--- MySQL Database connection using Python: -

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python 3 (ipykernel)

```
[1]: !pip install mysqlclient
```

```
Requirement already satisfied: mysqlclient in c:\users\chandan\anaconda3\lib\site-packages (2.2.4)
```

```
[14]: !pip install pandas
```

```
Requirement already satisfied: pandas in c:\users\chandan\anaconda3\lib\site-packages (2.2.2)
Requirement already satisfied: numpy>=1.26.0 in c:\users\chandan\anaconda3\lib\site-packages (from pandas) (1.26.4)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\chandan\anaconda3\lib\site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\chandan\anaconda3\lib\site-packages (from pandas) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\chandan\anaconda3\lib\site-packages (from pandas) (2023.3)
Requirement already satisfied: six>=1.5 in c:\users\chandan\anaconda3\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

```
[2]: from sqlalchemy import create_engine
     import pandas as pd
```

```
[3]: from urllib.parse import quote
     pd = quote("                ")
```

```
[4]: pd
```

```
[4]:
```

```
[5]: #conn_string='mysql://<username>:<password>@<host_name>/<database_name>'

     con_string=f'mysql://root:{pd}@localhost/museumdata'
```

```
[20]: db = create_engine(con_string)
      conn = db.connect()
```

File Edit View Run Kernel Settings Help Trusted

Code JupyterLab Python 3 (ipykernel)

```
conn = db.connect()
```

```
[7]: conn
```

```
[7]: <bound method Engine.connect of Engine(mysql://root:***@localhost/museumdata)>
```

```
[8]: f = ['artist','canvas_size','subject','work','museum','museum_hours','product_size']
```

```
[22]: for i in f:
          d=pd.read_csv(f"C:/Users/Chandan/Desktop/ETLHIVE/SQL Course/Datasets/MuseumDataset/{i}.csv")
          d.to_sql(i,con=conn,if_exists='replace',index=False)
```

