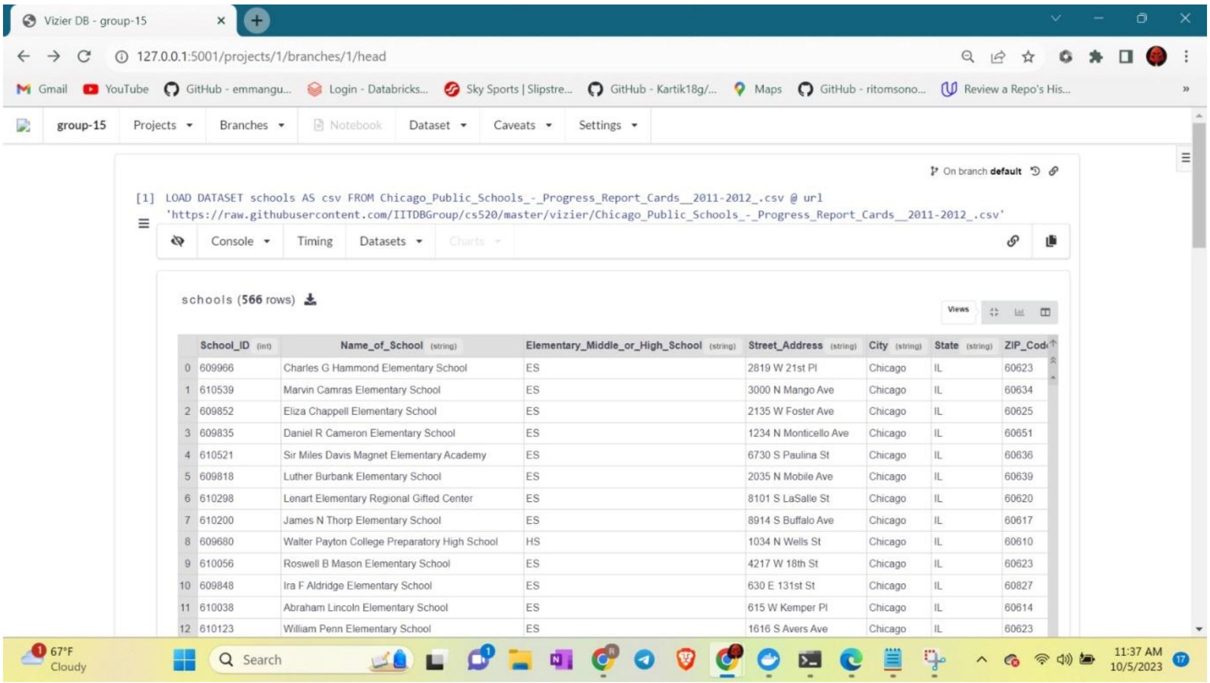
Data Warehousing, Integration and Provenance-CS520

Vizier Assignment

Group 15

Rohith Reddy Bairi rbairi@hawk.iit.edu
Deekshitha Tummala dtummala@hawk.iit.edu
Shirisha Vaddegoni svaddegoni@hawk.iit.edu
Brijesh Kumar Yadav Bandi bbandi@hawk.iit.edu

Task-1: Load a dataset and take a screenshot of the result.

