Code Basics

Telangana Government Tourism Department Analysis

By Monesh S

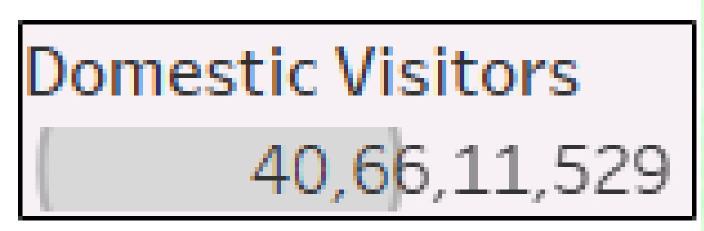
Agenda

- Introduction
- Tools Used
- Preprocessing
- Implementation
- Conclusions

Introduction

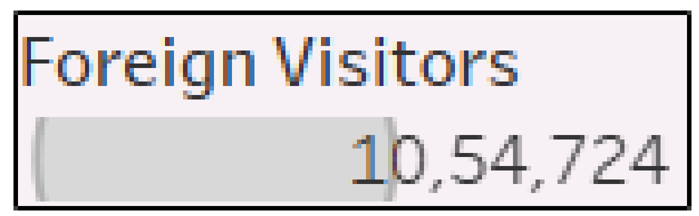
- Dataset consists of number of Domestic and Foreign Visitors to Telangana grouped by Districts with their Arrival Dates
- Domestic and Foreign Visitors are divided into seperate excel sheets and each of them are also seperated by years(2016-2019).