```
[ ]:
```

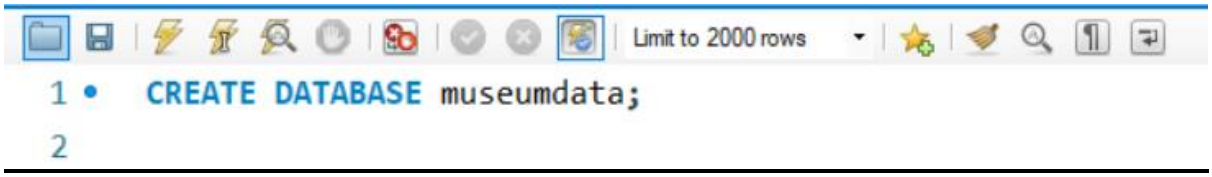
```
[16]: import pandas as pd
      d=pd.read_csv(r"C:\Users\Chandan\Desktop\ETLHIVE\SQL Course\Datasets\MuseumDataset\artist.csv")
      d
```

[16]:

	artist_id	full_name	first_name	middle_names	last_name	nationality	style	birth	death
0	500	Pierre-Auguste Renoir	Pierre	Auguste	Renoir	French	Impressionist	1841	1919
1	501	Alexandre Cabanel	Alexandre	NaN	Cabanel	French	Classicist	1823	1889
2	502	James Ensor	James	NaN	Ensor	Belgian	Expressionist	1860	1949
3	503	Maximilien Luce	Maximilien	NaN	Luce	French	Pointillist	1858	1941
4	504	August Macke	August	NaN	Macke	German	Expressionist	1887	1914
...	...	...	...	...	...	...	...	...	...
416	916	Jean-Baptiste-Camille Corot	Jean	Baptiste Camille	Corot	French	Romantic	1796	1875
417	917	Jan Toorop	Jan	NaN	Toorop	Dutch	Symbolist	1858	1928
418	918	Rembrandt Peale	Rembrandt	NaN	Peale	American	Neoclassical	1778	1860
419	919	Edmund Charles Tarbell	Edmund	Charles	Tarbell	American	Impressionist	1862	1938
420	920	Rembrandt Van Rijn	Rembrandt	NaN	Van Rijn	Dutch	Baroque	1606	1669

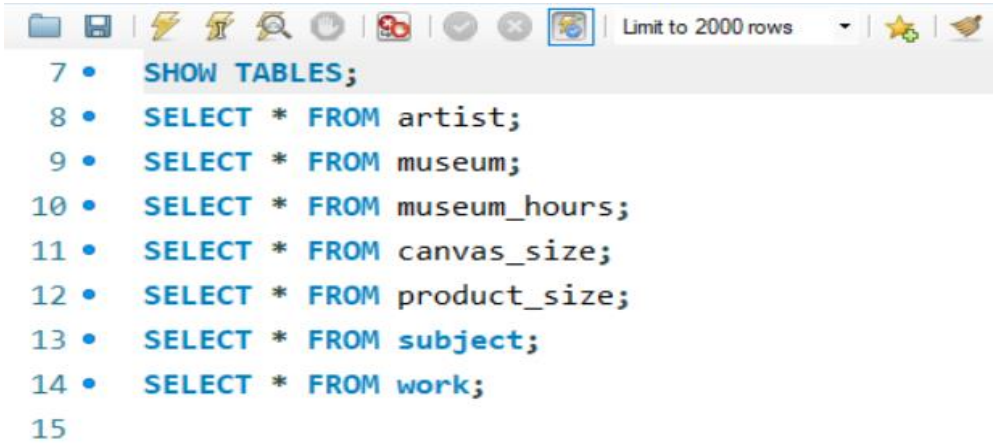
421 rows × 9 columns

--- **Creation of database in MySQL Workbench: -**



--- **Using the Created database: -**

--- **Showing Tables in the database and data in each and every table**:-

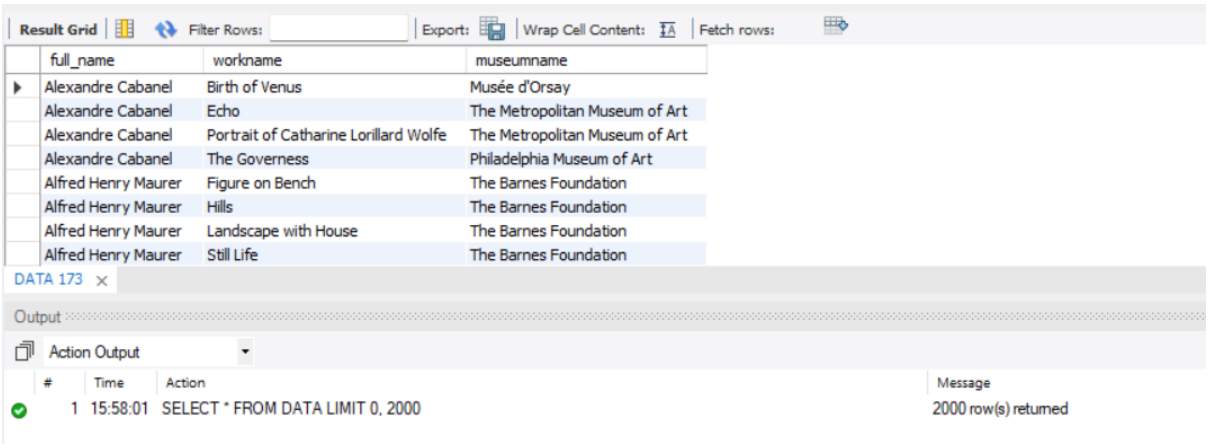


-- **SQL PROJECT**

-- 1. **Retrieve the full name of artists along with the names of the museums where their works are displayed** :-