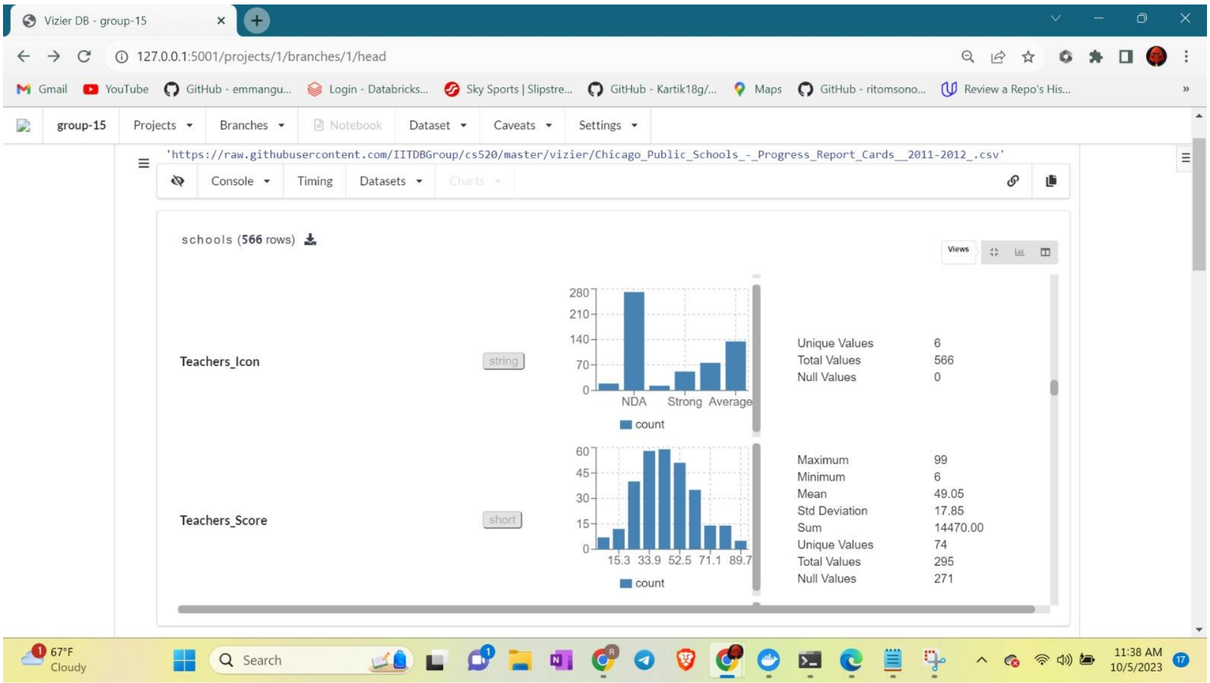


```
[1] LOAD DATASET schools AS csv FROM Chicago_Public_Schools_-_Progress_Report_Cards__2011-2012_.csv @ url
    'https://raw.githubusercontent.com/IITDBGroup/cs520/master/vizier/Chicago_Public_Schools_-_Progress_Report_Cards__2011-2012_.csv'
```

schools (566 rows)

	School_ID (int)	Name_of_School (string)	Elementary_Middle_or_High_School (string)	Street_Address (string)	City (string)	State (string)	ZIP_Cod
0	609966	Charles G Hammond Elementary School	ES	2819 W 21st Pl	Chicago	IL	60623
1	610539	Marvin Camras Elementary School	ES	3000 N Mango Ave	Chicago	IL	60634
2	609852	Eliza Chappell Elementary School	ES	2135 W Foster Ave	Chicago	IL	60625
3	609835	Daniel R Cameron Elementary School	ES	1234 N Monticello Ave	Chicago	IL	60651
4	610521	Sir Miles Davis Magnet Elementary Academy	ES	6730 S Paulina St	Chicago	IL	60636
5	609818	Luther Burbank Elementary School	ES	2035 N Mobile Ave	Chicago	IL	60639
6	610298	Lenart Elementary Regional Gifted Center	ES	8101 S LaSalle St	Chicago	IL	60620
7	610200	James N Thorp Elementary School	ES	8914 S Buffalo Ave	Chicago	IL	60617
8	609680	Walter Payton College Preparatory High School	HS	1034 N Wells St	Chicago	IL	60610
9	610056	Roswell B Mason Elementary School	ES	4217 W 18th St	Chicago	IL	60623
10	609848	Ira F Aldridge Elementary School	ES	630 E 131st St	Chicago	IL	60827
11	610038	Abraham Lincoln Elementary School	ES	615 W Kemper Pl	Chicago	IL	60614
12	610123	William Penn Elementary School	ES	1616 S Avers Ave	Chicago	IL	60623

Task-2: Select the detail view and look at the distributions of some columns. Then look at the column view and take a screenshot of the distribution for column Teachers_Score.