Total Visitors to each Districts



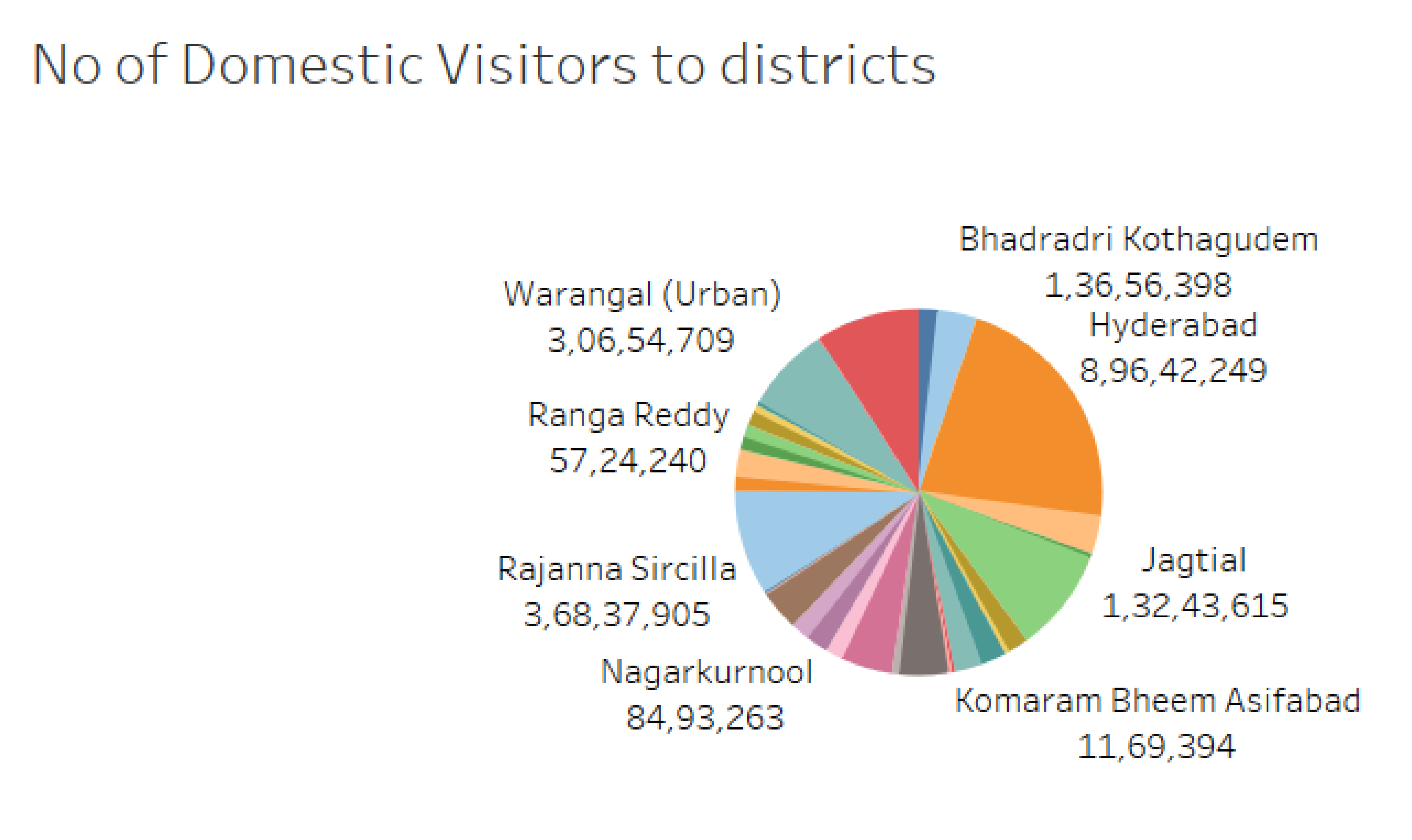
Domestic Visitors

40,66,11,529

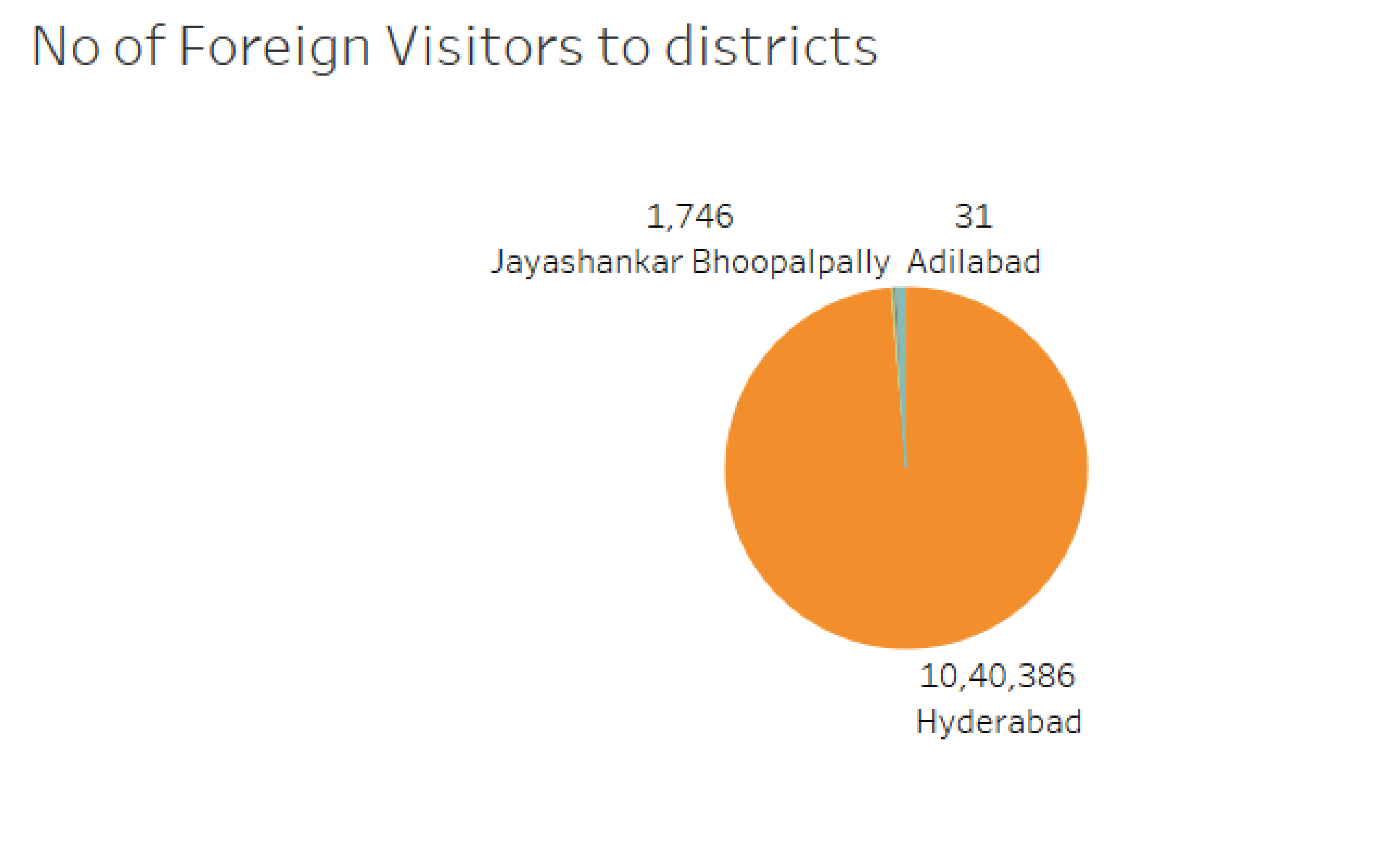


Foreign Visitors

10,54,724



No of Domestic Visitors to districts



No of Foreign Visitors to districts

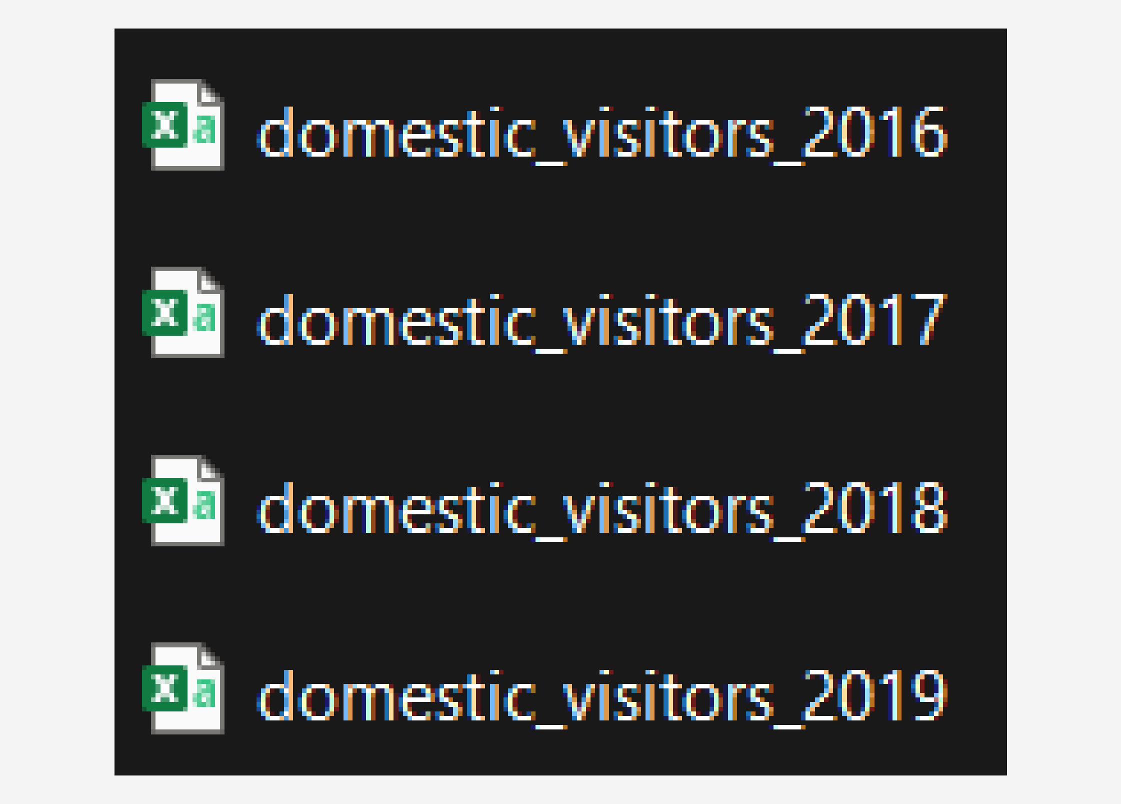
Tools Used

Data Transformation: Python

Data Analysis: SQL

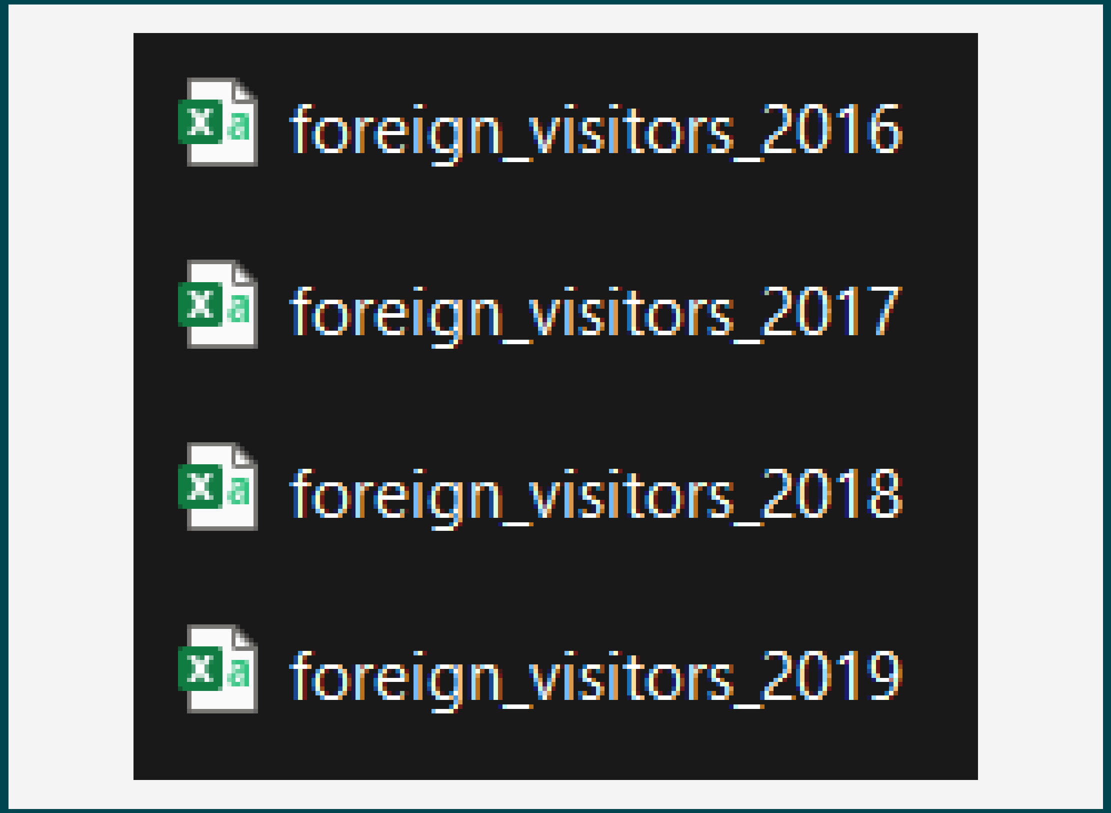
Data Visualization: Tableau

Preprocessing



domestic_visitors_2016
domestic_visitors_2017
domestic_visitors_2018
domestic_visitors_2019

- Domestic Visitors Data yearwise



foreign_visitors_2016
foreign_visitors_2017
foreign_visitors_2018
foreign_visitors_2019

- Foreign Visitors Data yearwise

Since the Data is split into different excel sheets we need to combine them as domestic and foreign visitors respectively using python and preprocess the data Workflow is as follows:

1. Import Pandas to read excel files and concatenate the sum.

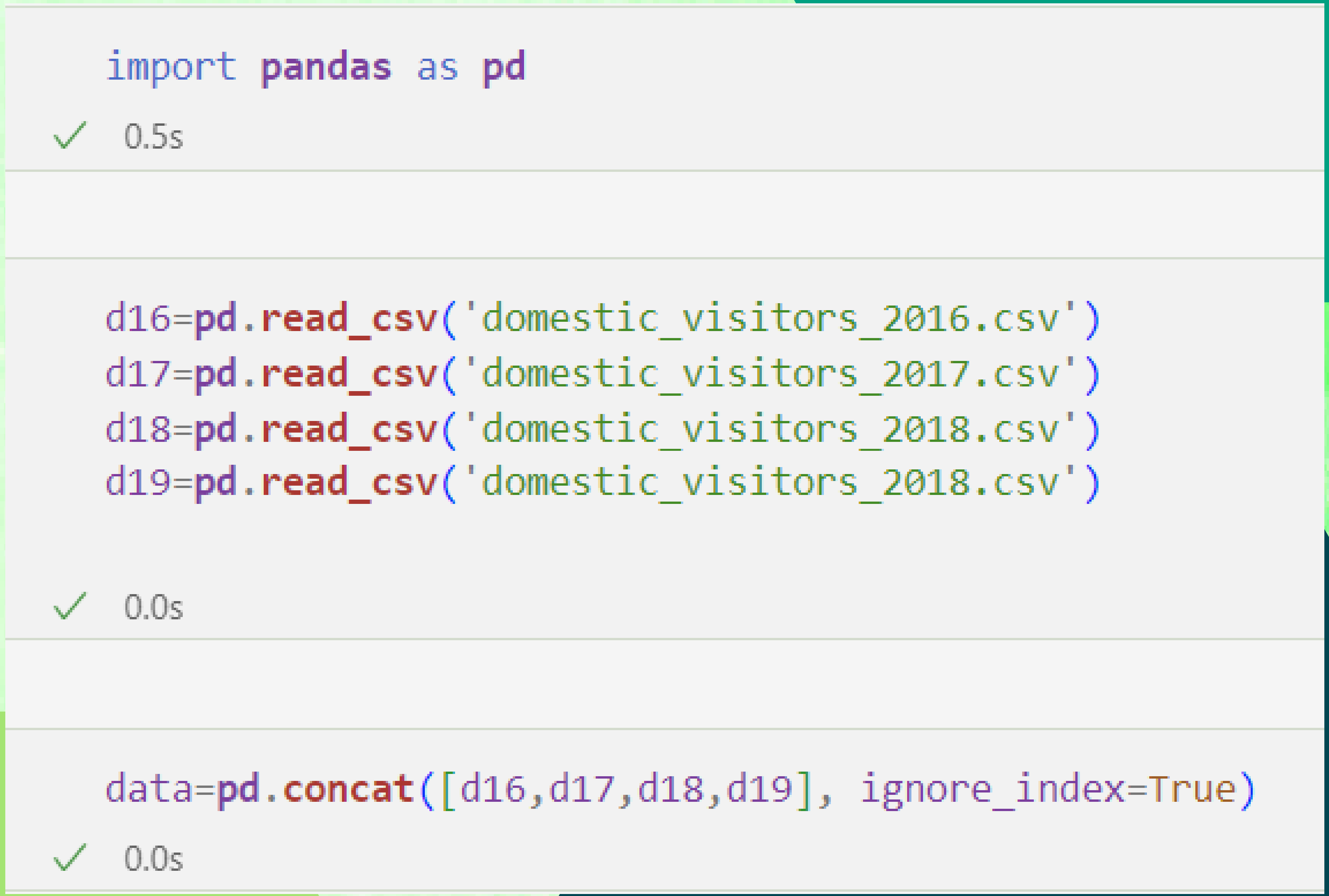
2. Check the Data formats using function and change the datatype as per requirement.

3. Check if all the values are present if not fill the values using functions.

4. If the data is preprocessed and no more modification is left then save the file as csv.

1.

- Importing Libraries
- Read CSV files
- Concatenating files



```
import pandas as pd
```

✓ 0.5s

```
d16=pd.read_csv('domestic_visitors_2016.csv')
d17=pd.read_csv('domestic_visitors_2017.csv')
d18=pd.read_csv('domestic_visitors_2018.csv')
d19=pd.read_csv('domestic_visitors_2018.csv')
```

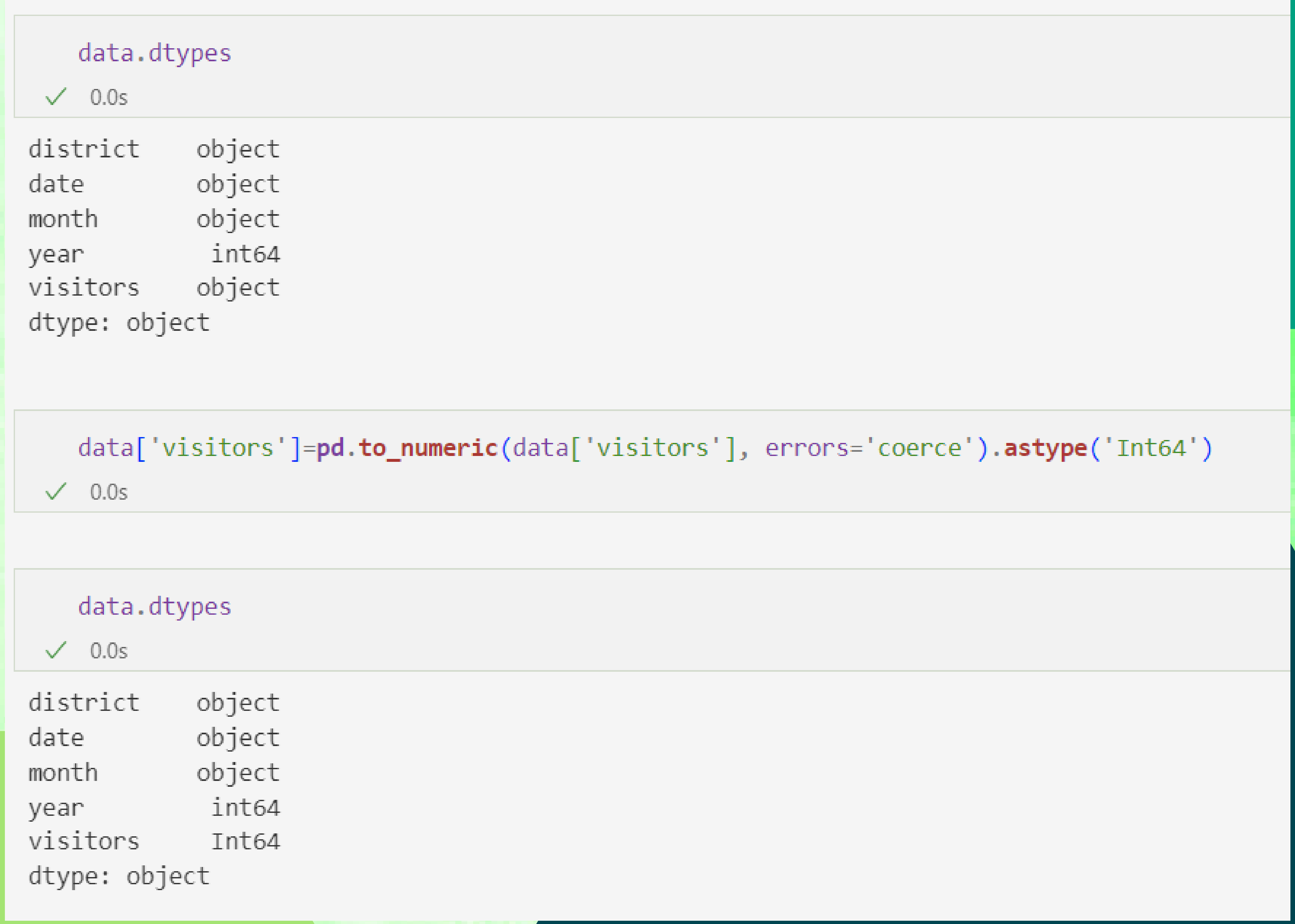
✓ 0.0s

```
data=pd.concat([d16,d17,d18,d19], ignore_index=True)
```

✓ 0.0s

2.

