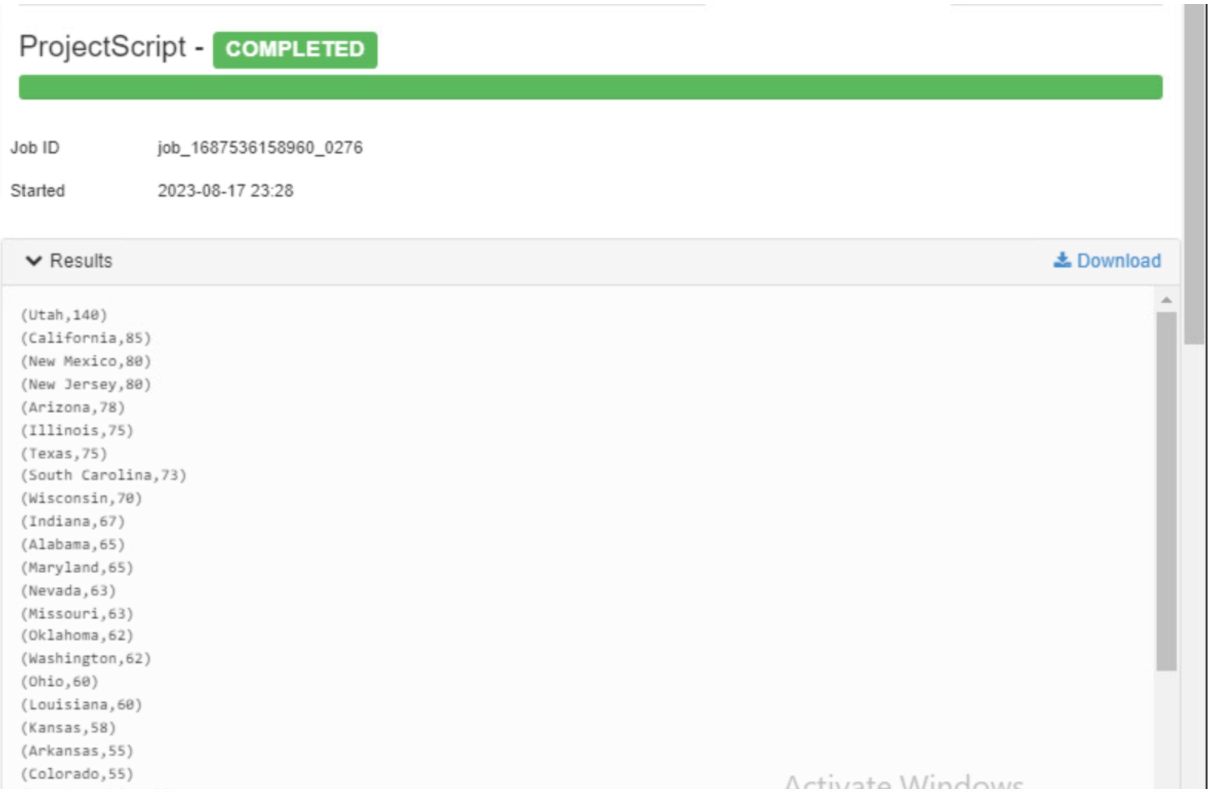
```
(Taylorsville,140)
(Layton,130)
(St. George,120)
(Ogden,110)
(Sandy,100)
(Orem,90)
(Los Angeles,85)
(West Jordan,80)
(Clifton,80)
(Gallup,80)
(Yuma,78)
(Newark,78)
(Trenton,76)
(Decatur,75)
(Carlsbad,75)
(Phoenix,75)
(Laredo,75)
(Vineland,75)
(Passaic,74)
(Columbia,73)
```

4)Find statename with the highest pollution level

```
max_pollution_state = FOREACH (GROUP filter_data BY StateName) {
  sorted = ORDER filter_data BY PollutionLevel DESC;
  top_state = LIMIT sorted 1;
  GENERATE FLATTEN(top_state.(StateName, PollutionLevel));
}

-- Display the result in descending order of pollution level
max_pollution_ordered_state = ORDER max_pollution_state BY PollutionLevel DESC;

-- Display the result
--DUMP max_pollution_ordered_state;
```