```
18 •  CREATE VIEW DATA AS
19    SELECT a.full_name, w.name as workname , m.name as museumname
20    FROM work as w
21    INNER JOIN artist as a
22    ON a.artist_id = w.artist_id
23    INNER JOIN museum as m
24    on w.museum_id=m.museum_id;
25
26 •  SELECT * FROM DATA;
```

OUTPUT:-



full_name	workname	museumname
Alexandre Cabanel	Birth of Venus	Musée d'Orsay
Alexandre Cabanel	Echo	The Metropolitan Museum of Art
Alexandre Cabanel	Portrait of Catharine Lorillard Wolfe	The Metropolitan Museum of Art
Alexandre Cabanel	The Governess	Philadelphia Museum of Art
Alfred Henry Maurer	Figure on Bench	The Barnes Foundation
Alfred Henry Maurer	Hills	The Barnes Foundation
Alfred Henry Maurer	Landscape with House	The Barnes Foundation
Alfred Henry Maurer	Still Life	The Barnes Foundation

DATA 173

Output

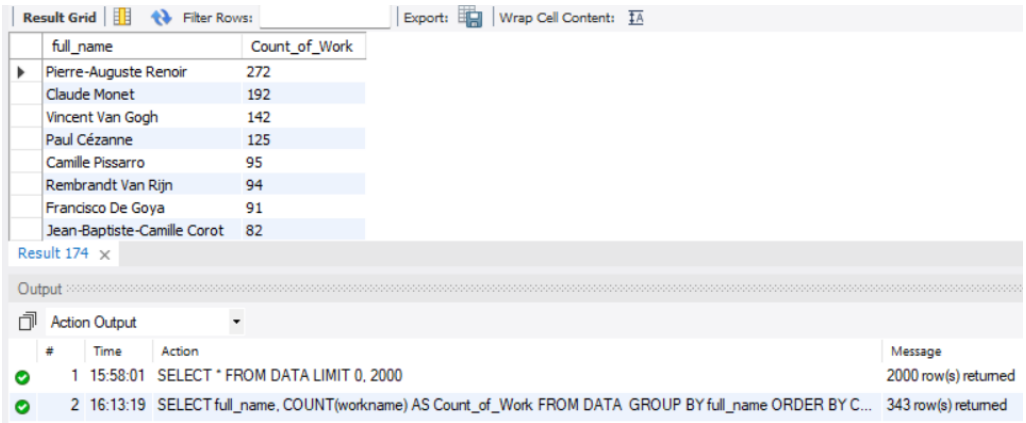
Action Output

#	Time	Action	Message
1	15:58:01	SELECT * FROM DATA LIMIT 0, 2000	2000 row(s) returned

-- 2. How many works does each artist have in the database? Display the artist's full name along with the count of their works, ordered by the count in descending order:-

```
30 •  SELECT full_name, COUNT(workname) AS Count_of_Work
31    FROM DATA
32    GROUP BY full_name
33    ORDER BY Count_of_work DESC;
```

OUTPUT:-



full_name	Count_of_Work
Pierre-Auguste Renoir	272
Claude Monet	192
Vincent Van Gogh	142
Paul Cézanne	125
Camille Pissarro	95
Rembrandt Van Rijn	94
Francisco De Goya	91
Jean-Baptiste-Camille Corot	82

Result 174

Output

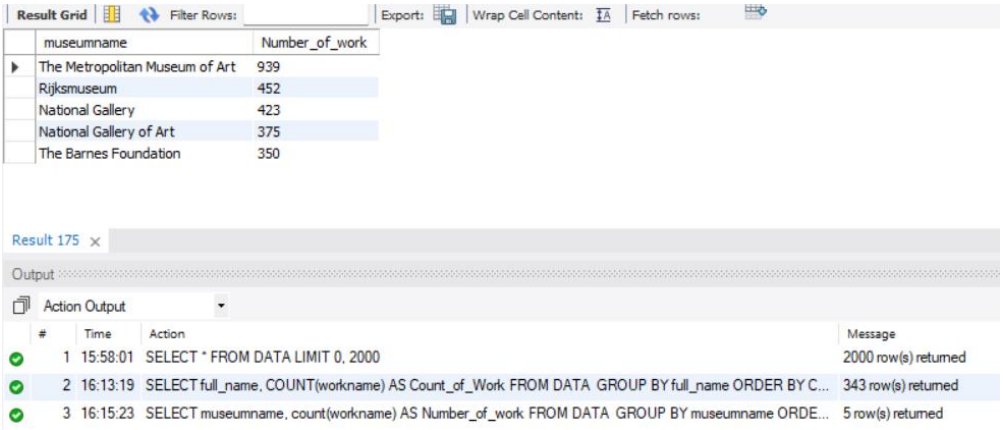
Action Output

#	Time	Action	Message
1	15:58:01	SELECT * FROM DATA LIMIT 0, 2000	2000 row(s) returned
2	16:13:19	SELECT full_name, COUNT(workname) AS Count_of_Work FROM DATA GROUP BY full_name ORDER BY C...	343 row(s) returned

-- 3. List the top 5 museums with the highest number of works displayed in the database, along with their respective counts:-