'https://raw.githubusercontent.com/IITDBGroup/cs520/master/vizier/Chicago_Public_Schools_-_Progress_Report_Cards__2011-2012_.csv'

schools (566 rows)

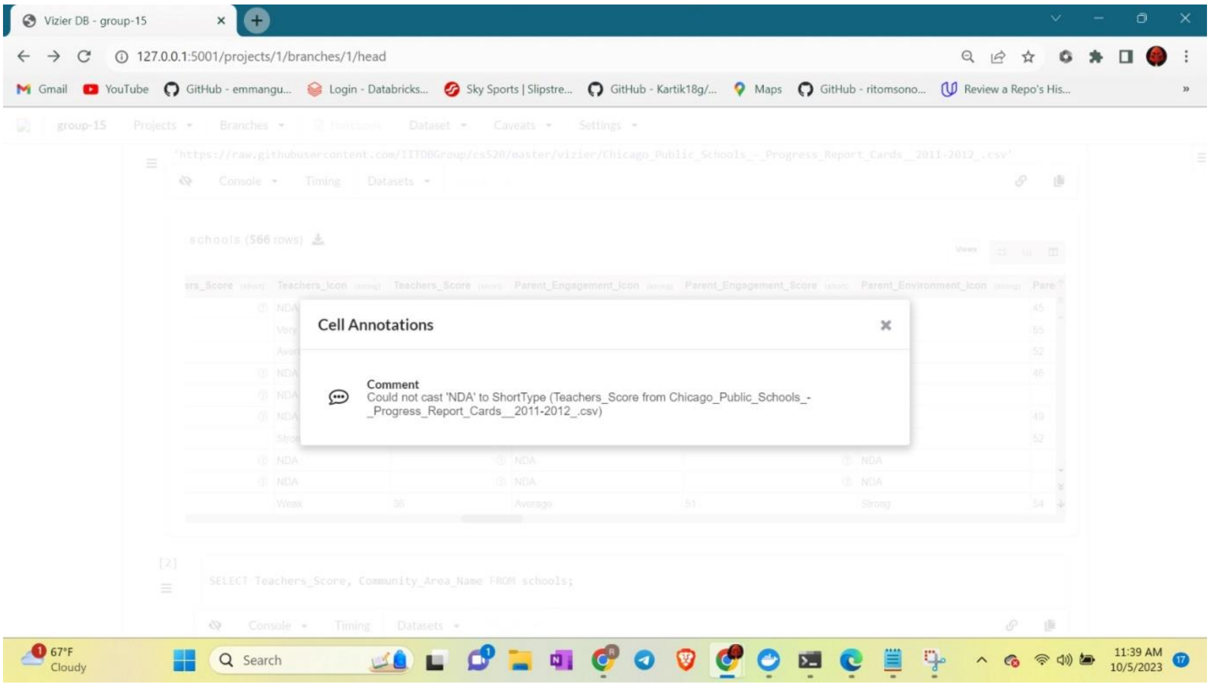
Teachers_Icon string

Unique Values	6
Total Values	566
Null Values	0

Teachers_Score short

Maximum	99
Minimum	6
Mean	49.05
Std Deviation	17.85
Sum	14470.00
Unique Values	74
Total Values	295
Null Values	271

Task-3: Click on one of the question marks for values in the teachers column and take a screenshot.