Job ID job_1687536158960_0276

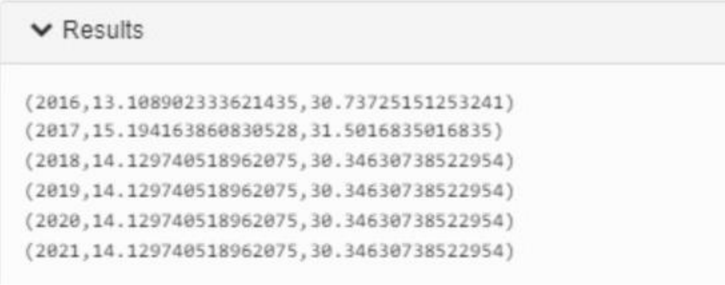
Started 2023-08-17 23:28

```
(Utah,140)
(California,85)
(New Mexico,80)
(New Jersey,80)
(Arizona,78)
(Illinois,75)
(Texas,75)
(South Carolina,73)
(Wisconsin,70)
(Indiana,67)
(Alabama,65)
(Maryland,65)
(Nevada,63)
(Missouri,63)
(Oklahoma,62)
(Washington,62)
(Ohio,60)
(Louisiana,60)
(Kansas,58)
(Arkansas,55)
(Colorado,55)
```

5)Show the average pollution level and hospital admission according to the year

```
pollution_data = FILTER filter_data BY Year > 0 AND HospitalAdmissions >= 0;
pollution_data = FOREACH pollution_data GENERATE Year, HospitalAdmissions,
PollutionLevel;

-- Group data by Year and calculate average HospitalAdmissions and PollutionLevel
grouped_data = GROUP pollution_data BY Year;
result = FOREACH grouped_data {
  avg_hospital_admissions = AVG(pollution_data.HospitalAdmissions);
  avg_pollution_level = AVG(pollution_data.PollutionLevel);
  GENERATE group AS Year, avg_hospital_admissions, avg_pollution_level;
}
-- Store the result in a new relation (or alias) for visualization
final_result = ORDER result BY Year;
--dump final_result;
```



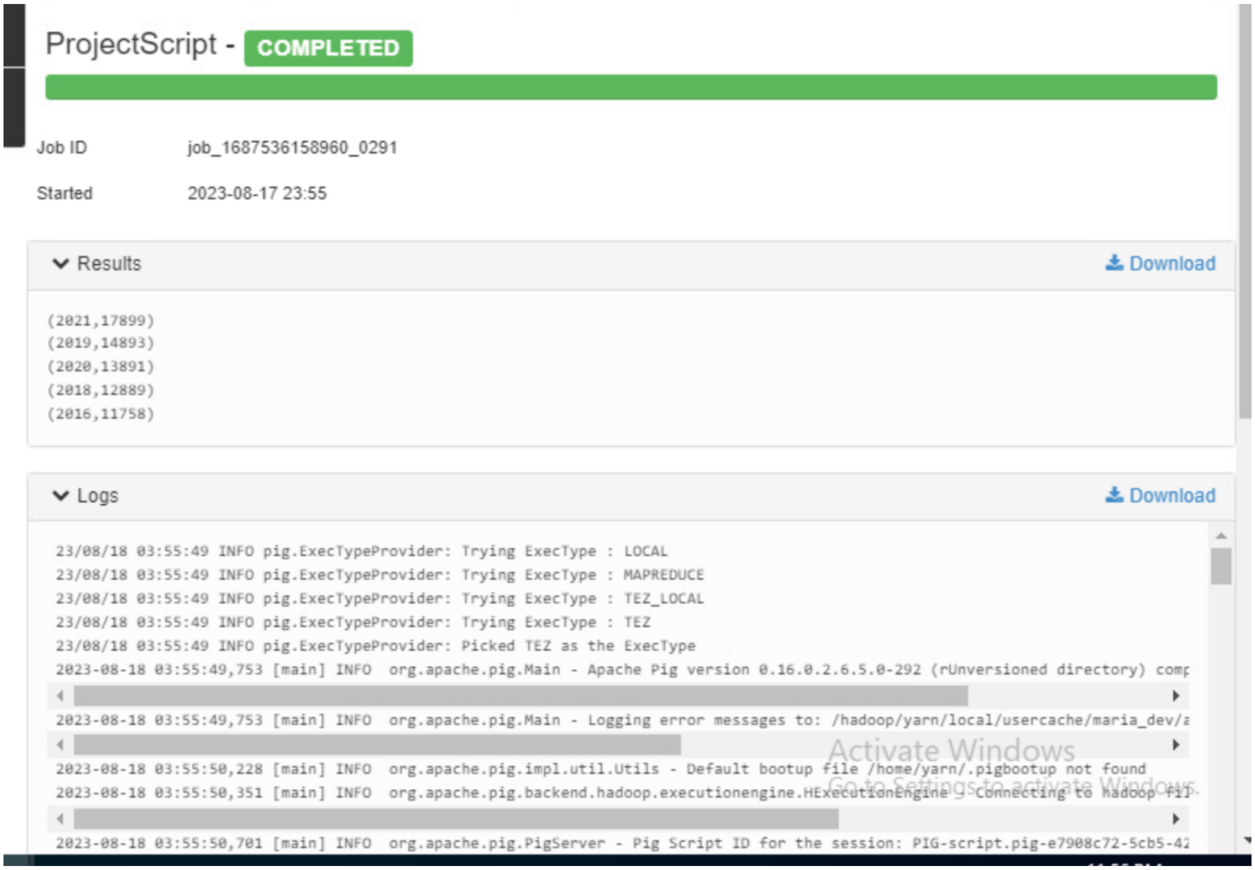
```
(2016,13.108902333621435,30.73725151253241)
(2017,15.194163860830528,31.5016835016835)
(2018,14.129740518962075,30.34630738522954)
(2019,14.129740518962075,30.34630738522954)
(2020,14.129740518962075,30.34630738522954)
(2021,14.129740518962075,30.34630738522954)
```