```
37 •  SELECT museumname, count(workname) AS Number_of_work
38    FROM DATA
39    GROUP BY museumname
40    ORDER BY Number_of_work DESC
41    LIMIT 5;
```

OUTPUT:-



museumname	Number_of_work
The Metropolitan Museum of Art	939
Rijksmuseum	452
National Gallery	423
National Gallery of Art	375
The Barnes Foundation	350

Result 175

Output

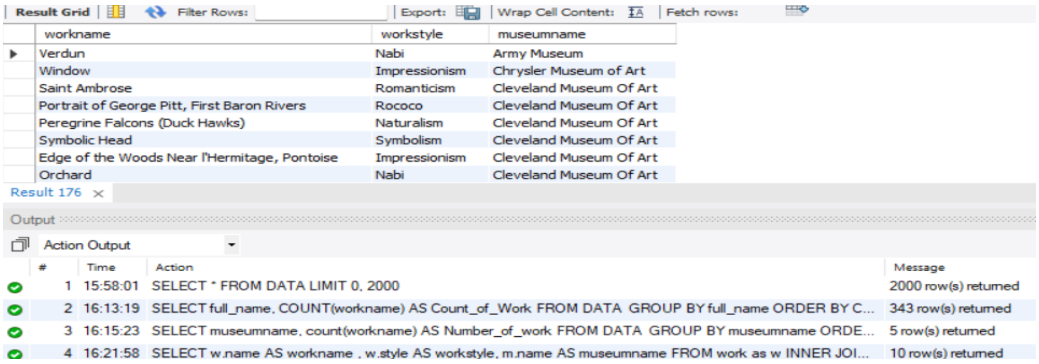
Action Output

#	Time	Action	Message
1	15:58:01	SELECT * FROM DATA LIMIT 0, 2000	2000 row(s) returned
2	16:13:19	SELECT full_name, COUNT(workname) AS Count_of_Work FROM DATA GROUP BY full_name ORDER BY C...	343 row(s) returned
3	16:15:23	SELECT museumname, count(workname) AS Number_of_work FROM DATA GROUP BY museumname ORDE...	5 row(s) returned

-- 4. <u>Display the names and styles of the works along with the corresponding museum names, ordered by the museum name in ascending order. Limit the results to 10</u>:-