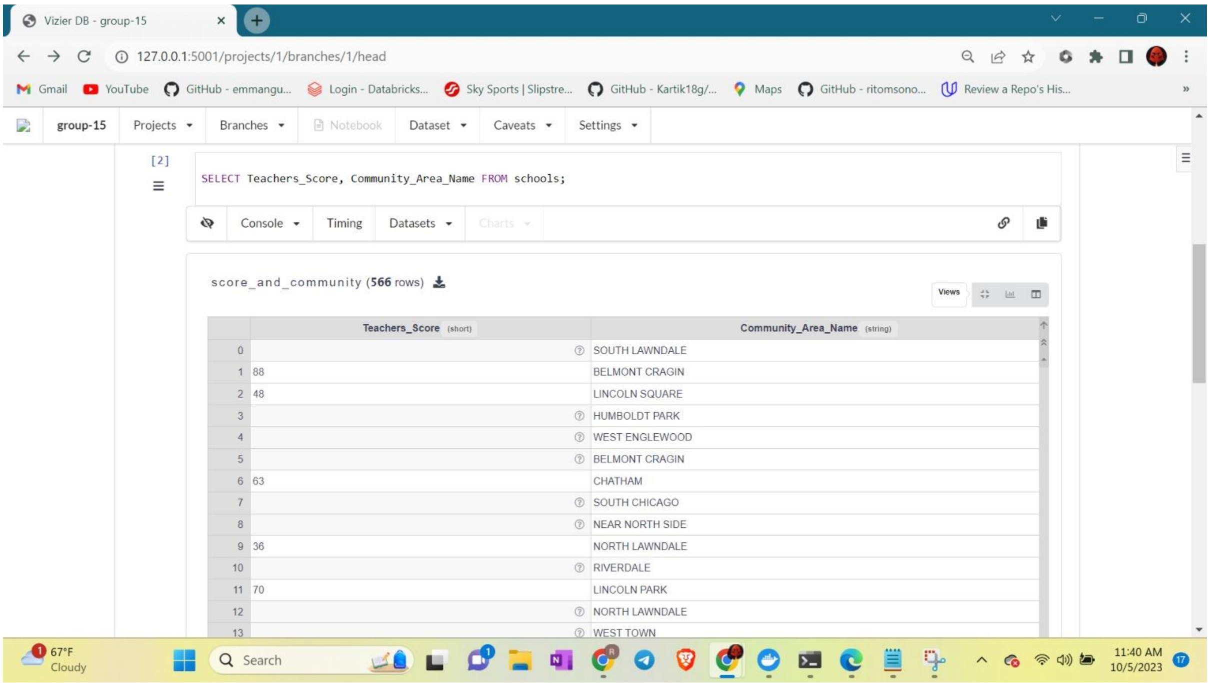


Cell Annotations

Comment
Could not cast 'NDA' to ShortType (Teachers_Score from Chicago_Public_Schools_-_Progress_Report_Cards__2011-2012_.csv)

SELECT Teachers_Score, Community_Area_Name FROM schools;

Task-4: Create a SQL cell and write a query that returns columns Teachers_Score and Community_Area_Name. SQL results can be stored as new dataset score_and_community. And take a screenshot of the result.

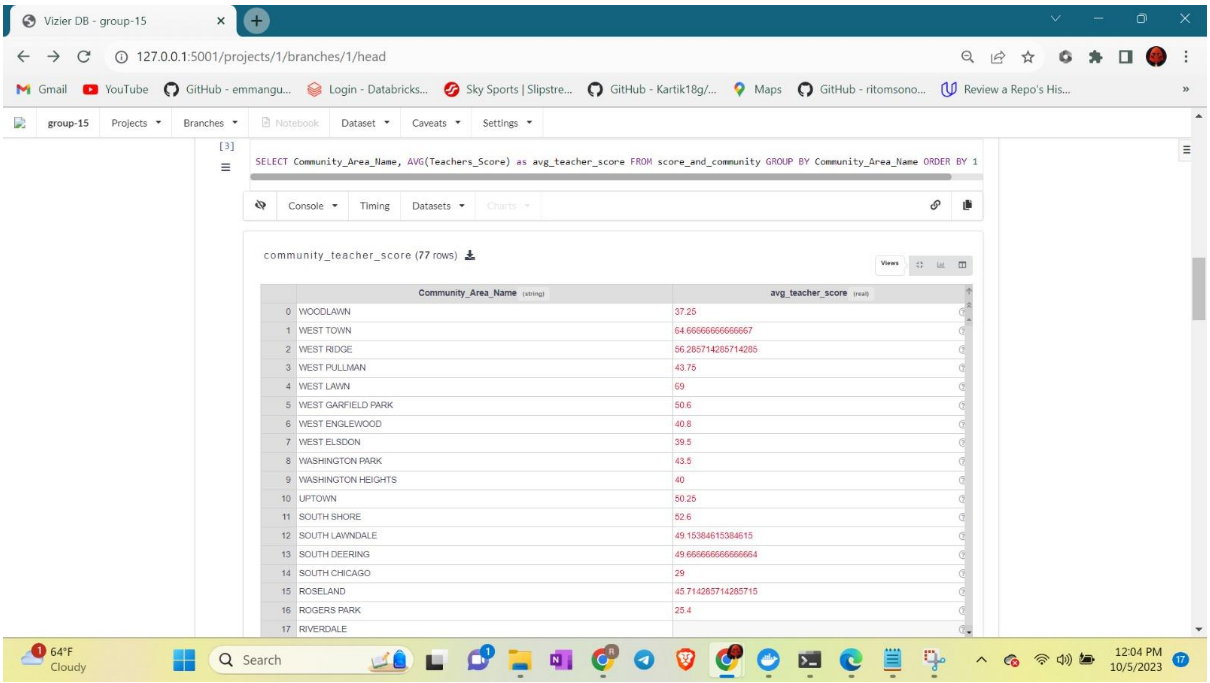


```
SELECT Teachers_Score, Community_Area_Name FROM schools;
```