6) highest carbon Monoxide in year

```
-- Group data by year and calculate the sum of CarbonMonoxide for each year
grouped_data = GROUP filter_data BY Year;
sum_carbon_monoxide = FOREACH grouped_data GENERATE group AS year,
SUM(filter_data.CarbonMonoxide) AS totalCarbonMonoxide;

-- Find the year with the highest total carbon monoxide
max_carbon_monoxide = ORDER sum_carbon_monoxide BY totalCarbonMonoxide DESC;
highest_carbon_monoxide_year = LIMIT max_carbon_monoxide 5;

-- Print the result
--DUMP highest_carbon_monoxide_year;
```



Job ID job_1687536158960_0291

Started 2023-08-17 23:55

```
(2021,17899)
(2019,14893)
(2020,13891)
(2018,12889)
(2016,11758)
```

```
23/08/18 03:55:49 INFO pig.ExecTypeProvider: Trying ExecType : LOCAL
23/08/18 03:55:49 INFO pig.ExecTypeProvider: Trying ExecType : MAPREDUCE
23/08/18 03:55:49 INFO pig.ExecTypeProvider: Trying ExecType : TEZ_LOCAL
23/08/18 03:55:49 INFO pig.ExecTypeProvider: Trying ExecType : TEZ
23/08/18 03:55:49 INFO pig.ExecTypeProvider: Picked TEZ as the ExecType
2023-08-18 03:55:49,753 [main] INFO  org.apache.pig.Main - Apache Pig version 0.16.0.2.6.5.0-292 (rUnversioned directory) comp
2023-08-18 03:55:49,753 [main] INFO  org.apache.pig.Main - Logging error messages to: /hadoop/yarn/local/usercache/maria_dev/a
2023-08-18 03:55:50,228 [main] INFO  org.apache.pig.impl.util.Utils - Default bootup file /home/yarn/.pigbootup not found
2023-08-18 03:55:50,351 [main] INFO  org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - Connecting to hadoop fil
2023-08-18 03:55:50,701 [main] INFO  org.apache.pig.PigServer - Pig Script ID for the session: PIG-script.pig-e7908c72-5cb5-42
```

