A Project On

Plugging Into The Future: An Exploration Of Electricity Consumption Patterns

Submitted in partial fulfilment of Requirements for the award of the degree of

Bachelor of Technology

In

Computer Science and Engineering

By

K .Chandhana Priya

K .Hemanth Kumar

D S .Janani

J .Vimal

KUPPAM ENGINEERING COLLEGE

(Approved by AICTE & Affiliated to JNTU-A)

K.E.S Nagar, Kuppam-517425

ACKNOWLEDGEMENT

It is our privilege and pleasure to express a profound sense of respect, gratitude and indebtedness to our faculty mentor guide Mr V. Shekar Professor & Head, Dept. of Electrical and Electronnics and Engineering, Kuppam Engineering College, for his/her indefatigable inspiration, guidance,cogent discussion, constructive criticisms and encouragement throughout this dissertation work.

We express our sincere gratitude to K.Logesh , Professor & Head,
Department of Computer Science and Engineering, Kupppam Engineering College, for his suggestions, motivations and co-operation for the successful completion of the work.

We extend our sincere thanks to Dr .S.Sudhakar Babu , Principal ,Kuppam Engineering College ,for his encouragement and constant help.

DECLARATION

Project title: Plugging into the Future: An Exploration of Electricity Consumption Patterns

We K.Chandana priya, K.Hemanth kumar, D s .Janani, J.Vimal certify that this project is my own work, based on my personal study and/or research and that we have acknowledged all material and sources used in its preparation, whether they be books, articles, reports, lecture notes, and any other kind of document, electronic or personal communication.

we also certify that this project has not previously been submitted for assessment in any academic capacity, and that I have not copied in part or whole or otherwise plagiarised the work of other persons.

Team Members:

K.Chandana priya (Team Leader)

K.Hemanth kumar

D S . Janani

J.Vimal

ABSTRACT

India is the world's third-largest producer and third-largest consumer of electricity. The national electric grid in India has an installed capacity of 370.106 GW as of 31 March 2020. Renewable power plants, which also include large hydroelectric plants, constitute 35.86% of India's total installed capacity.

During the fiscal year (FY) 2019–20, the total electricity generation in the country was 1,598 TWh, of which 1,383.5 TWh generated by utilities.
The gross electricity consumption per capita in FY2019 was 1,208 kWh.

In 2015-16, electric energy consumption in agriculture was recorded as being the highest (17.89%) worldwide. The per capita electricity consumption is low compared to most other countries despite India having a low electricity tariff.

In light of the recent COVID-19 situation, when everyone has been under lockdown for the months of March to June the impacts of the lockdown on economic activities have been faced by every sector in a positive or a negative way.

The dataset is exhaustive in its demonstration of energy consumption state wise. Analysing Electricity Consumption in India from Jan 2019 till 5th December 2020. This dataset contains a record of Electricity consumption in each states of India, here we are going to analyse State wise , Region wise and Overall Electricity consumption in India.

CHAPTER -1

DEFINE PROBLEM

→Specify the Business Problem:

The lockdown announced by the government to control the Covid-19 pandemic willl adversely impact electricity demand, cash flows for discoms and lead to payment delays for power generation and transmission companies, ICRA said on Monday.
The lockdown has resulted in a shutdown of the industrial and commercial establishments and stoppage of passenger railway services.

→Business Requirements:

India's power system has undergone massive transformation, thanks to significant reforms by the Government of India. These reforms led to the creation of a single national power grid, boosted access to electricity for its citizens and promoted the dynamic growth of renewable energy. Renewables (including large hydropower) made up more than a third of today's 370 GW of installed capacity and 25% of electricity generation in 2019. After years of scarcity, the system has been in surplus in recent years thanks to investment in generation, and the private sector now accounts for 50% of total capacity. But some of the toughest, but most-needed, reforms remain to be done.

→Literature Survey:

COVID-19 has an especially adverse impact on the energy sector directly and indirectly. A study on crises and opportunities in terms of energy and artificial intelligence (AI) technologies during the COVID-19 pandemic for India found that the decline in electricity demand has directly resulted in a decrease in the use of coal by power plants. India's energy demand fell 26% within 10 days after imposing lockdown. This drop is much larger than the average global energy demand reduction of 6%. Another study observed that before starting

the lockdown, the daily energy consumption across the country attained a greater magnitude of around 3500 GWh. Then, its consumption started to fall and reached near 2500 GWh on 1 April 2020.