score_and_community (566 rows)

	Teachers_Score (short)	Community_Area_Name (string)
0		SOUTH LAWNDALE
1	88	BELMONT CRAGIN
2	48	LINCOLN SQUARE
3		HUMBOLDT PARK
4		WEST ENGLEWOOD
5		BELMONT CRAGIN
6	63	CHATHAM
7		SOUTH CHICAGO
8		NEAR NORTH SIDE
9	36	NORTH LAWNDALE
10		RIVERDALE
11	70	LINCOLN PARK
12		NORTH LAWNDALE
13		WEST TOWN

Task-5: Create a SQL cell and write a query over the score_and_community dataset that computes the result as described above. Call the result dataset community_teacher_scores. And take a screenshot of the result.

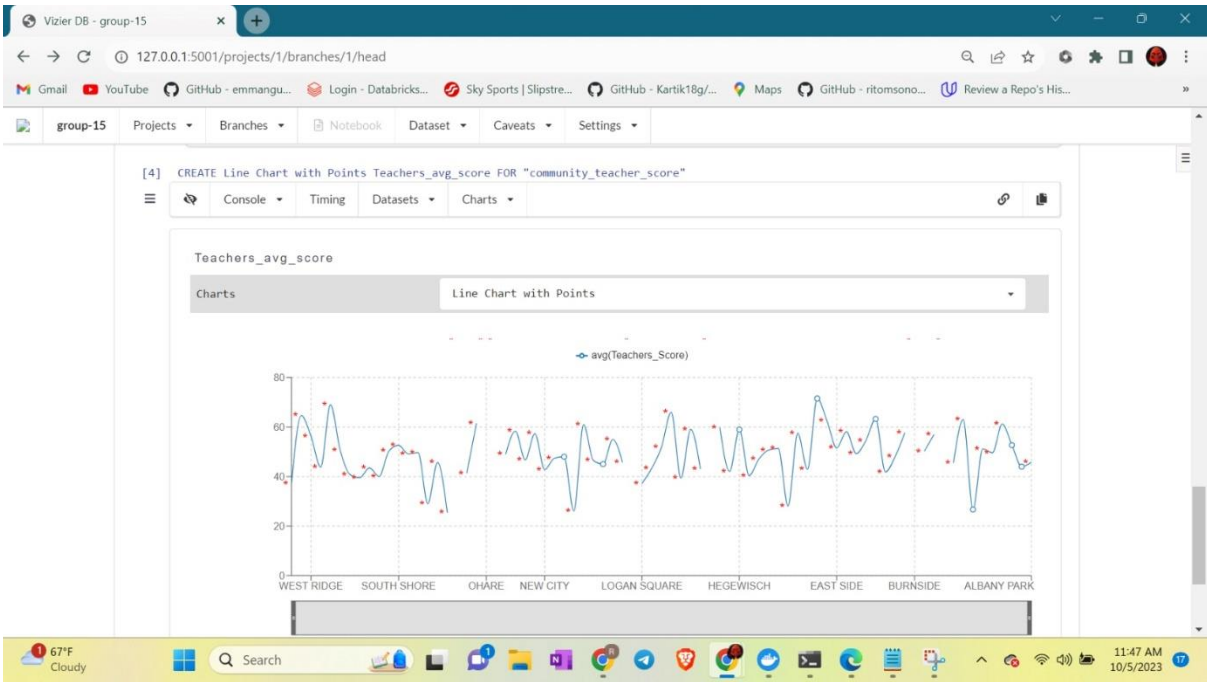


```
SELECT Community_Area_Name, AVG(Teachers_Score) as avg_teacher_score FROM score_and_community GROUP BY Community_Area_Name ORDER BY 1
```

community_teacher_score (77 rows)