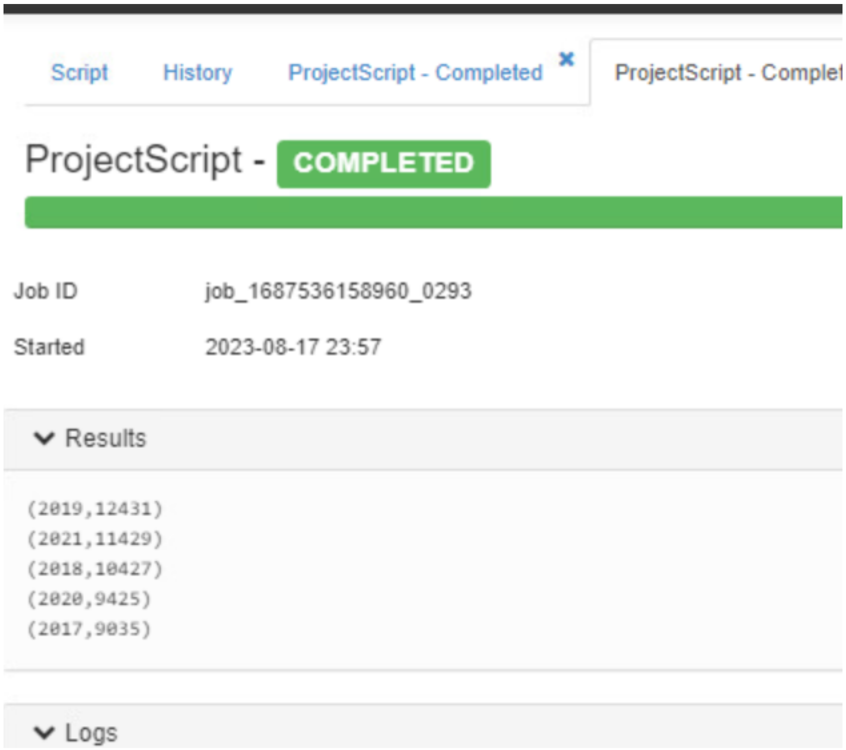
-7) highest sulfur dioxide in a year

-- Assuming you have already loaded and filtered the data as shown in your code
filtered_data = FILTER air_pollution BY Id IS NOT NULL AND CityName IS NOT NULL;

-- Group the filtered data by year and calculate the sum of SulfurDioxide for each year
grouped_by_year = GROUP filtered_data BY Year;
sum_sulfur_dioxide = FOREACH grouped_by_year GENERATE group AS year, SUM(filtered_data.SulfurDioxide) AS total_sulfur_dioxide;