- Check Data Types
- Change Data Type using functions as per Requirement



```
data.dtypes
```

✓ 0.0s

```
district    object
date        object
month       object
year         int64
visitors    object
dtype: object
```

```
data['visitors']=pd.to_numeric(data['visitors'], errors='coerce').astype('Int64')
```

✓ 0.0s

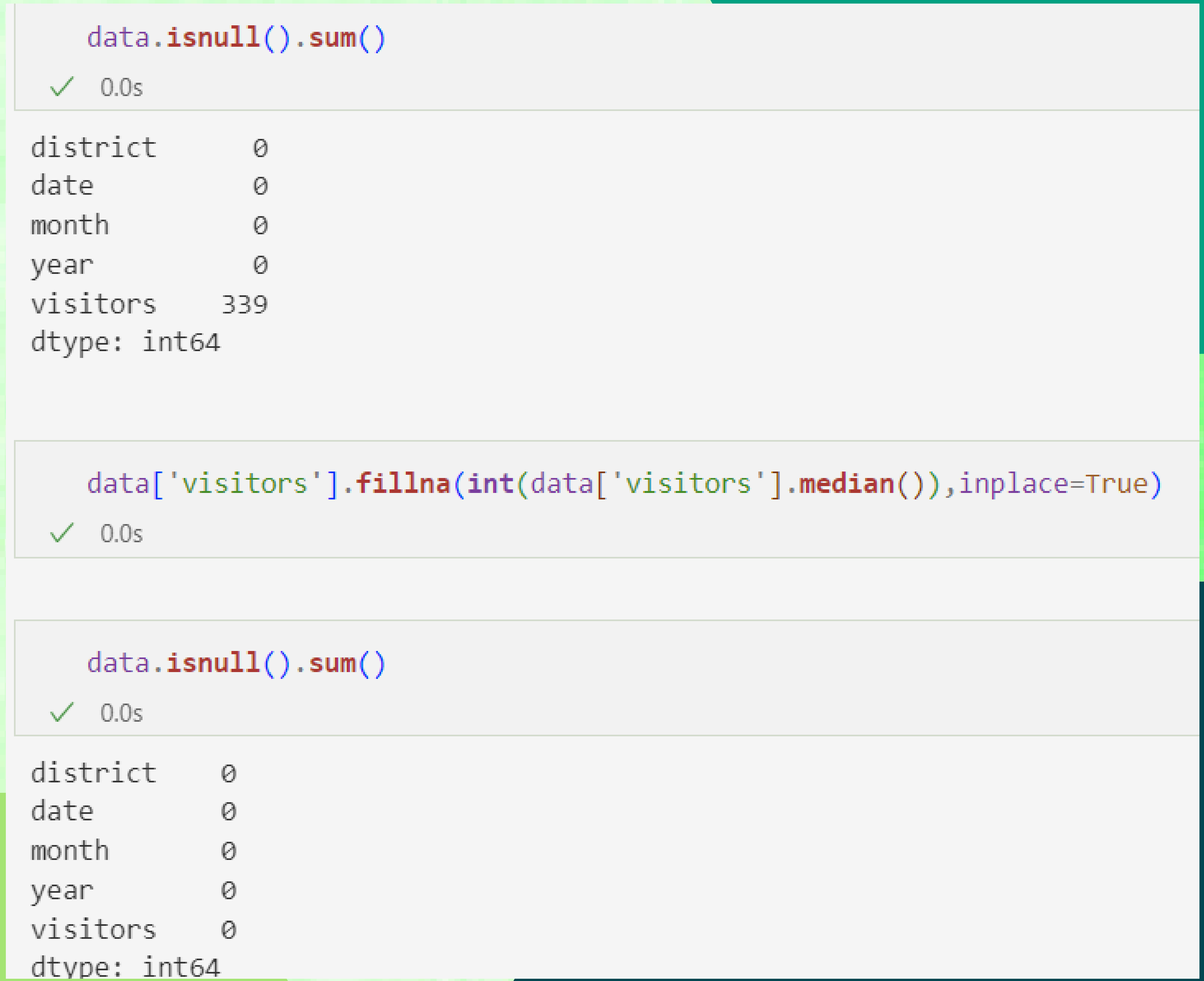
```
data.dtypes
```

✓ 0.0s

```
district    object
date        object
month       object
year         int64
visitors     Int64
dtype: object
```

3.

- Check if Data has Null values
- If it has null values fill using fillna methods.



```
data.isnull().sum()
```

✓ 0.0s

```
district      0
date          0
month         0
year          0
visitors    339
dtype: int64
```

```
data['visitors'].fillna(int(data['visitors'].median()),inplace=True)
```

✓ 0.0s

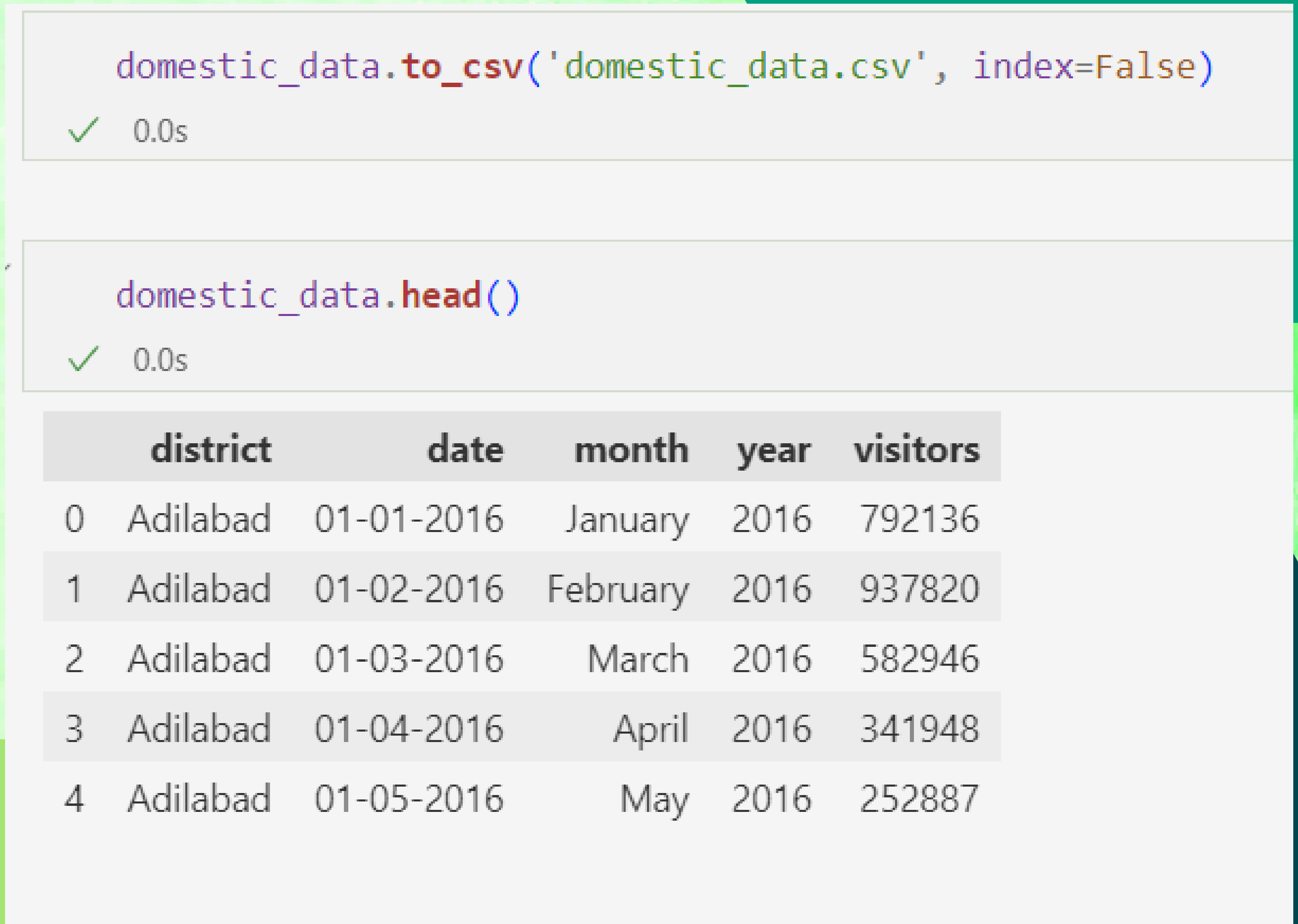
```
data.isnull().sum()
```

✓ 0.0s

```
district    0
date        0
month       0
year        0
visitors    0
dtype: int64
```

4.

- Save the concatenated file to csv.
- Check data once using head function



```
domestic_data.to_csv('domestic_data.csv', index=False)
```