	Community_Area_Name (string)	avg_teacher_score (real)
0	WOODLAWN	37.25
1	WEST TOWN	64.66666666666667
2	WEST RIDGE	56.285714285714285
3	WEST PULLMAN	43.75
4	WEST LAWN	69
5	WEST GARFIELD PARK	50.6
6	WEST ENGLEWOOD	40.8
7	WEST ELSDON	39.5
8	WASHINGTON PARK	43.5
9	WASHINGTON HEIGHTS	40
10	UPTOWN	50.25
11	SOUTH SHORE	52.6
12	SOUTH LAWNDALE	49.15384615384615
13	SOUTH DEERING	49.666666666666664
14	SOUTH CHICAGO	29
15	ROSELAND	45.714285714285715
16	ROGERS PARK	25.4
17	RIVERDALE	

Task-6: Create a line chart of the aggregation result by creating a plot cell and take a screenshot of the result.

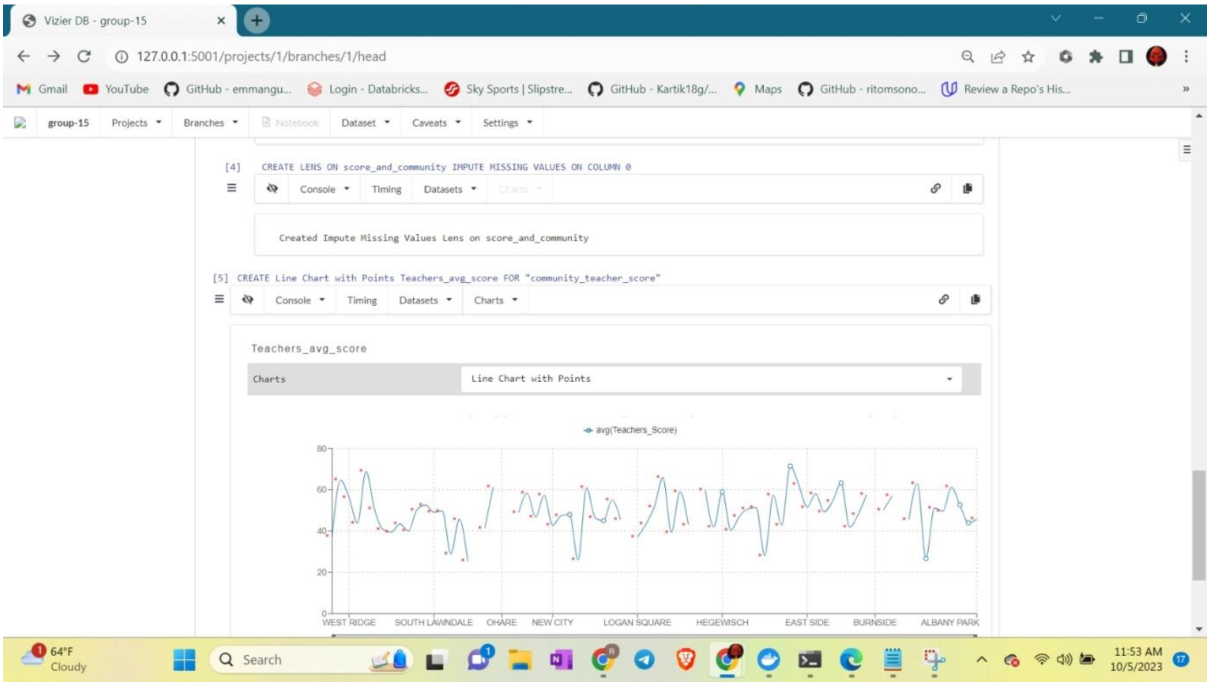


```
[4]  CREATE Line Chart with Points Teachers_avg_score FOR "community_teacher_score"
```