-- Find the year with the highest total SulfurDioxide
max_sulfur_dioxide = ORDER sum_sulfur_dioxide BY total_sulfur_dioxide DESC;
max_year = LIMIT max_sulfur_dioxide 5;

-- Display the year with the highest SulfurDioxide level
--DUMP max_year;



ProjectScript - COMPLETED

Job ID job_1687536158960_0293

Started 2023-08-17 23:57

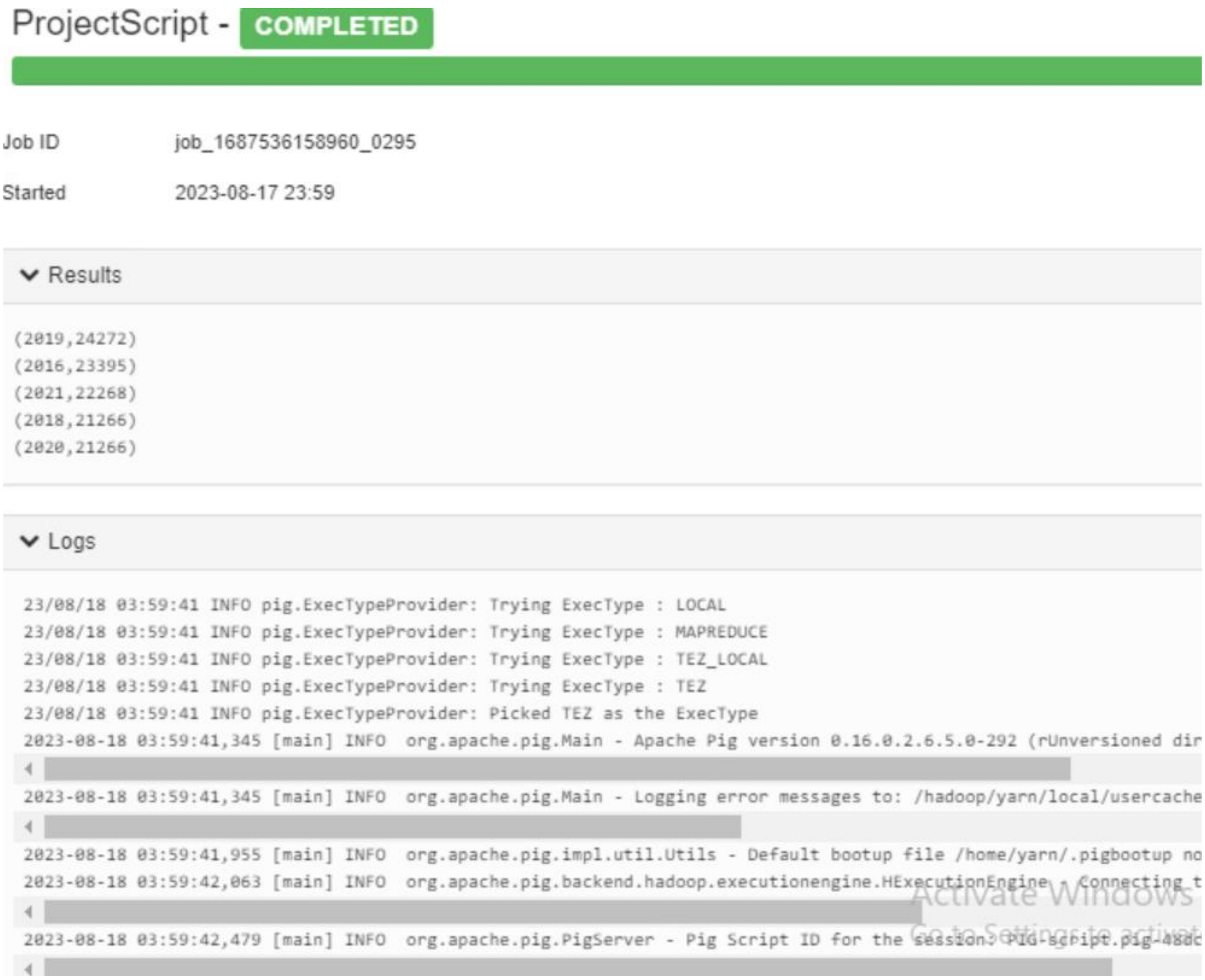
Results

```
(2019,12431)
(2021,11429)
(2018,10427)
(2020,9425)
(2017,9035)
```