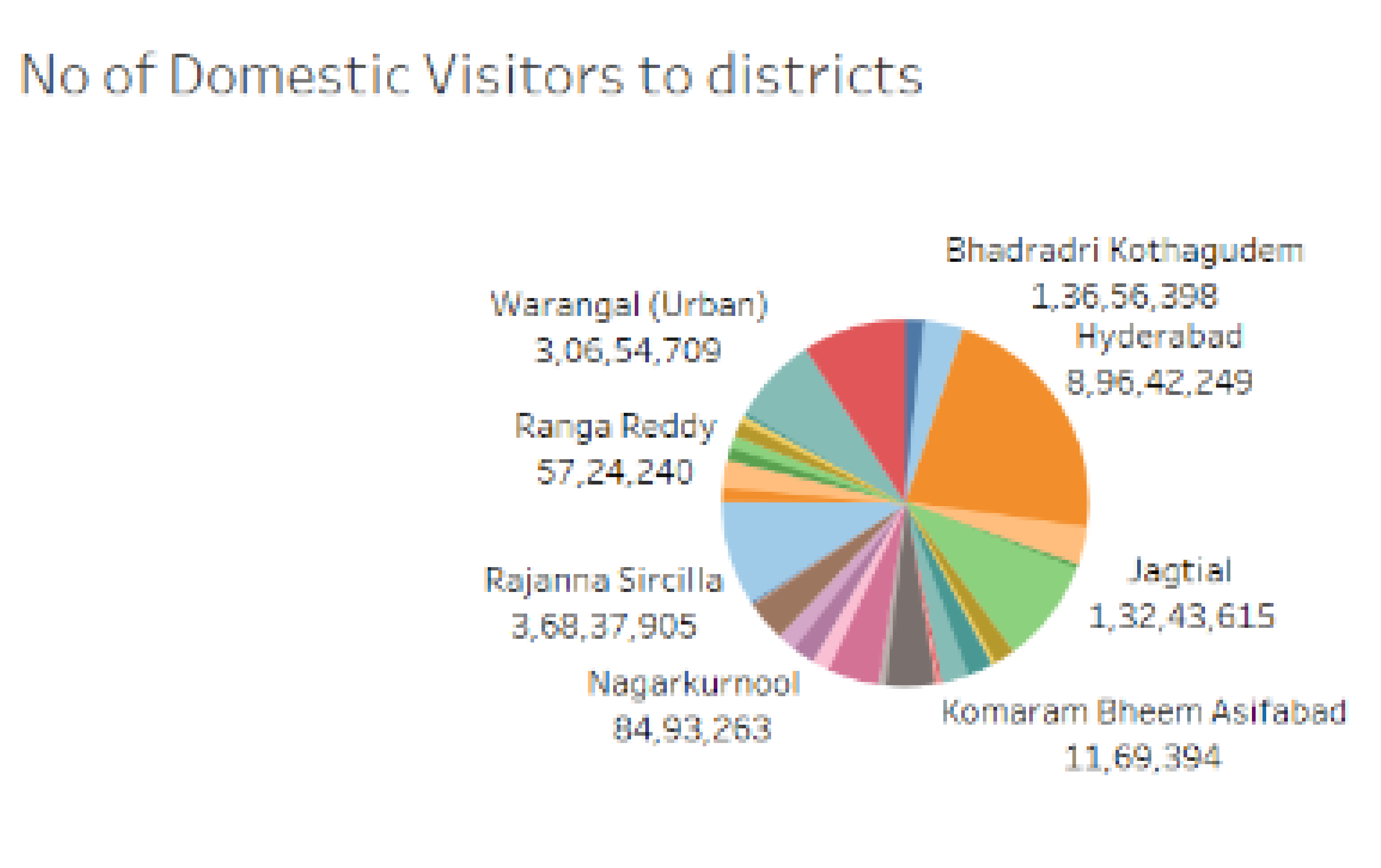
✓ 0.0s

```
domestic_data.head()
```

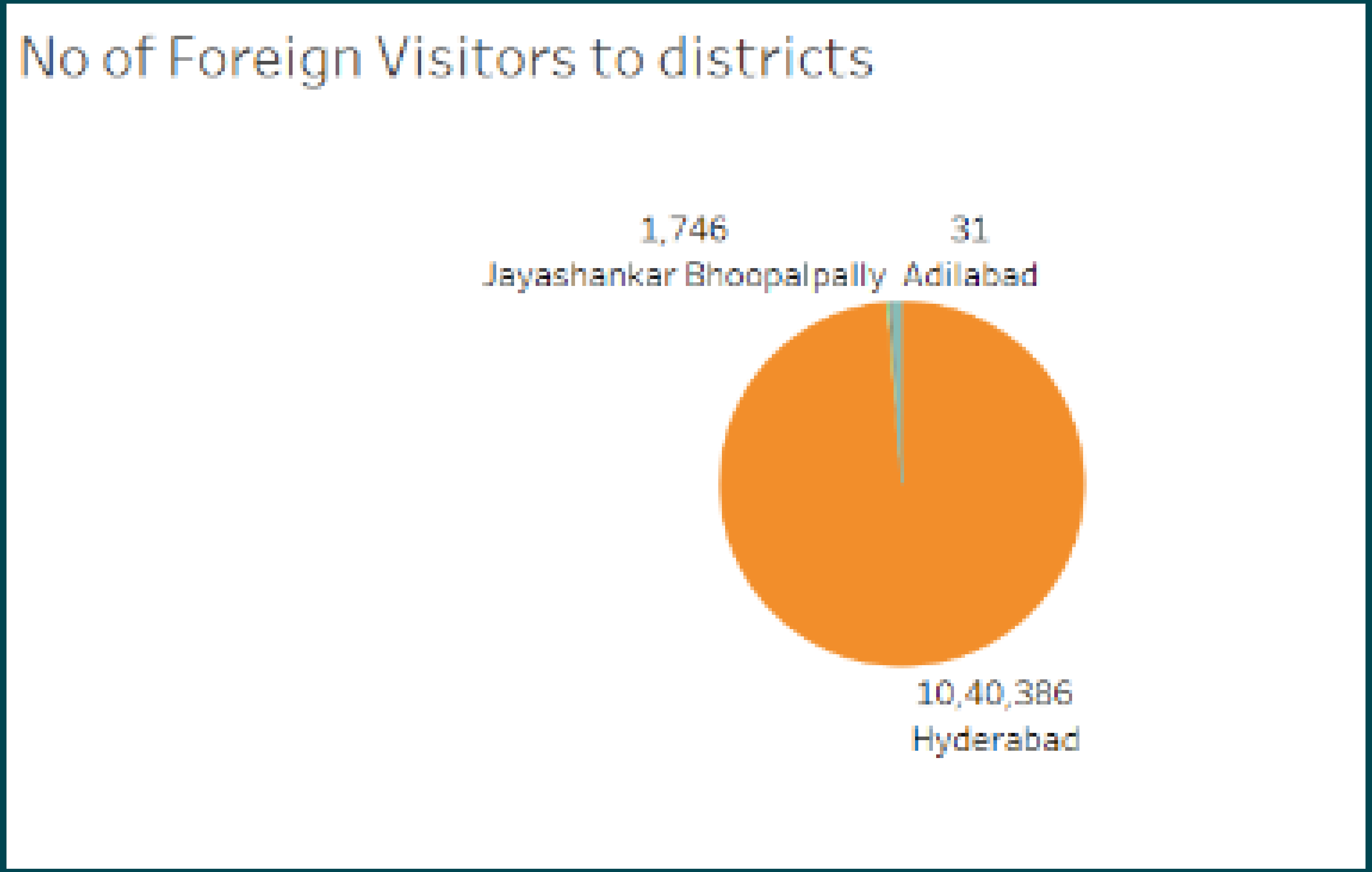
✓ 0.0s

	district	date	month	year	visitors
0	Adilabad	01-01-2016	January	2016	792136
1	Adilabad	01-02-2016	February	2016	937820
2	Adilabad	01-03-2016	March	2016	582946
3	Adilabad	01-04-2016	April	2016	341948
4	Adilabad	01-05-2016	May	2016	252887

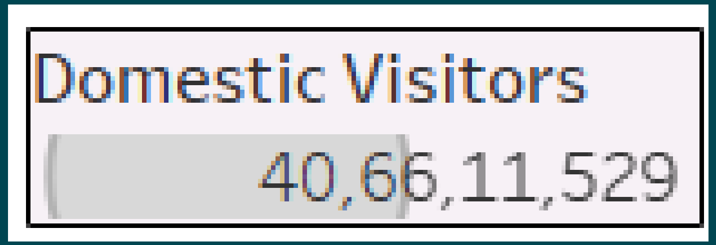
Implementation



No of Domestic Visitors to districts

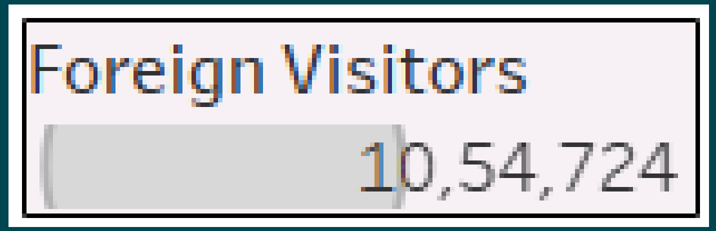


No of Foreign Visitors to districts



Domestic Visitors

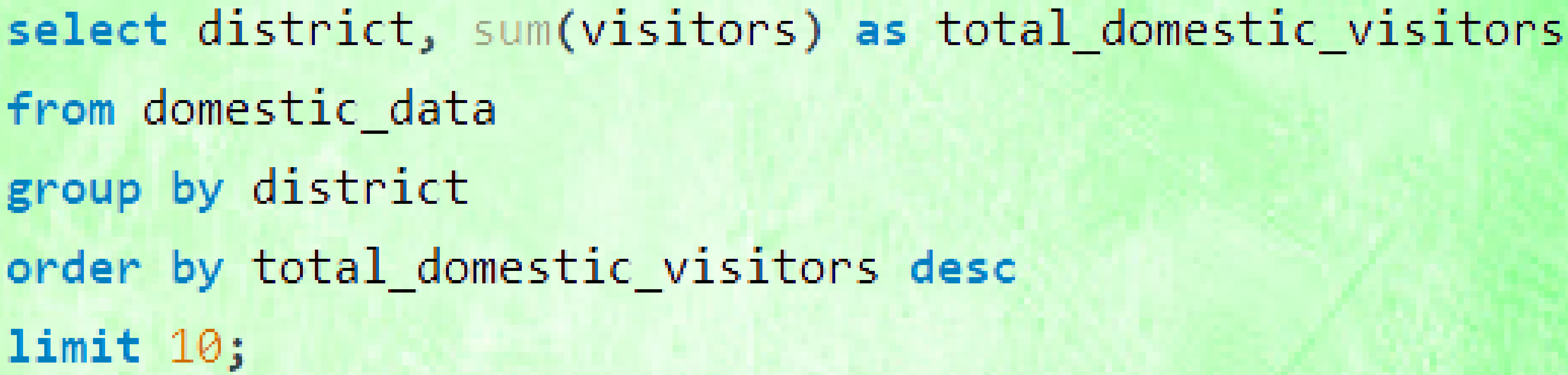
40,66,11,529



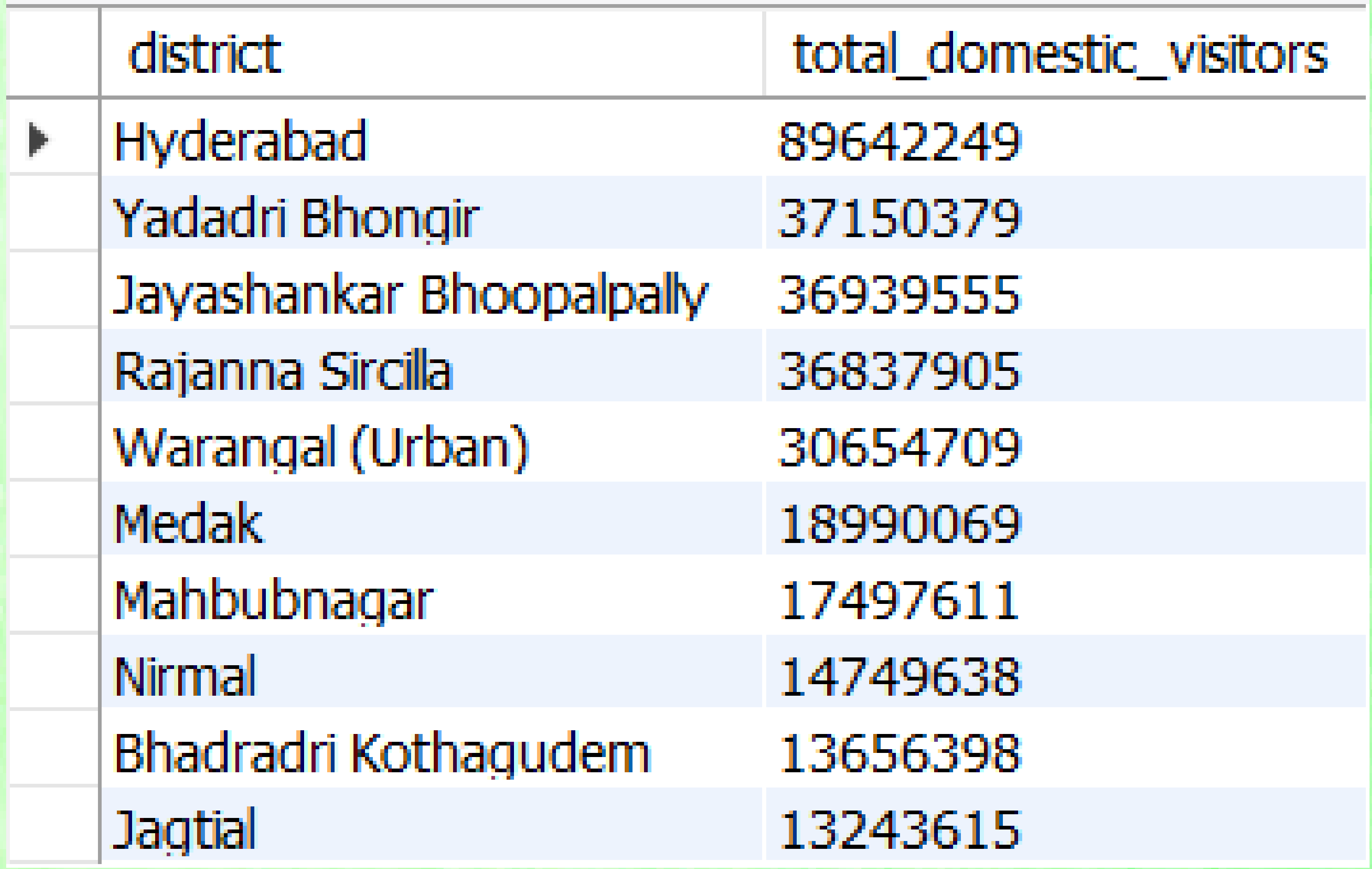
Foreign Visitors

10,54,724

1.Top 10 Districts that has Highest Number of Domestic Visitors Overall (2016-2019)

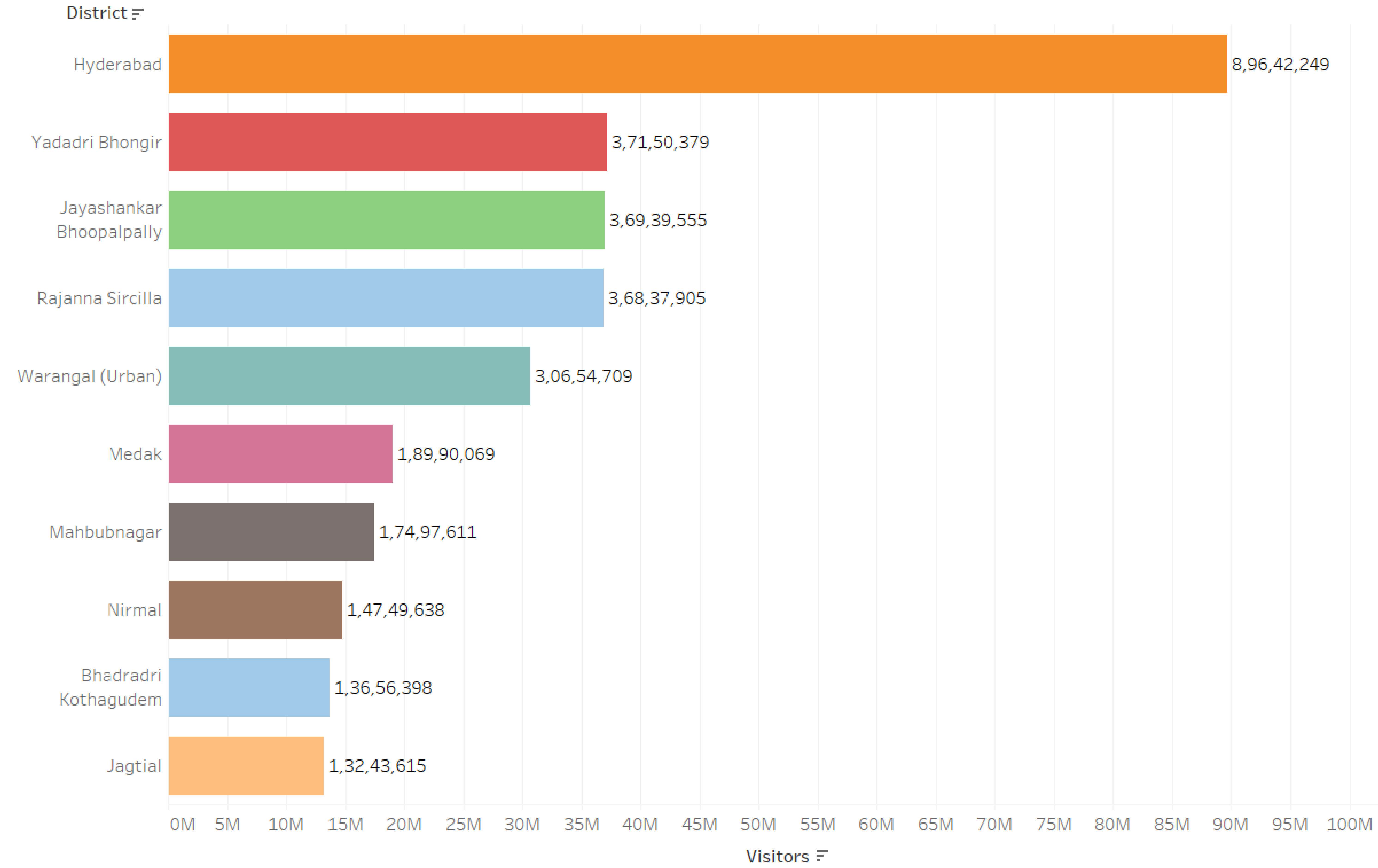


```sql
select district, sum(visitors) as total_domestic_visitors
from domestic_data
group by district
order by total_domestic_visitors desc
limit 10;
```