```
45 •  SELECT w.name AS workname , w.style AS workstyle, m.name AS museumname
46    FROM work as w
47    INNER JOIN museum as m
48    ON w.museum_id = m.museum_id
49    ORDER BY museumname
50    LIMIT 10;
```

OUTPUT:-



workname	workstyle	museumname
Verdun	Nabi	Army Museum
Window	Impressionism	Chrysler Museum of Art
Saint Ambrose	Romanticism	Cleveland Museum Of Art
Portrait of George Pitt, First Baron Rivers	Rococo	Cleveland Museum Of Art
Peregrine Falcons (Duck Hawks)	Naturalism	Cleveland Museum Of Art
Symbolic Head	Symbolism	Cleveland Museum Of Art
Edge of the Woods Near l'Hermitage, Pontoise	Impressionism	Cleveland Museum Of Art
Orchard	Nabi	Cleveland Museum Of Art

Result 176

Output

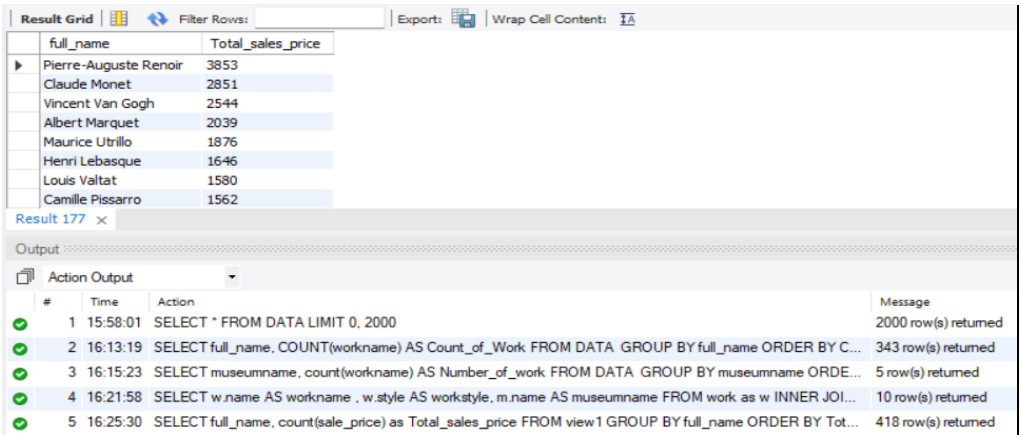
Action Output

#	Time	Action	Message
1	15:58:01	SELECT * FROM DATA LIMIT 0, 2000	2000 row(s) returned
2	16:13:19	SELECT full_name, COUNT(workname) AS Count_of_Work FROM DATA GROUP BY full_name ORDER BY C...	343 row(s) returned
3	16:15:23	SELECT museumname, count(workname) AS Number_of_work FROM DATA GROUP BY museumname ORDE...	5 row(s) returned
4	16:21:58	SELECT w.name AS workname , w.style AS workstyle, m.name AS museumname FROM work as w INNER JOI...	10 row(s) returned

-- 5. <u>Show the total sale price for each artist's works. Display the artist's full name along with the total sale price, ordered by the total sale price in descending order</u>:-

```
59 •  CREATE VIEW view1 as
60    SELECT a.full_name, p.sale_price
61    FROM artist as a
62    INNER JOIN work as w
63    ON a.artist_id = w.artist_id
64    INNER JOIN product_size as p
65    ON w.work_id = p.work_id;
66
67 •  SELECT * FROM view1;
68
69 •  SELECT full_name, count(sale_price) as Total_sales_price
70    FROM view1
71    GROUP BY full_name
72    ORDER BY Total_sales_price DESC;
```

OUTPUT:-



Result Grid | Filter Rows: | Export: | Wrap Cell Content:

full_name	Total_sales_price
Pierre-Auguste Renoir	3853
Claude Monet	2851
Vincent Van Gogh	2544
Albert Marquet	2039
Maurice Utrillo	1876
Henri Lebasque	1646
Louis Valtat	1580
Camille Pissarro	1562

Result 177

Output

Action Output

#	Time	Action	Message
1	15:58:01	SELECT * FROM DATA LIMIT 0, 2000	2000 row(s) returned
2	16:13:19	SELECT full_name, COUNT(workname) AS Count_of_Work FROM DATA GROUP BY full_name ORDER BY C...	343 row(s) returned
3	16:15:23	SELECT museumname, count(workname) AS Number_of_work FROM DATA GROUP BY museumname ORDE...	5 row(s) returned
4	16:21:58	SELECT w.name AS workname , w.style AS workstyle, m.name AS museumname FROM work as w INNER JOI...	10 row(s) returned
5	16:25:30	SELECT full_name, count(sale_price) as Total_sales_price FROM view1 GROUP BY full_name ORDER BY Tot...	418 row(s) returned

-- 6. <u>List artists who have more than 3 works in the database, along with the count of their works</u>:-