Teachers_avg_score

Charts: Line Chart with Points

Task-7: Insert a new cell above the SQL cell that computes the average teacher scores (notebooks in Vizier are executed top down) by pressing the three bars below the cell number. Select *"Impute Missing Values"*, select the score_and_community dataset and Teachers_Score as the column to be imputed, and select mean as the imputation method and take a screenshot of the updated line chart.



```
[4]  CREATE LENS ON score_and_community IMPUTE MISSING VALUES ON COLUMN 0
```

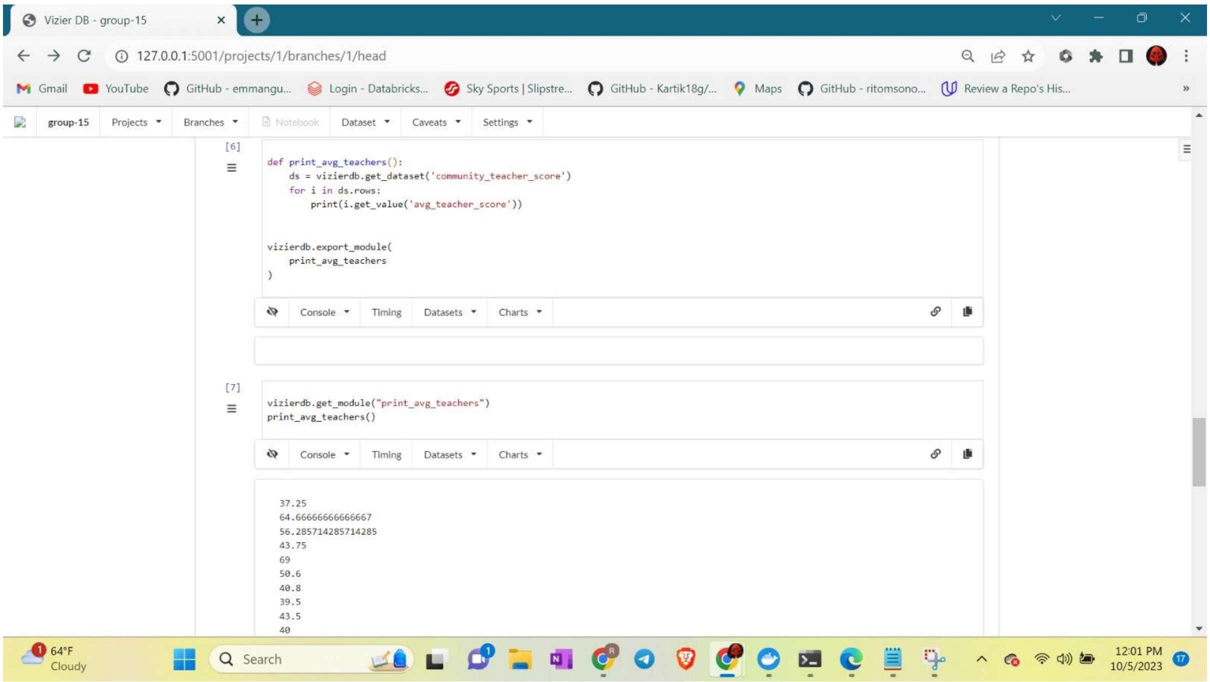
```
Created Impute Missing Values Lens on score_and_community
```

```
[5]  CREATE Line Chart with Points Teachers_avg_score FOR "community_teacher_score"
```

Teachers_avg_score

Charts: Line Chart with Points

Task-8: Create a Python cell at the end of the notebook and create a function called print_avg_teachers that uses Vizier's API to get a handle for this dataset and print all values of the avg_teacher_score column. Hint: Use the “Show Code Examples” button to see example Vizier API usage and see here for the API documentation. Then use vizierdb.export_module to export the function. Then create a second Python cell and use vizierdb.get_model("print_avg_teachers") for importing the function and then call it. Take a screenshot of the result.