8) **highest Nitrogen in a year**

```
-- Group the data by year and calculate the sum of Nitrogen levels for each year
grouped_data = GROUP filter_data BY Year;
sum_nitrogen_by_year = FOREACH grouped_data GENERATE group AS Year,
SUM(filter_data.Nitrogen) AS TotalNitrogen;

-- Find the year with the highest total Nitrogen levels
max_nitrogen_year = ORDER sum_nitrogen_by_year BY TotalNitrogen DESC;
top_year = LIMIT max_nitrogen_year 5;
```



ProjectScript - COMPLETED

Job ID job_1687536158960_0295

Started 2023-08-17 23:59

Results

```
(2019,24272)
(2016,23395)
(2021,22268)
(2018,21266)
(2020,21266)
```

Logs

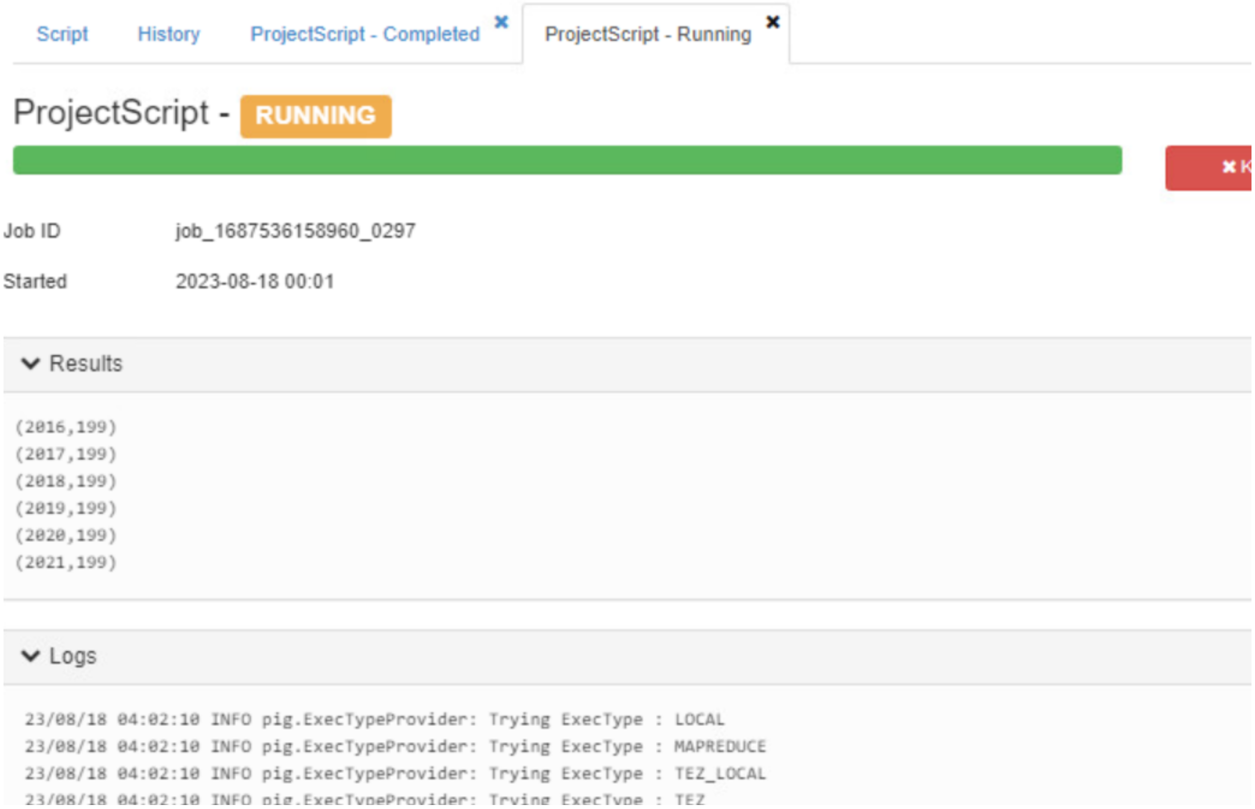
```
23/08/18 03:59:41 INFO pig.ExecTypeProvider: Trying ExecType : LOCAL
23/08/18 03:59:41 INFO pig.ExecTypeProvider: Trying ExecType : MAPREDUCE
23/08/18 03:59:41 INFO pig.ExecTypeProvider: Trying ExecType : TEZ_LOCAL
23/08/18 03:59:41 INFO pig.ExecTypeProvider: Trying ExecType : TEZ
23/08/18 03:59:41 INFO pig.ExecTypeProvider: Picked TEZ as the ExecType
2023-08-18 03:59:41,345 [main] INFO  org.apache.pig.Main - Apache Pig version 0.16.0.2.6.5.0-292 (rUnversioned dir
2023-08-18 03:59:41,345 [main] INFO  org.apache.pig.Main - Logging error messages to: /hadoop/yarn/local/usercache
2023-08-18 03:59:41,955 [main] INFO  org.apache.pig.impl.util.Utils - Default bootup file /home/yarn/.pigbootup no
2023-08-18 03:59:42,063 [main] INFO  org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - Connecting t
2023-08-18 03:59:42,479 [main] INFO  org.apache.pig.PigServer - Pig Script ID for the session: PIG-script.pig-48dc
```

9) Highest AQI level in year

```
-- Group the data by year and find the maximum AQI level for each year
grouped_by_year = GROUP filter_data BY Year;
max_aqi_per_year = FOREACH grouped_by_year GENERATE group AS year,
MAX(filter_data.AQILevel) AS max_aqi_level;

-- Order the results by AQI level in descending order
sorted_results = ORDER max_aqi_per_year BY max_aqi_level DESC;

-- Display the final results
--DUMP sorted_results;
```