```
77 •  CREATE VIEW view2 as
78    SELECT a.full_name, w.work_id
79    FROM  artist as a
80    INNER JOIN work as w
81    ON a.artist_id = w.artist_id;
82
83 •  SELECT * FROM view2;
84
85 •  SELECT full_name, count(work_id) as  Count_of_work
86    FROM view2
87    GROUP BY full_name
88    HAVING Count_of_work > 3;
```

OUTPUT:-

full_name	Count_of_work
Abraham Mignon	6
Albrecht Adam	8
Édouard Adam	15
Adélaïde Labille-Guiard	12
Adèle Romany	9
Adolph Ulrich Wertmuller	7
Adriaan De Lelie	15
Aelbert Cuyp	28

Result 179

Output

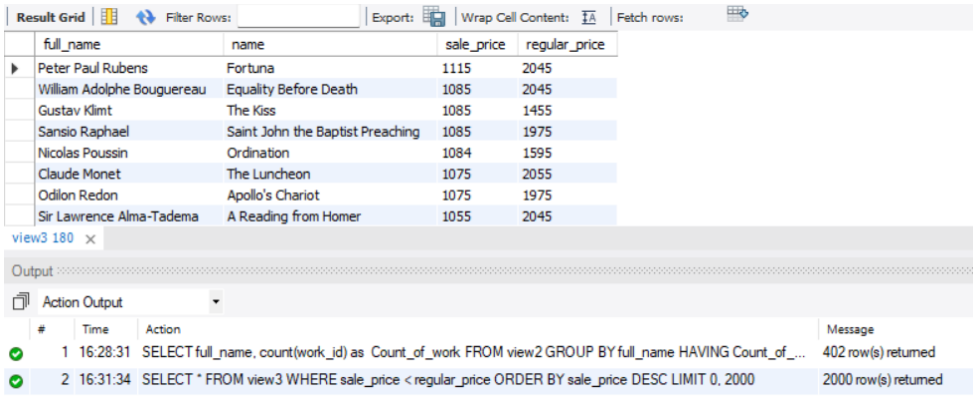
Action Output

#	Time	Action	Message
1	16:28:31	SELECT full_name, count(work_id) as Count_of_work FROM view2 GROUP BY full_name HAVING Count_of_...	402 row(s) returned

-- 7. Retrieve the names of works and their corresponding artists' full names for works that have a sale price smaller than their regular price:-

```
96 •  CREATE VIEW view3 as
97    SELECT a.full_name, w.name, p.sale_price, p.regular_price
98    FROM artist as a
99    INNER JOIN work as w
100   ON a.artist_id = w.artist_id
101   INNER JOIN product_size as p
102   ON w.work_id = p.work_id;
103
104 •  SELECT * FROM view3;
105
106 •  SELECT * FROM view3
107   WHERE sale_price < regular_price
108   ORDER BY sale_price DESC;
```

OUTPUT:-



Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Fetch rows:

full_name	name	sale_price	regular_price
Peter Paul Rubens	Fortuna	1115	2045
William Adolphe Bouguereau	Equality Before Death	1085	2045
Gustav Klimt	The Kiss	1085	1455
Sansio Raphael	Saint John the Baptist Preaching	1085	1975
Nicolas Poussin	Ordination	1084	1595
Claude Monet	The Luncheon	1075	2055
Odilon Redon	Apollo's Chariot	1075	1975
Sir Lawrence Alma-Tadema	A Reading from Homer	1055	2045

view3 180 ×

Output

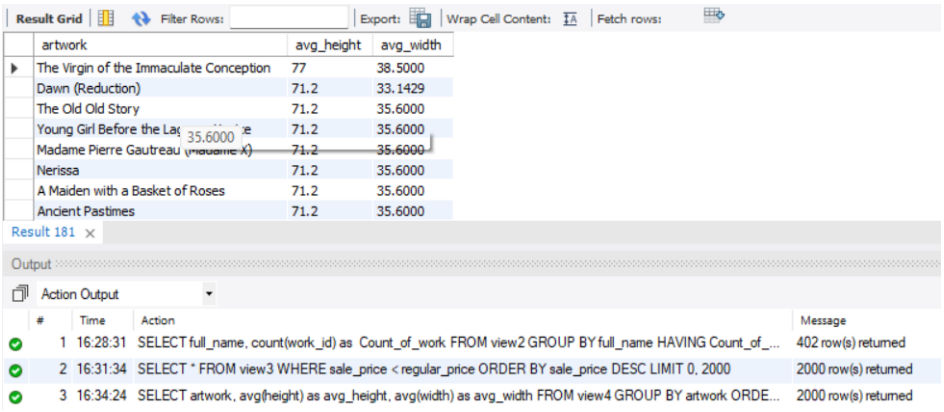
Action Output

#	Time	Action	Message
1	16:28:31	SELECT full_name, count(work_id) as Count_of_work FROM view2 GROUP BY full_name HAVING Count_of_...	402 row(s) returned
2	16:31:34	SELECT * FROM view3 WHERE sale_price < regular_price ORDER BY sale_price DESC LIMIT 0, 2000	2000 row(s) returned

-- 8. Calculate the average height and width of the artworks in the database. Display the average height and width:-