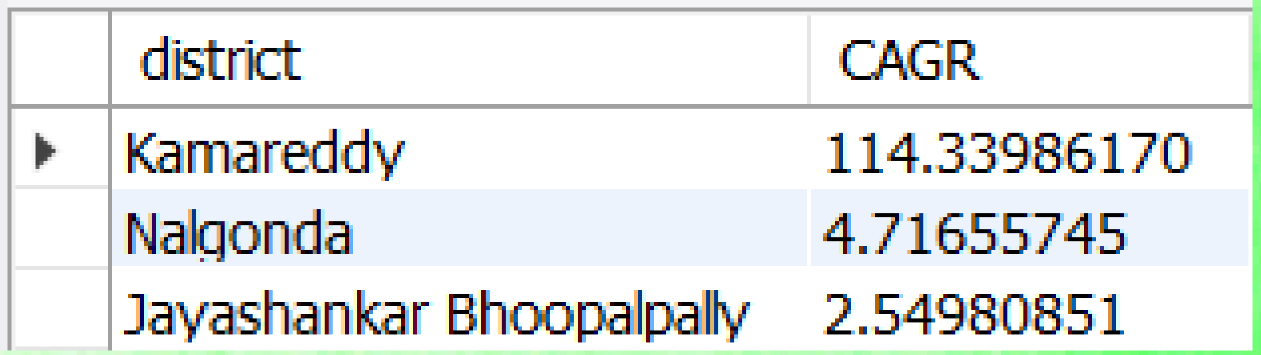
district	total_domestic_visitors
Hyderabad	89642249
Yadadri Bhongir	37150379
Jayashankar Bhoopalpally	36939555
Rajanna Sircilla	36837905
Warangal (Urban)	30654709
Medak	18990069
Mahbubnagar	17497611
Nirmal	14749638
Bhadradri Kothagudem	13656398
Jagtial	13243615

Top 10 Districts Having Highest Number of Domestic Visitors Overall

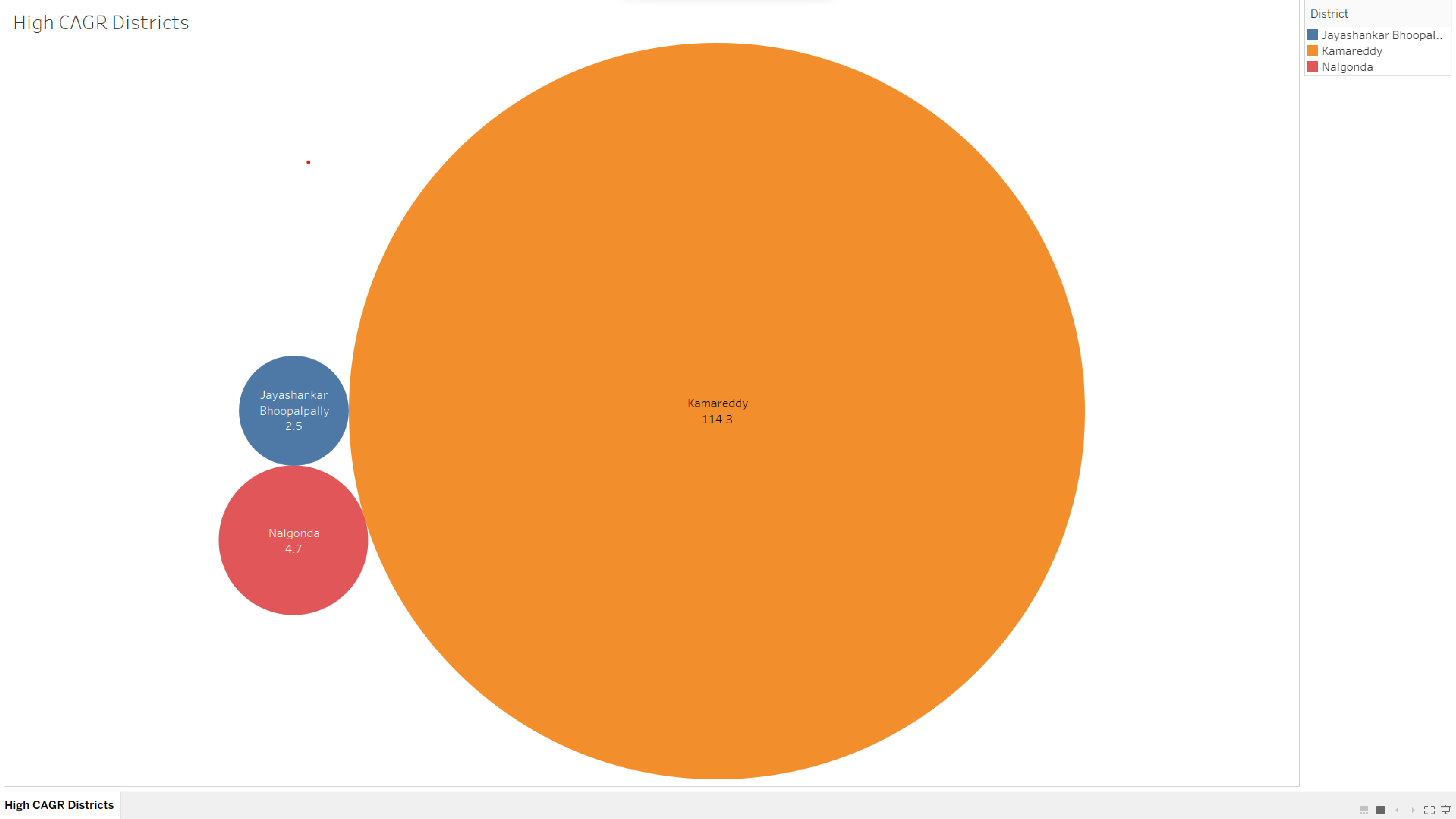


2.Top 3 Districts having Highest CAGR Value of Visitors

```sql
select district, avg(cagr) as CAGR
from
(
    select district, year, visitors,
    visitors/prev_visitors - 1 as cagr
    from
       (
            select *, lag(visitors, 1) over
            (partition by district order by year) as prev_visitors
            from domestic_data
    )a
)b
group by district
order by CAGR desc
limit 3;
```



district	CAGR
Kamareddy	114.33986170
Nalgonda	4.71655745
Jayashankar Bhoopalpally	2.54980851



High CAGR Districts

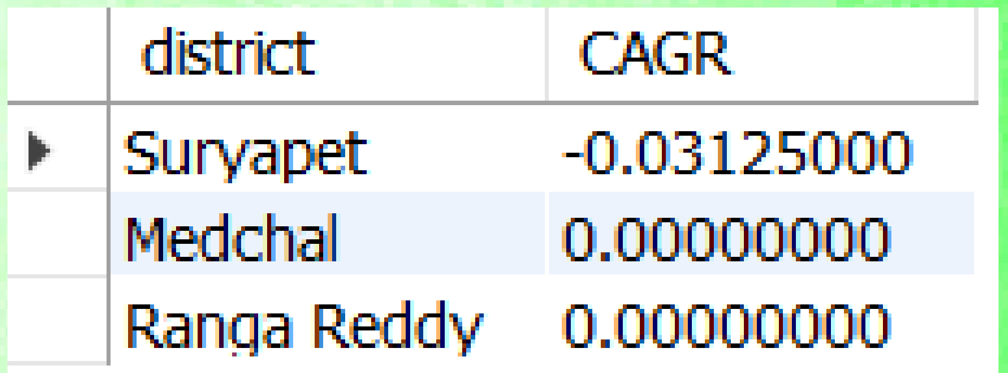
District

- 🟦 Jayashankar Bhoopal..
- 🟧 Kamareddy
- 🟥 Nalgonda

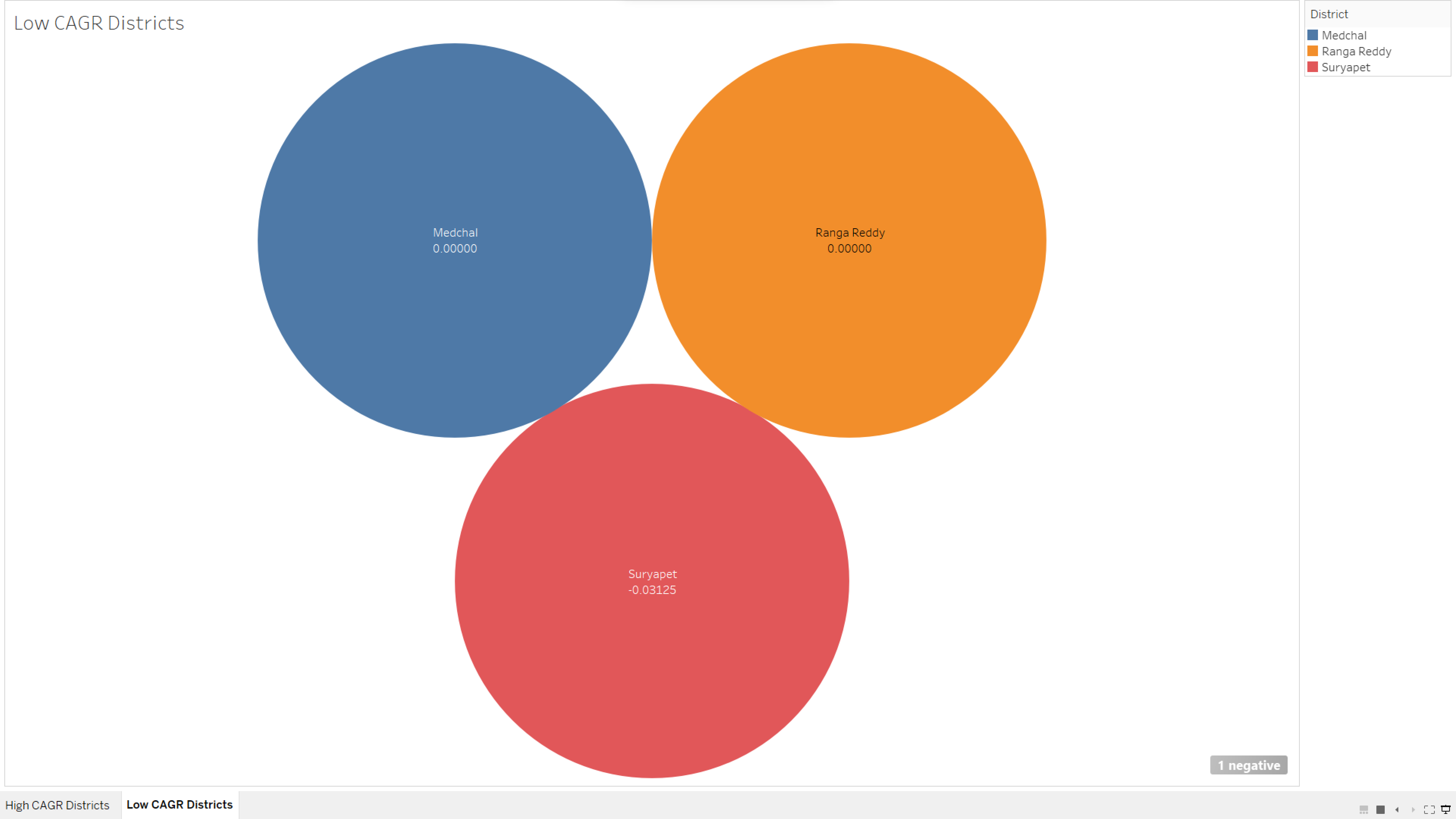
High CAGR Districts

3.Bottom 3 Districts having Lowest CAGR Value of Visitors

```
select district, avg(cagr) as CAGR
from
(
    select district, year, visitors,
    visitors/prev_visitors - 1 as cagr
    from
       (
            select *, lag(visitors, 1) over
            (partition by district order by year) as prev_visitors
            from domestic_data
    )a
)b
group by district
order by CAGR
limit 3;
```