→Social Impact:

In India, the most COVID-19 affected sectors include aviation, tourism, hospitality, retail, manufacturing and automotive. The impact on the power sector is relatively less pronounced. The following table summarizes the market environment, potential impact and key response for the power sector in India.

CHAPTER-2

DATA COLLECTION AND EXTRACTION FORM DATABASE

→Collect the Dataset:

There are various techniques for primary data collection, including:

a. Surveys and Questionnaire

b. Interviews

c. Observations

d. Experiments

e. Focus Groups.

f. Published Sources

g. Online Databases

Data collection tools:

Word Association

- Sentence Completion
- Role-PlayingIn-Person SurveysOnline
- /Web Surveys
- Mobile Surveys
- Phone Surveys

Ex:we have downloaded dataset consumption of electricity in india at the year 2019 to2020

→**Storing Data in DB**: Data Storage system can be explained as the capacity secured by the database management system in the memory of the server allocated for the database and the related operations.

There is also a term known as 'Database Quota' used for indicating the data storage units, which can be defined as a fraction allocated for each database system from the total limit set for the complete database management system. In a relational database type, the data storage is usually in the form of tables, columns, rows, and their corresponding relationships. In other words, it is normally a structured system arranged in a way to allocate the required space for the contents and operations of the database.

→**Perform SQL Operations**:

In computer programming, create, read, update, and delete (CRUD) are the four basic functions of persistent storage. Alternate words are sometimes used when defining the four basic functions of CRUD, such as retrieve instead of read, modify instead of update, or destroy instead of delete.

CRUD is also sometimes used to describe user interface conventions that facilitate viewing, searching, and changing information; often using computer-based forms and reports. The term was likely first popularized by James Martin in his 1983 book managing the Data-base Environment. The acronym may be extended to CRUDL to cover listing of large data sets which bring additional complexity such as pagination when the data sets are too large to hold easily in memory.

Connect db with Tableau: Before you can build a view and analyze your data, you must first connect Tableau to your data. Tableau supports connecting to a wide variety of data, stored in a variety of places. For example, your data might be stored on your computer in a spreadsheet or a text file, or in a big data, relational, or cube (multidimensional) database on a server in

your enterprise. Or, you might connect to public domain data available on the web such as U.S. Census Bureau information, or to a cloud database source, such as Google Analytics,

Tableau Desktop

When you launch Tableau Desktop, the data connectors that are available to you are listed on the **Connect** pane, which is the left pane on the **Start** page. Under **Search for Data**, select **Tableau Server** to find data using Tableau Server or Tableau Cloud. File types are listed next, then common server types, or servers that you've recently connected to. Click **More** to see the complete list of data connectors you can use.

For supported files and databases, Tableau provides native connectors that are built for and optimized for those types of data. If your file or database type is listed under **Connect**, use this native connector to connect to your data. If your file or database type isn't listed, you might have the option of creating your own connection using **Other Databases (JDBC)**, **Other Databases (ODBC)**, a **Web Data Connector**, or a **Connector Plugin** built using the Tableau Connector SDK. Tableau provides limited support for connections that you create using these options.

The data connectors supported by your copy of Tableau Desktop are determined by the version you purchased. For more information, see the list of data connectors(Link opens in a new window) on the Tableau website. After you've connected to data, you can save the connections to have them show up under the **Saved data sources** section on the **Connect** pane.

You supply different information for each data connection that you want to make. For example, for most data connections, you'll need to supply a server name and your sign-in information. With some data connections, you can Run Initial SQL statements, and SSL-enabled servers require that you select the **Require SSL** check box when you connect. The following sections discuss the specific information you must provide for each type of data you want to connect.

CHAPTER-3

DATA PREPARATION

Abraham Lincoln might easily have been discussing data preparation steps for analytics when he said, “If I had 8 hours to chop down a tree, I would spend 6 sharpening my axe.” Spending 75% of the allotted time on preparation may seem like a lot. But in fact, most industry observers report that data preparation steps for business analysis or machine learning consume 70 to 80% of the time spent by data scientists and analysts.

The data preparation consists of the following steps:

- Access the data.
- Ingest (or fetch) the data.
- Cleanse the data.
- Format the data.
- Combine the data.
- And finally, analyze the data.

Access:

There are many sources of business data within any organization. Examples include endpoint data, customer data, marketing data, and all their associated repositories. This first essential data preparation step involves identifying the necessary data and its repositories. This is not simply identifying all possible data sources and repositories, but identifying all that are applicable to the desired analysis. This means that there must first be a plan that includes the specific questions to be answered by the data analysis.