```
115 •  CREATE VIEW view4 as
116    SELECT w.name as artwork, c.height, c.width
117    FROM product_size as p
118    INNER JOIN canvas_size as c
119    ON p.size_id = c.size_id
120    INNER JOIN work as w
121    ON p.work_id = w.work_id;
122
123 •  SELECT * FROM view4;
124
125 •  SELECT artwork, avg(height) as avg_height, avg(width) as avg_width
126    FROM view4
127    GROUP BY artwork
128    ORDER BY avg_height DESC;
```

OUTPUT:-



artwork	avg_height	avg_width
The Virgin of the Immaculate Conception	77	38.5000
Dawn (Reduction)	71.2	33.1429
The Old Old Story	71.2	35.6000
Young Girl Before the La...	71.2	35.6000
Madame Pierre Gautreau (Madame X)	71.2	35.6000
Nerissa	71.2	35.6000
A Maiden with a Basket of Roses	71.2	35.6000
Ancient Pastimes	71.2	35.6000

Result 181

Output

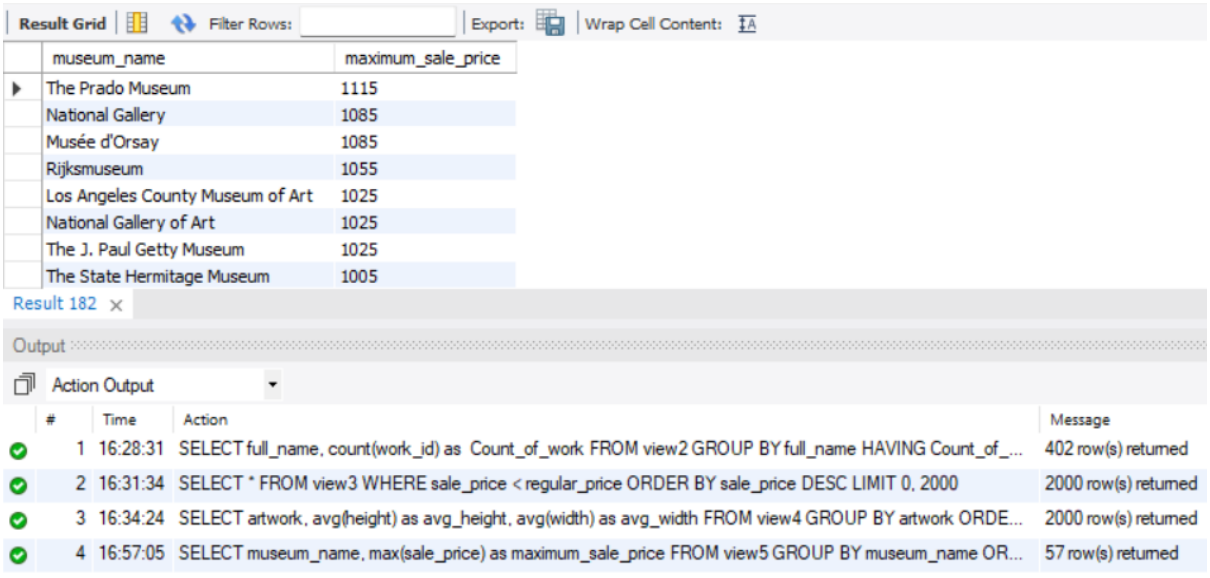
Action Output

#	Time	Action	Message
1	16:28:31	SELECT full_name, count(work_id) as Count_of_work FROM view2 GROUP BY full_name HAVING Count_of_...	402 row(s) returned
2	16:31:34	SELECT * FROM view3 WHERE sale_price < regular_price ORDER BY sale_price DESC LIMIT 0, 2000	2000 row(s) returned
3	16:34:24	SELECT artwork, avg(height) as avg_height, avg(width) as avg_width FROM view4 GROUP BY artwork ORDE...	2000 row(s) returned

-- 9. <u>Find the maximum sale price among all the works in each museum. Display the museum name along with the maximum sale price</u>:-