district	CAGR
Suryapet	-0.03125000
Medchal	0.00000000
Ranga Reddy	0.00000000

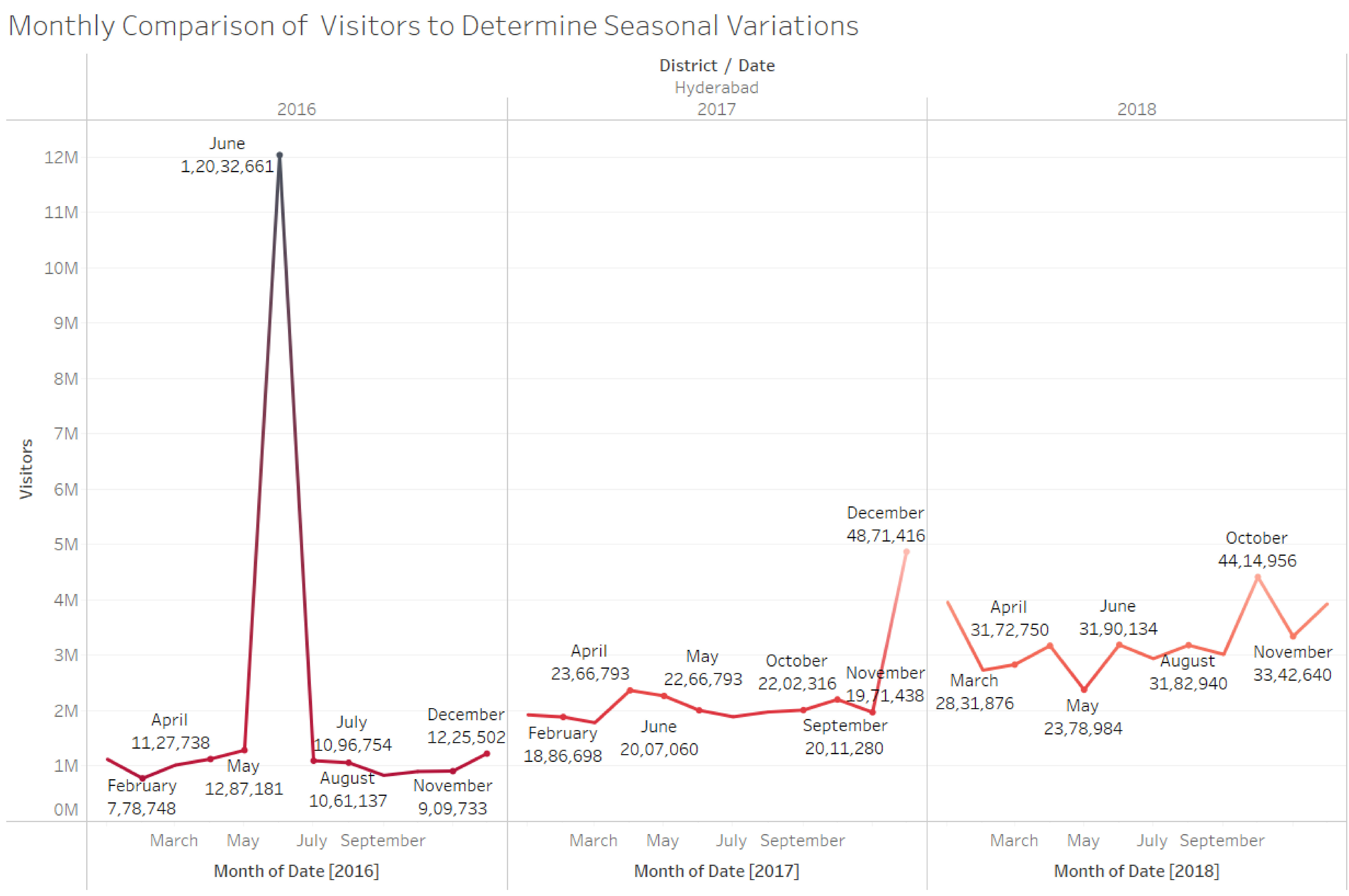


Low CAGR Districts

District

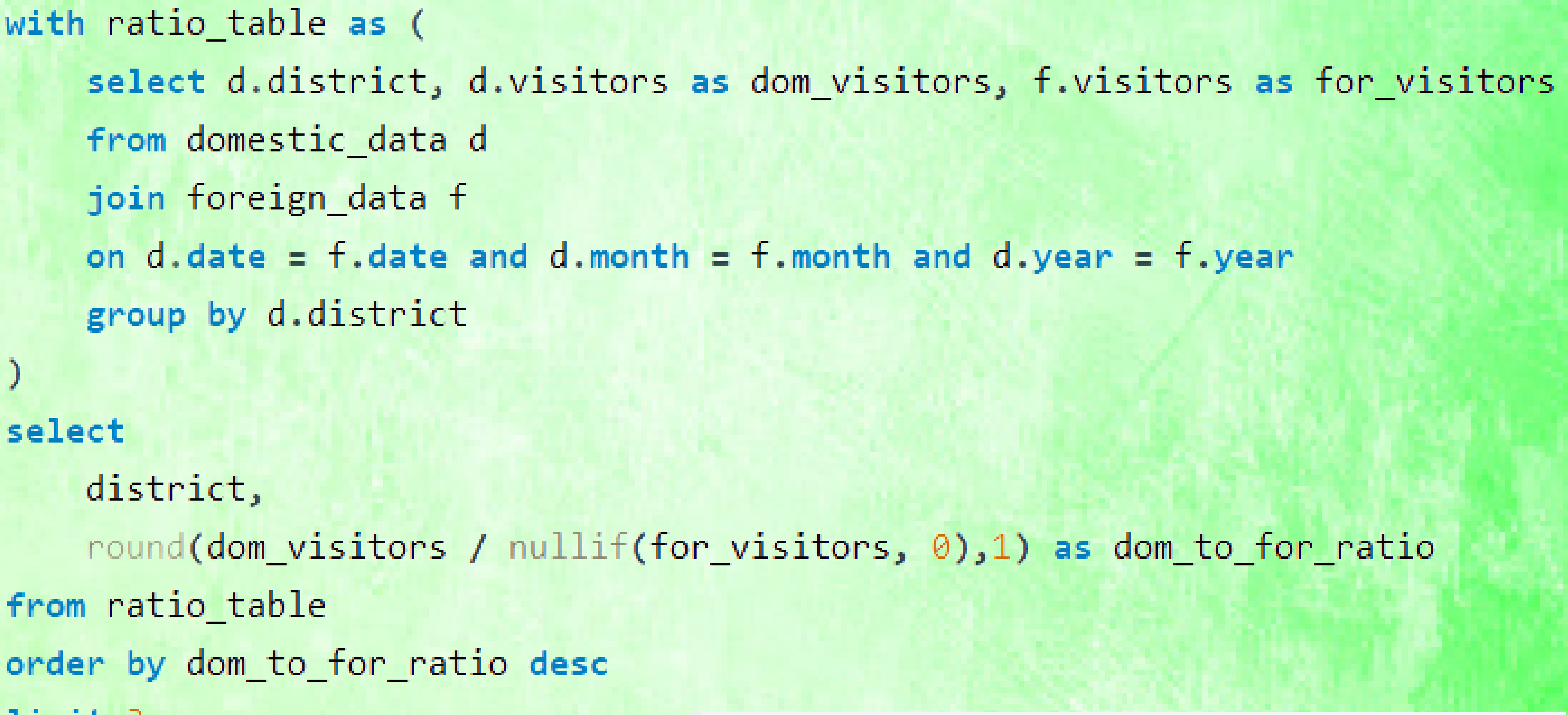
🟦 Medchal
🟧 Ranga Reddy
🟥 Suryapet

4.Peak and Low Season months for Hyderabad

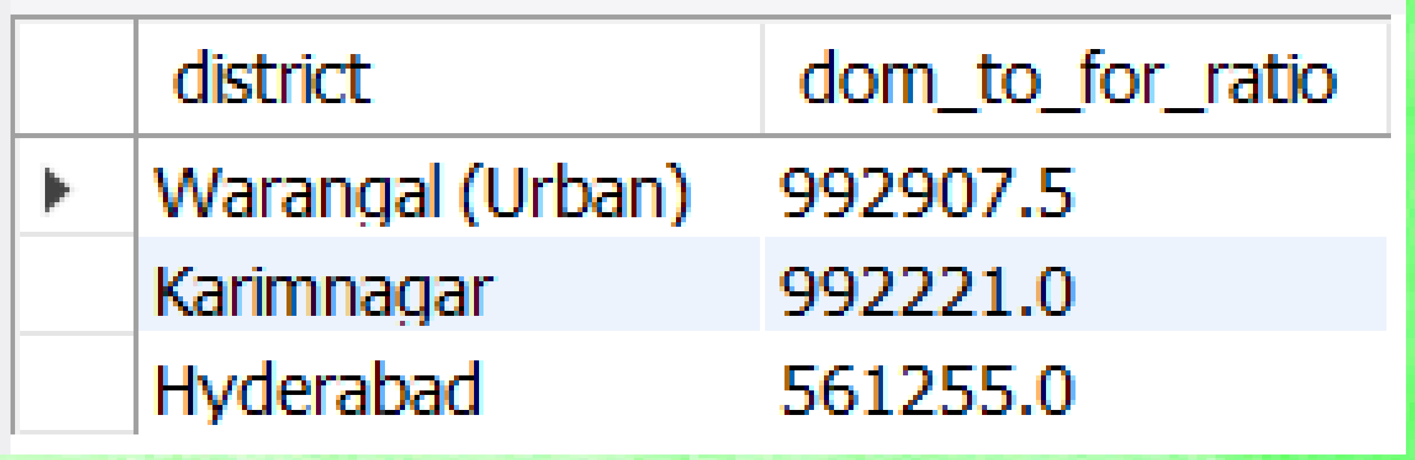


5. High Domestic to Foreign Ratio

Top 3



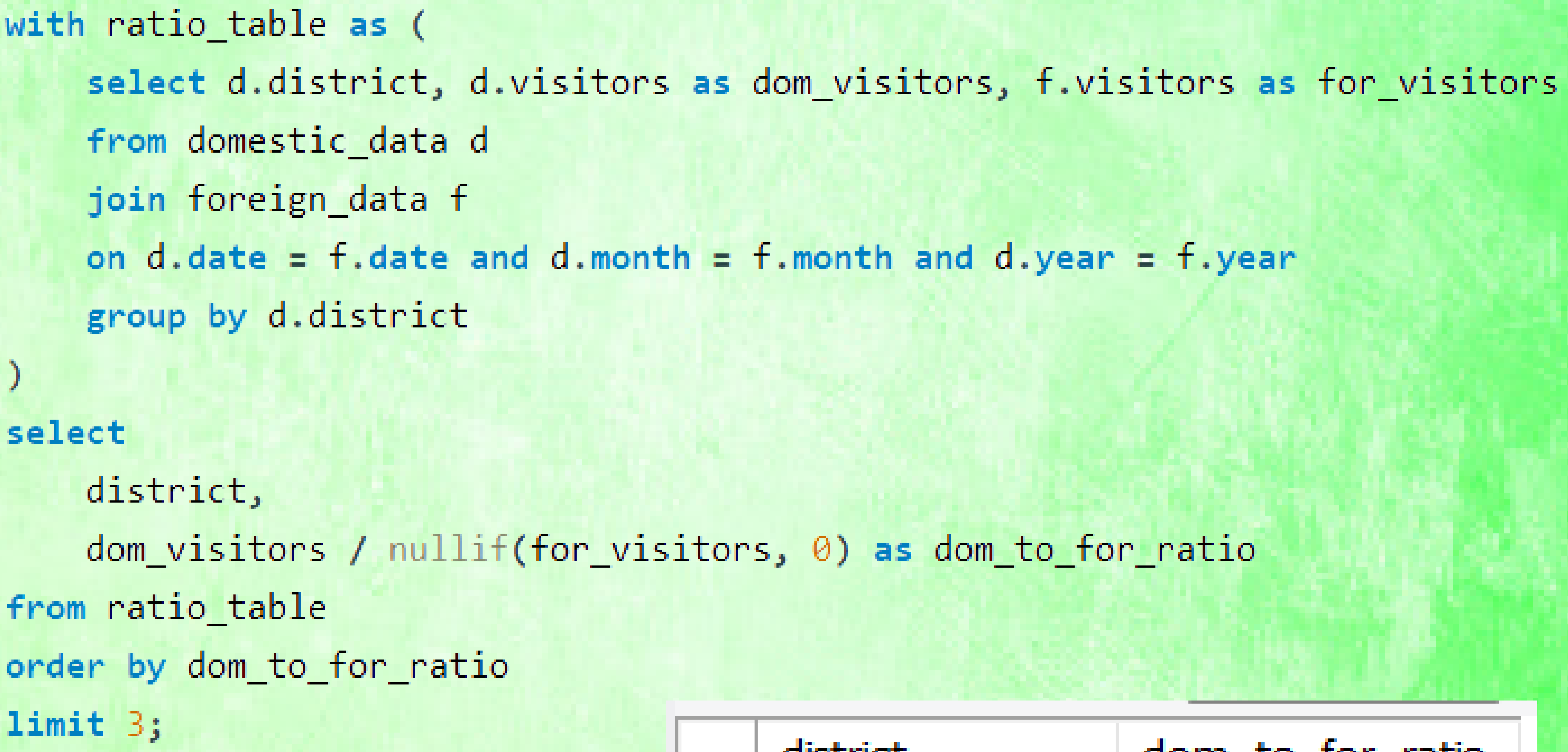
```
with ratio_table as (
    select d.district, d.visitors as dom_visitors, f.visitors as for_visitors
    from domestic_data d
    join foreign_data f
    on d.date = f.date and d.month = f.month and d.year = f.year
    group by d.district
)
select
    district,
    round(dom_visitors / nullif(for_visitors, 0),1) as dom_to_for_ratio
from ratio_table
order by dom_to_for_ratio desc
limit 3;
```