Job ID job_1687536158960_0297

Started 2023-08-18 00:01

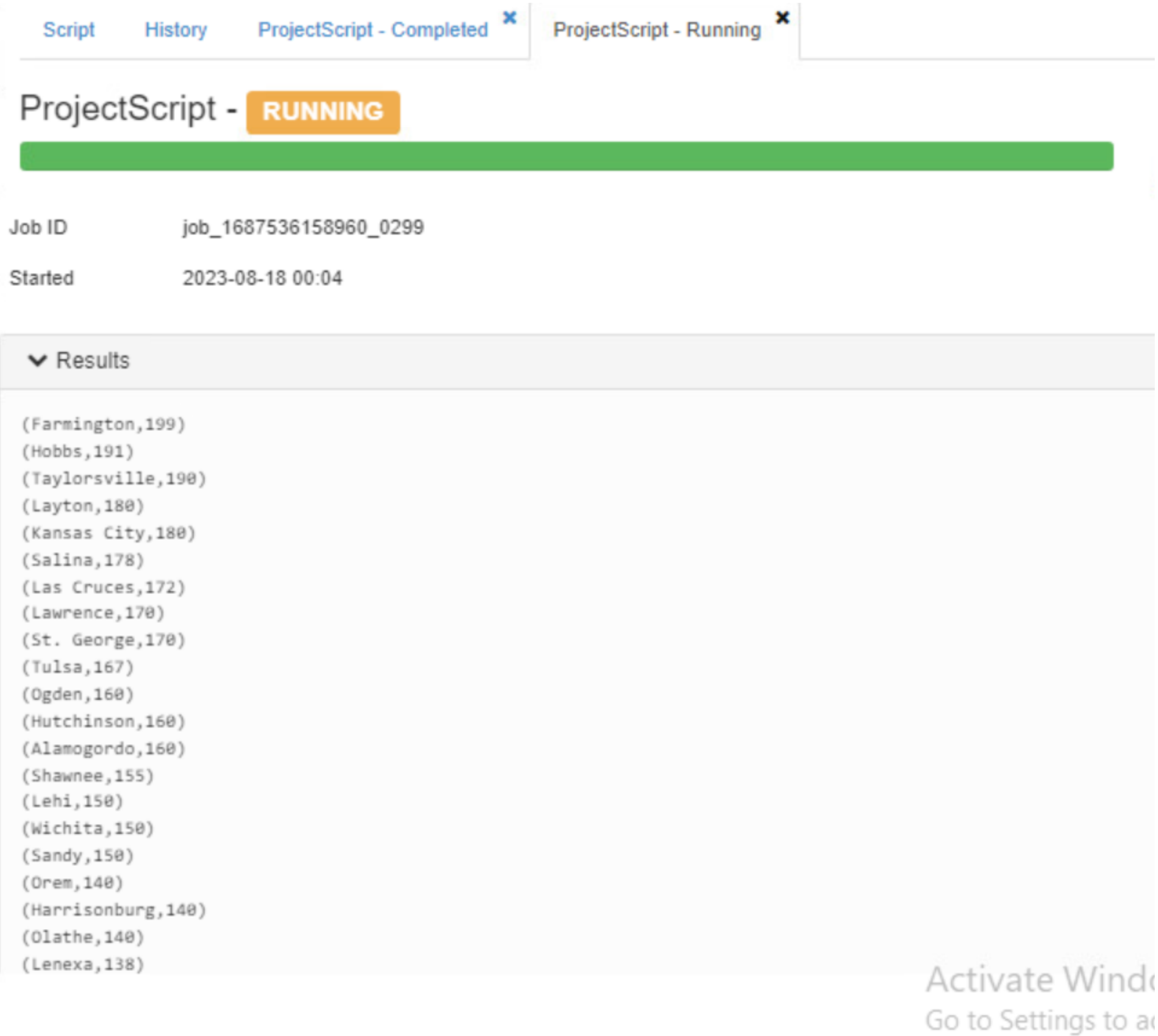
```
(2016,199)
(2017,199)
(2018,199)
(2019,199)
(2020,199)
(2021,199)
```

```
23/08/18 04:02:10 INFO pig.ExecTypeProvider: Trying ExecType : LOCAL
23/08/18 04:02:10 INFO pig.ExecTypeProvider: Trying ExecType : MAPREDUCE
23/08/18 04:02:10 INFO pig.ExecTypeProvider: Trying ExecType : TEZ_LOCAL
23/08/18 04:02:10 INFO pig.ExecTypeProvider: Trying ExecType : TEZ
```

10)city has highest AQI LEVEL

```
max_aqi_city = FOREACH (GROUP air_pollution BY CityName) {
  ordered = ORDER air_pollution BY PollutionLevel DESC;
  top_city = LIMIT ordered 1;
  GENERATE FLATTEN(top_city.CityName) AS CityName, FLATTEN(top_city.AQILevel) AS AQILevel;
}


-- Order the results by AQI level in descending order
max_aqi_city_or = ORDER max_aqi_city BY AQILevel DESC;
--DUMP max_aqi_city_or;
```