```
136 •  CREATE VIEW view5 AS
137    SELECT m.name as museum_name, p.sale_price
138    FROM product_size as p
139    INNER JOIN work as w
140    ON p.work_id = w.work_id
141    INNER JOIN museum as m
142    ON w.museum_id = m.museum_id;
143
144 •  SELECT * FROM view5;
145
146 •  SELECT museum_name, max(sale_price) as maximum_sale_price
147    FROM view5
148    GROUP BY museum_name
149    ORDER BY maximum_sale_price DESC;
```

OUTPUT:-



museum_name	maximum_sale_price
The Prado Museum	1115
National Gallery	1085
Musée d'Orsay	1085
Rijksmuseum	1055
Los Angeles County Museum of Art	1025
National Gallery of Art	1025
The J. Paul Getty Museum	1025
The State Hermitage Museum	1005

Result 182

Output

Action Output

#	Time	Action	Message
1	16:28:31	SELECT full_name, count(work_id) as Count_of_work FROM view2 GROUP BY full_name HAVING Count_of_...	402 row(s) returned
2	16:31:34	SELECT * FROM view3 WHERE sale_price < regular_price ORDER BY sale_price DESC LIMIT 0, 2000	2000 row(s) returned
3	16:34:24	SELECT artwork, avg(height) as avg_height, avg(width) as avg_width FROM view4 GROUP BY artwork ORDE...	2000 row(s) returned
4	16:57:05	SELECT museum_name, max(sale_price) as maximum_sale_price FROM view5 GROUP BY museum_name OR...	57 row(s) returned

-- 10. Concatenate the first name and last name of artists along with their nationality, separated by a comma. Display the concatenated string along with the count of works by each artist, ordered by the count in descending order :-

```
154 •  SELECT * FROM artist;
155 •  CREATE VIEW view6 as
156    SELECT concat_ws(',', first_name, last_name, nationality) as Intro, full_name from artist;
157
158 •  SELECT * FROM view6;
159
160 •  CREATE VIEW Intro as
161    SELECT v.Intro as concatinated ,a.full_name,w.work_id,a.artist_id
162    FROM view6 as v
163    INNER JOIN artist as a
164    ON v.full_name = a.full_name
165    INNER JOIN work as w
166    ON a.artist_id = w.artist_id;
167
168 •  SELECT * FROM Intro;
169
170 •  SELECT concatinated, full_name, count(work_id) as Work_CNT
171    FROM Intro
172    GROUP BY concatinated,full_name
173    ORDER BY Work_CNT DESC;
```

OUTPUT:-

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

concatinated	full_name	Work_CNT
Pierre,Renoir,French	Pierre-Auguste Renoir	469
Claude,Monet,French	Claude Monet	378
Vincent,Van Gogh,Dutch	Vincent Van Gogh	308
Maurice,Utrillo,French	Maurice Utrillo	253
Albert,Marquet,French	Albert Marquet	233
Henri,Lebasque,French	Henri Lebasque	200
Camille,Pissarro,French	Camille Pissarro	188
Louis,Valtat,French	Louis Valtat	184

Result 183

Output

Action Output

#	Time	Action	Message
1	16:28:31	SELECT full_name, count(work_id) as Count_of_work FROM view2 GROUP BY full_name HAVING Count_of_...	402 row(s) returned
2	16:31:34	SELECT * FROM view3 WHERE sale_price < regular_price ORDER BY sale_price DESC LIMIT 0, 2000	2000 row(s) returned
3	16:34:24	SELECT artwork, avg(height) as avg_height, avg(width) as avg_width FROM view4 GROUP BY artwork ORDE...	2000 row(s) returned
4	16:57:05	SELECT museum_name, max(sale_price) as maximum_sale_price FROM view5 GROUP BY museum_name OR...	57 row(s) returned
5	16:59:07	SELECT concatinated, full_name, count(work_id) as Work_CNT FROM Intro GROUP BY concatinated,full_nam...	421 row(s) returned