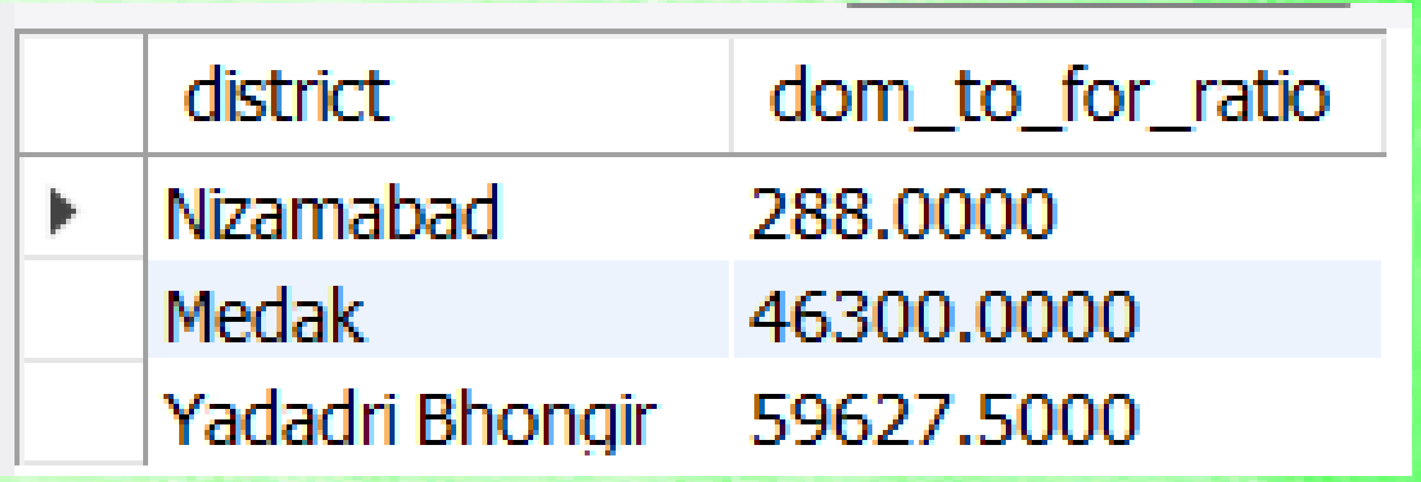
district	dom_to_for_ratio
Warangal (Urban)	992907.5
Karimnagar	992221.0
Hyderabad	561255.0

High Domestic to Foreign Ratio

Bottom 3



```
with ratio_table as (
    select d.district, d.visitors as dom_visitors, f.visitors as for_visitors
    from domestic_data d
    join foreign_data f
    on d.date = f.date and d.month = f.month and d.year = f.year
    group by d.district
)
select
    district,
    dom_visitors / nullif(for_visitors, 0) as dom_to_for_ratio
from ratio_table
order by dom_to_for_ratio
limit 3;
```

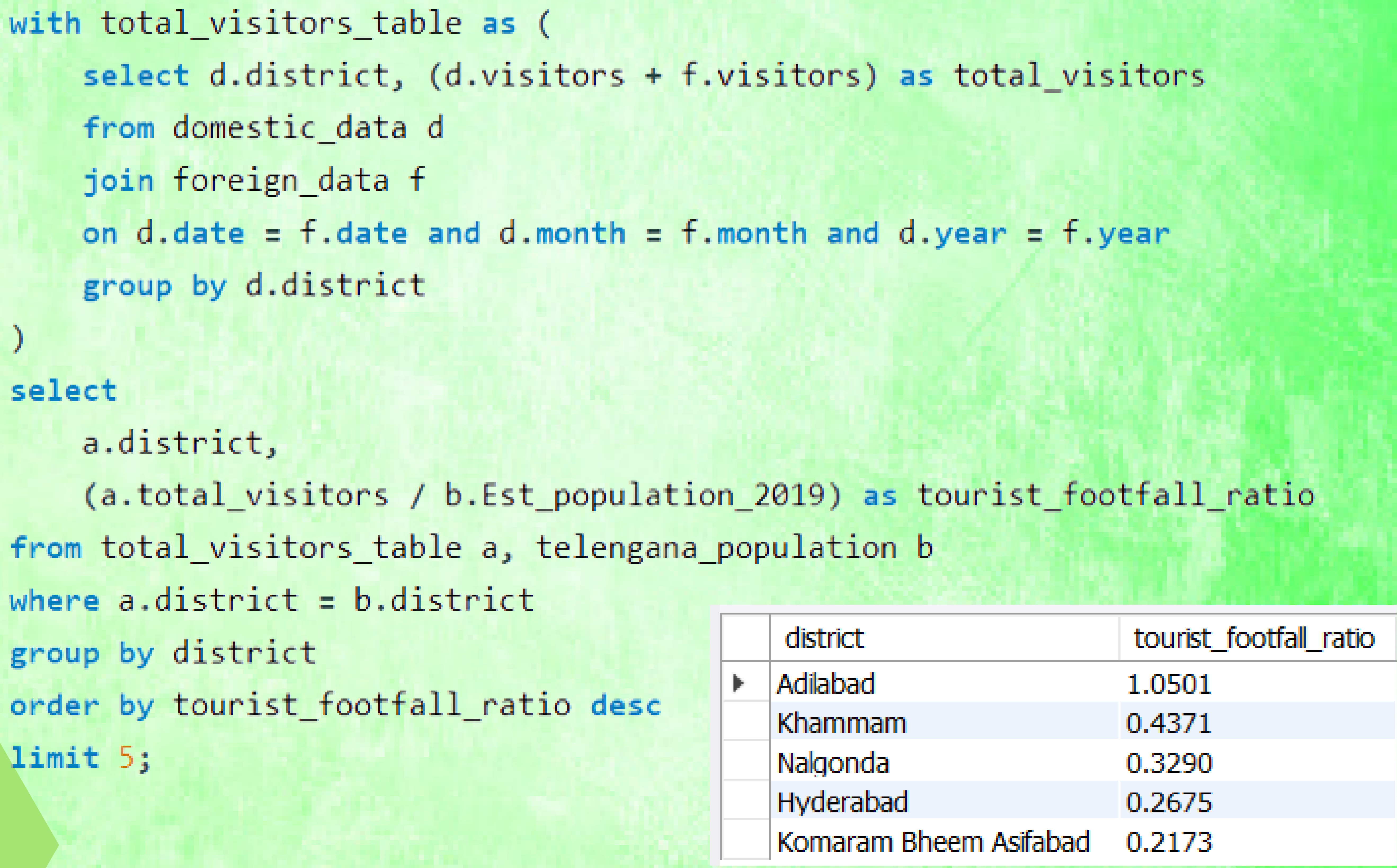


district	dom_to_for_ratio
Nizamabad	288.0000
Medak	46300.0000
Yadadri Bhongir	59627.5000

Secondary Research Questions

6. Population to Tourist Footfall Ratio

Top 5



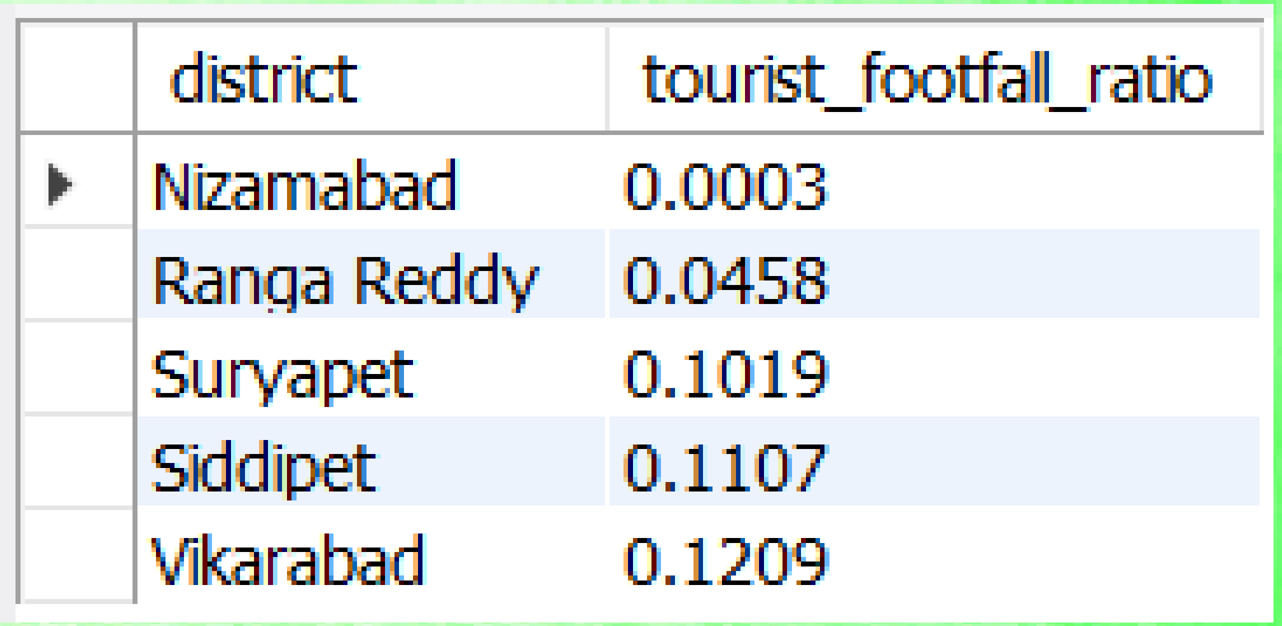
```
with total_visitors_table as (
    select d.district, (d.visitors + f.visitors) as total_visitors
    from domestic_data d
    join foreign_data f
    on d.date = f.date and d.month = f.month and d.year = f.year
    group by d.district
)
select
    a.district,
    (a.total_visitors / b.Est_population_2019) as tourist_footfall_ratio
from total_visitors_table a, telengana_population b
where a.district = b.district
group by district
order by tourist_footfall_ratio desc
limit 5;
```

district	tourist_footfall_ratio
Adilabad	1.0501
Khammam	0.4371
Nalgonda	0.3290
Hyderabad	0.2675
Komaram Bheem Asifabad	0.2173

Population to Tourist Footfall Ratio

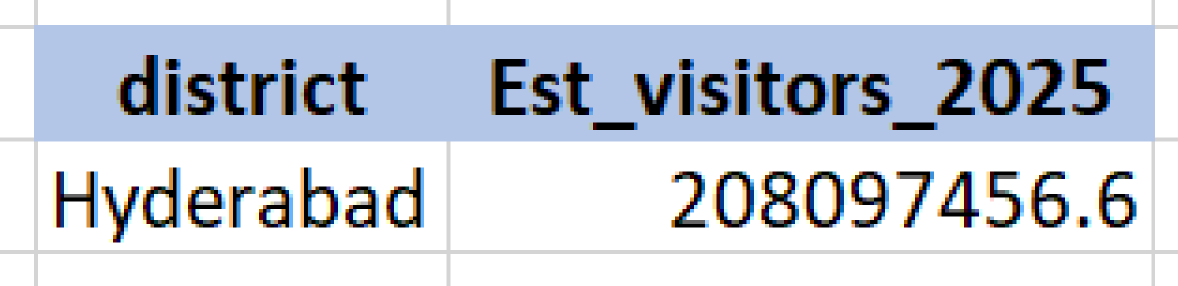
Bottom 5

```sql
with total_visitors_table as (
    select d.district, (d.visitors + f.visitors) as total_visitors
    from domestic_data d
    join foreign_data f
    on d.date = f.date and d.month = f.month and d.year = f.year
    group by d.district
)
select
    a.district,
    (a.total_visitors / b.Est_population_2019) as tourist_footfall_ratio
from total_visitors_table a, telengana_population b
where a.district = b.district
group by district
order by tourist_footfall_ratio
limit 5;
```