Job ID job_1687536158960_0299

Started 2023-08-18 00:04

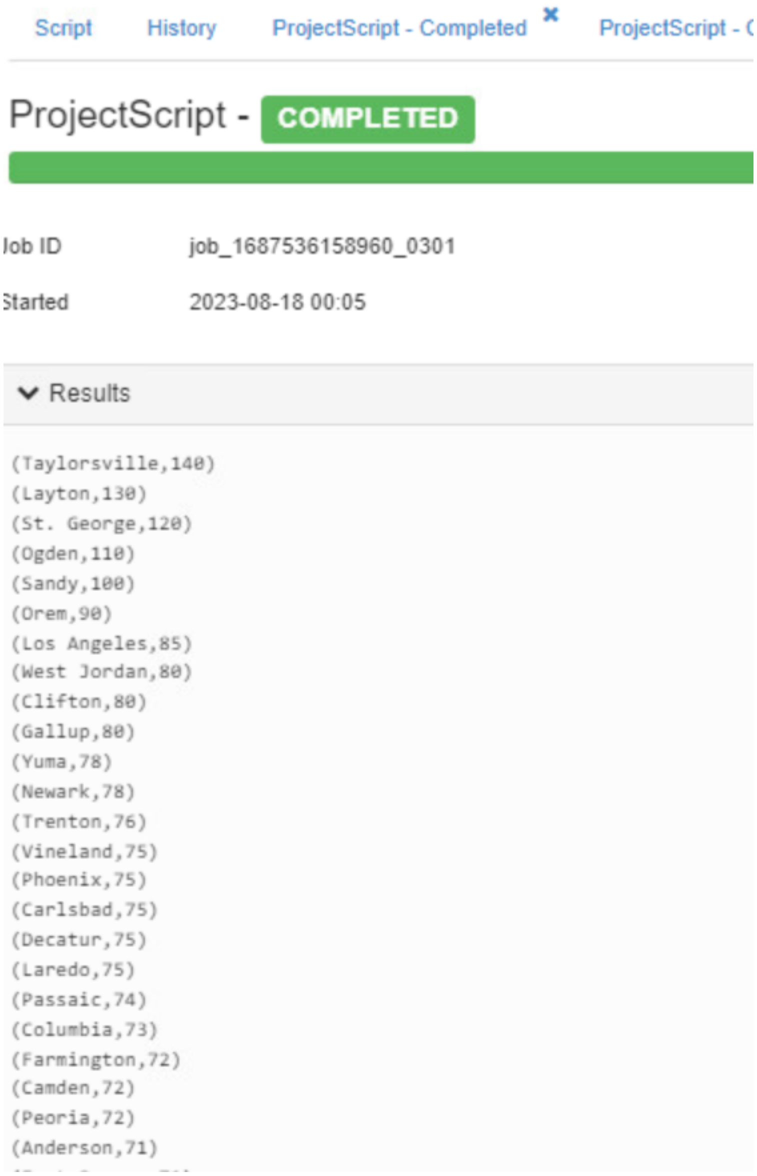
```
(Farmington,199)
(Hobbs,191)
(Taylorsville,190)
(Layton,180)
(Kansas City,180)
(Salina,178)
(Las Cruces,172)
(Lawrence,170)
(St. George,170)
(Tulsa,167)
(Ogden,160)
(Hutchinson,160)
(Alamogordo,160)
(Shawnee,155)
(Lehi,150)
(Wichita,150)
(Sandy,150)
(Orem,140)
(Harrisonburg,140)
(Olathe,140)
(Lenexa,138)
```

11)city has highest pollution level

```
-- Calculate the city with the highest pollution level
max_pollution = FOREACH (GROUP air_pollution BY CityName) {
  ordered = ORDER air_pollution BY PollutionLevel DESC;
  top_city = LIMIT ordered 1;
  GENERATE FLATTEN(top_city.CityName), FLATTEN(top_city.PollutionLevel);
}
max_pollution_de = ORDER max_pollution BY PollutionLevel DESC;

-- Display the result
--DUMP max_pollution_de;
```

Air Pollution in the USA From 2016 to 2021 **Janki Patel(N01533282)**
Vrushali Ponkia(N01530336)



ProjectScript - COMPLETED

Job ID job_1687536158960_0301

Started 2023-08-18 00:05

Results

```
(Taylorsville,140)
(Layton,130)
(St. George,120)
(Ogden,110)
(Sandy,100)
(Orem,90)
(Los Angeles,85)
(West Jordan,80)
(Clifton,80)
(Gallup,80)
(Yuma,78)
(Newark,78)
(Trenton,76)
(Vineland,75)
(Phoenix,75)
(Carlsbad,75)
(Decatur,75)
(Laredo,75)
(Passaic,74)
(Columbia,73)
(Farmington,72)
(Camden,72)
(Peoria,72)
(Anderson,71)
```