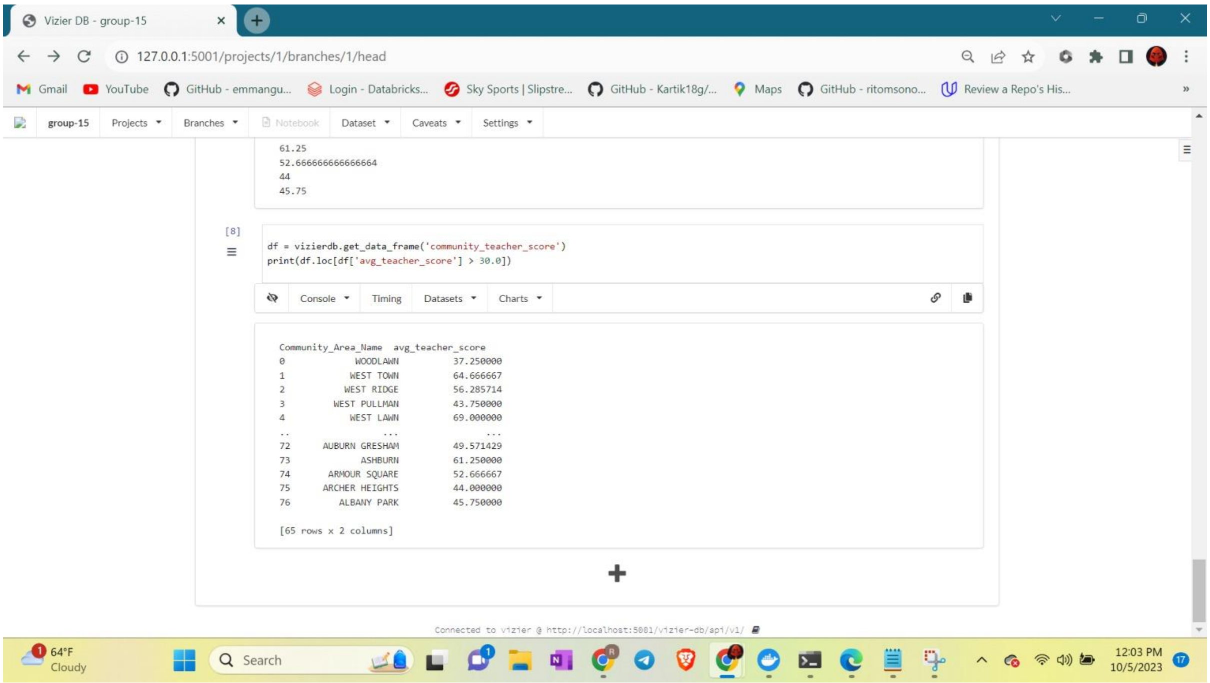
```
def print_avg_teachers():
    ds = vizierdb.get_dataset('community_teacher_score')
    for i in ds.rows:
        print(i.get_value('avg_teacher_score'))


vizierdb.export_module(
    print_avg_teachers
)
```

```
vizierdb.get_module("print_avg_teachers")
print_avg_teachers()
```

```
37.25
64.66666666666667
56.285714285714285
43.75
69
50.6
40.8
39.5
43.5
40
```

Task-9: Create another Python cell and use Vizier’s API to access the dataset community_teacher_scores as a DataFrame, then filter out rows where the avg_teacher_score is larger than or equal to 30.0 and then print the remaining rows and take a screenshot.



```
61.25
52.666666666666664
44
45.75
```

```
df = vizierdb.get_data_frame('community_teacher_score')
print(df.loc[df['avg_teacher_score'] > 30.0])
```

```
Community_Area_Name  avg_teacher_score
0             WOODLAWN          37.250000
1            WEST TOWN          64.666667
2           WEST RIDGE          56.285714
3         WEST PULLMAN          43.750000
4            WEST LAWN          69.000000
..                 ...                ...
72      AUBURN GRESHAM          49.571429
73            ASHBURN          61.250000
74       ARMOUR SQUARE          52.666667
75      ARCHER HEIGHTS          44.000000
76         ALBANY PARK          45.750000

[65 rows x 2 columns]
```