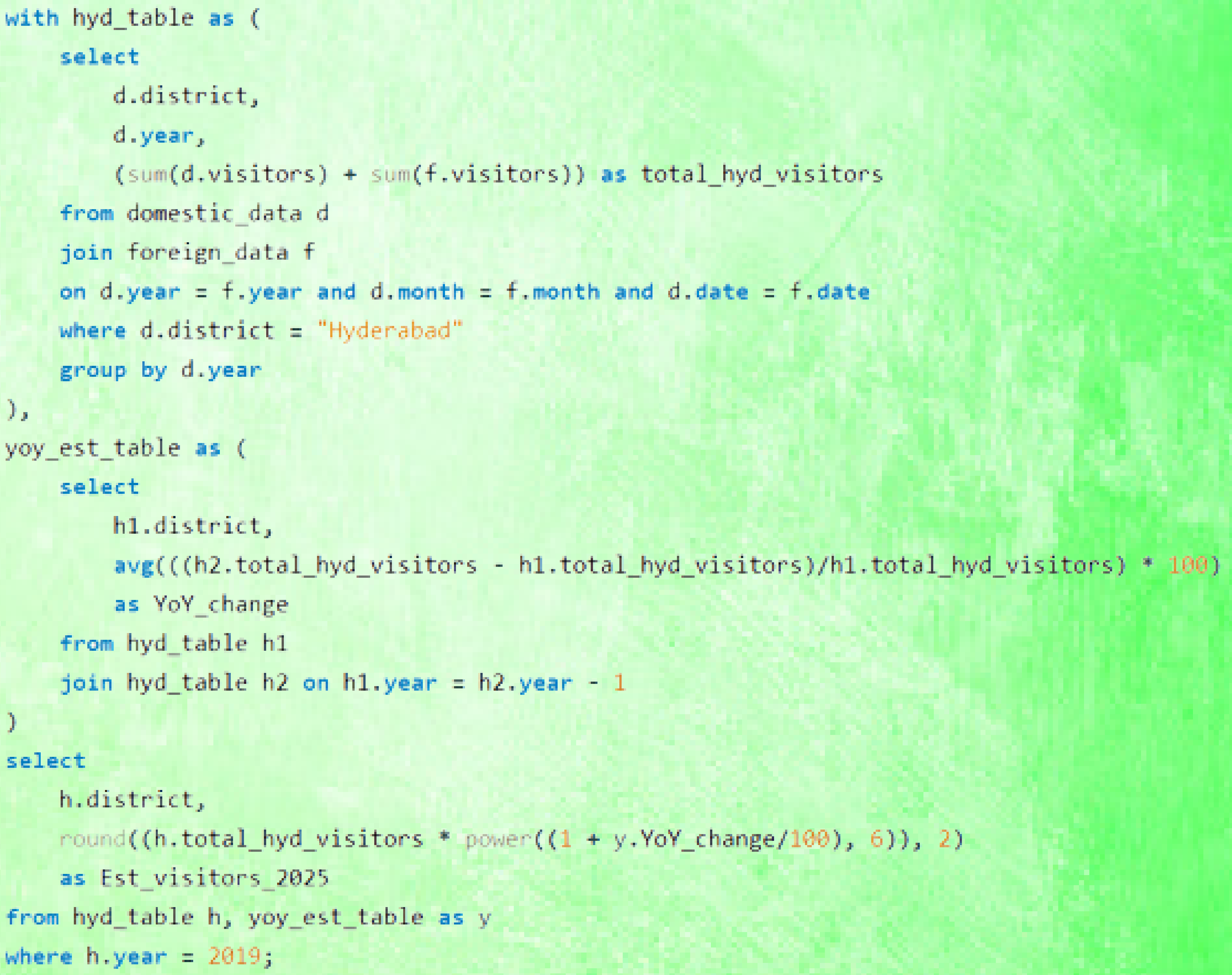


district	tourist_footfall_ratio
Nizamabad	0.0003
Ranga Reddy	0.0458
Suryapet	0.1019
Siddipet	0.1107
Vikarabad	0.1209

7. Projected Number of Domestic and Foreign in Hyderabad in 2025 based on growth rates from previous years.

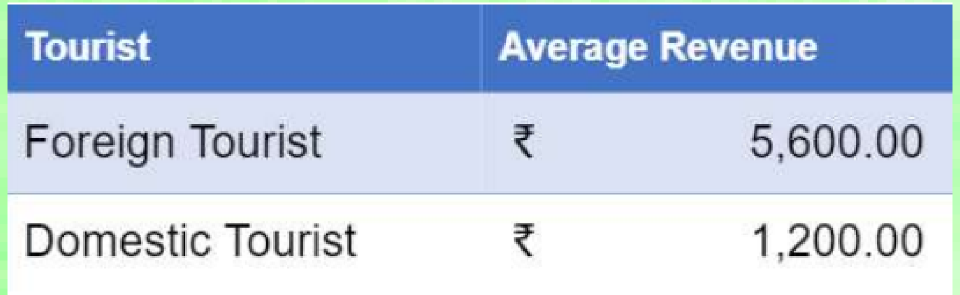


district	Est_visitors_2025
Hyderabad	208097456.6

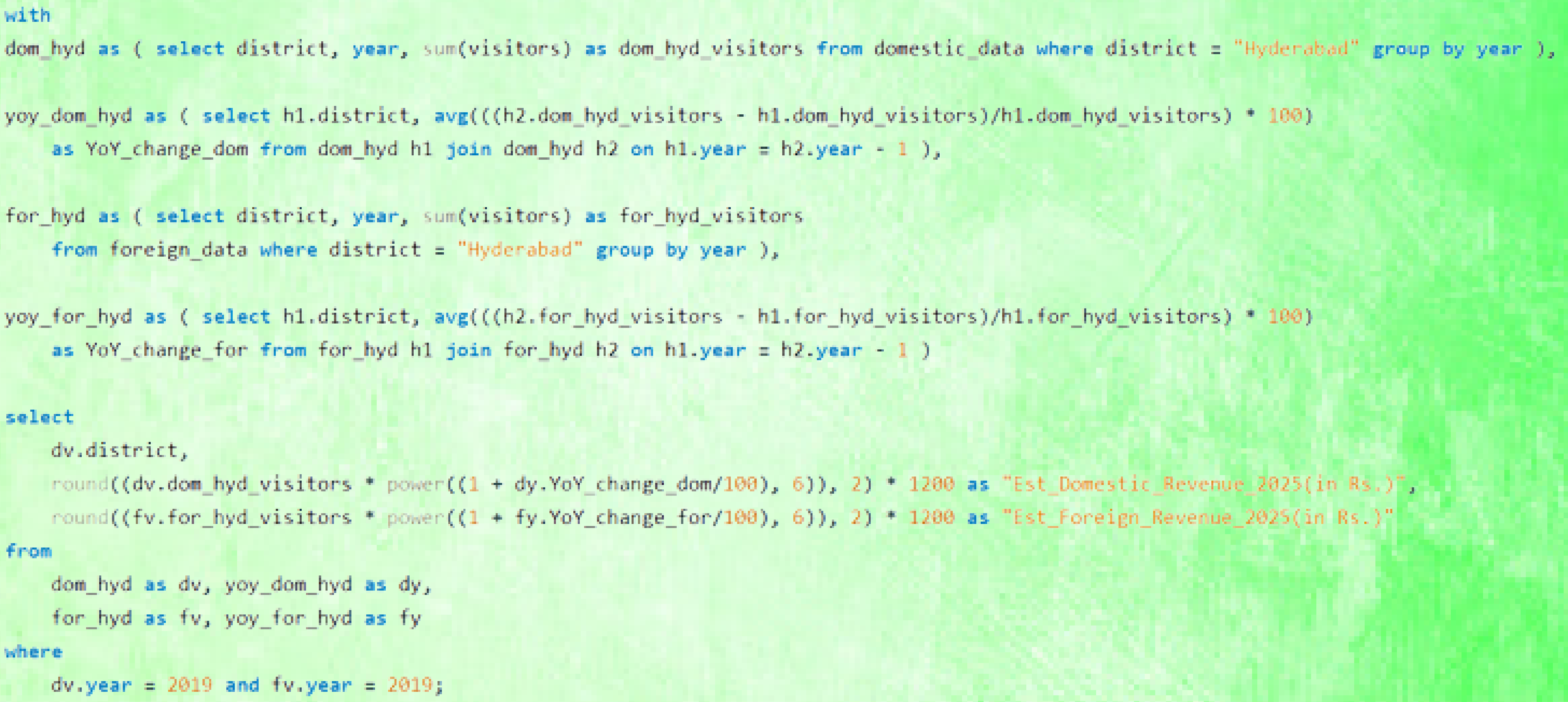


```
with hyd_table as (
    select
        d.district,
        d.year,
        (sum(d.visitors) + sum(f.visitors)) as total_hyd_visitors
    from domestic_data d
    join foreign_data f
    on d.year = f.year and d.month = f.month and d.date = f.date
    where d.district = "Hyderabad"
    group by d.year
),
yoy_est_table as (
    select
        h1.district,
        avg(((h2.total_hyd_visitors - h1.total_hyd_visitors)/h1.total_hyd_visitors) * 100)
        as YoY_change
    from hyd_table h1
    join hyd_table h2 on h1.year = h2.year - 1
)
select
    h.district,
    round((h.total_hyd_visitors * power((1 + y.YoY_change/100), 6)), 2)
    as Est_visitors_2025
from hyd_table h, yoy_est_table as y
where h.year = 2019;
```

8. Estimated Revenue for Hyderabad in 2025 based on average spend per tourist



Tourist	Average Revenue
Foreign Tourist	₹ 5,600.00
Domestic Tourist	₹ 1,200.00



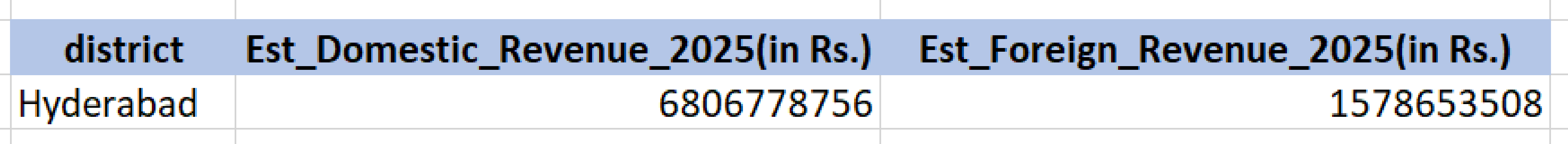
```
with
dom_hyd as ( select district, year, sum(visitors) as dom_hyd_visitors from domestic_data where district = "Hyderabad" group by year ),

yoy_dom_hyd as ( select h1.district, avg(((h2.dom_hyd_visitors - h1.dom_hyd_visitors)/h1.dom_hyd_visitors) * 100)
    as YoY_change_dom from dom_hyd h1 join dom_hyd h2 on h1.year = h2.year - 1 ),

for_hyd as ( select district, year, sum(visitors) as for_hyd_visitors
    from foreign_data where district = "Hyderabad" group by year ),

yoy_for_hyd as ( select h1.district, avg(((h2.for_hyd_visitors - h1.for_hyd_visitors)/h1.for_hyd_visitors) * 100)
    as YoY_change_for from for_hyd h1 join for_hyd h2 on h1.year = h2.year - 1 )

select
    dv.district,
    round((dv.dom_hyd_visitors * power((1 + dy.YoY_change_dom/100), 6)), 2) * 1200 as "Est_Domestic_Revenue_2025(in Rs.)",
    round((fv.for_hyd_visitors * power((1 + fy.YoY_change_for/100), 6)), 2) * 1200 as "Est_Foreign_Revenue_2025(in Rs.)"
from
    dom_hyd as dv, yoy_dom_hyd as dy,
    for_hyd as fv, yoy_for_hyd as fy
where
    dv.year = 2019 and fv.year = 2019;
```



district	Est_Domestic_Revenue_2025(in Rs.)	Est_Foreign_Revenue_2025(in Rs.)
Hyderabad	6806778756	1578653508

Conclusion

CODE BASICS

This project demonstrates the power and versatility of using Python, SQL, and Tableau together to analyze and visualize data.

Python was used for data preprocessing and cleaning,
SQL for querying and aggregating data and
Tableau for creating interactive visualizations and dashboards.

By integrating these tools, we were able to gain valuable insights into the data and effectively communicate our findings. This project showcases the potential of using a data-driven approach to solve real-world problems and make informed decisions.