Ingest:

Once the data is identified, it needs to be brought into the analysis tools. The data will likely be some combination of structured and semi-structured data in different types of repositories. Importing it all into a common repository is necessary for the subsequent steps in the pipeline. Access and ingest tend to be manual processes with significant variations in exactly what needs to be done. Both data preparation steps require a combination of business and IT expertise and are therefore best done by a small team. This step is also the first opportunity for data validation.

Cleanse:

Cleansing the data ensures that the data set can provide valid answers when the data is analyzed. This step could be done manually for small data sets but requires automation for most realistically sized data sets. There are software tools available for this processing. If custom processing is needed, many data engineers rely on applications coded in Python. There are many different problems possible with the ingested data. There could be missing values, out-of-range values, nulls, and whitespaces that obfuscate values, as well as outlier values that could skew analysis results. Outliers are particularly challenging when they are the result of combining two or more variables in the data set. Data engineers need to plan carefully for how they are going to cleanse their data.

Format:

Once the data set has been cleansed; it needs to be formatted. This step includes resolving issues like multiple date formats in the data or inconsistent abbreviations. It is also possible that some data variables are not needed for the analysis and should therefore be deleted from the analysis data set. This is another data preparation step that will benefit from automation. Cleansing and formatting steps should be saved into a repeatable recipe data scientists or engineers can apply to similar data sets in the future. For example, a monthly analysis of sales and support data would likely have the same sources that need the same cleansing and formatting steps each month.

Combine:

When the data set has been cleansed and formatted, it may be transformed by merging, splitting, or joining the input sets. Once the combining step is complete, the data is ready to be moved to the data warehouse staging area. Once data is loaded into the staging area, there is a second opportunity for validation.

Analyze:

Once the analysis has begun, changes to the data set should only be made with careful consideration. During analysis, algorithms are often adjusted and compared to other results. Changes to the data can skew analysis results and make it impossible to determine whether the different results are caused by changes to the data or the algorithms.

CHAPTER-4

DATA VISUALIZATIONS

Data visualization is the process of graphical representation of data in the form of geographic maps, charts, sparklines, infographics, heat maps, or statistical graphs. Data presented through visual elements is easy to understand and analyze, enabling the effective extraction of actionable insights from the data.

Data Visualization tools:

Some of the best data visualization tools include Google Charts, Tableau, Grafana, Chartist, FusionCharts, Datawrapper, Infogram, and ChartBlocks etc. These tools support a variety of visual styles, be simple and easy to use, and be capable of handling a large volume of data.Data is becoming increasingly important every day. For any organisation, you can understand how important data is while making crucial decisions. For the same reason, data visualisation is grabbing people's attention. Modern data visualisation tools and advanced software are on the market. A data visualisation tool is a software that is used to visualise data. The features of each tool vary, but at their most basic, they allow you to input a dataset and graphically alter it. Most, but not all, come with pre-built templates for creating simple visualisations.

Data visualisation tools used in our project :

➔ **Tableau:**

One of the most widely used data visualization tools, Tableau, offers interactive visualization solutions to more than 57,000 companies.

Providing integration for advanced databases, including Teradata, SAP, My SQL, Amazon AWS, and Hadoop, Tableau efficiently creates visualizations and graphics from large, constantly-evolving datasets used for artificial intelligence, machine learning, and Big Data applications.

The Pros of Tableau:

- Excellent visualization capabilities
- Easy to use

The Cons of Tableau:

- The pricing is a bit on the higher side
- Auto-refresh and report scheduling options are not available

Tableau Installation – Steps to Download & Install:

Here are the steps of the Tableau installation process in detail:
Step 1: To download Tableau Desktop, first we need to open our browser and go to Tableau's official website.

Once the website opens, click on the **TRY NOW** button to initiate the download process.

Step 2: This will take us to the download page for Tableau Desktop.

When we scroll down on this page, we will get a list of system requirements to download Tableau Desktop. Here, they have provided platform wise requirements, that is, for Windows, Mac, etc. Make sure to check the requirements for your platform before starting the download.
Step 3: At the top of this page, it will ask you to enter a business E-mail. Please note that you must not enter your personal E-mail ID here. Enter the appropriate E-mail address and click on **DOWNLOAD FREE TRIAL**.

Step 4: As soon as you will click on the download button, a .exe file will start downloading automatically. You will find a file download icon at the left-most bottom of your screen. It will take around 4 to 5 minutes until the download completes.

Once the download gets completed, open the exe file.

In this way, our Tableau Desktop download is complete. Next, we will move on to installing Tableau Desktop in our system.

Installing Tableau on desktop:

In this section, we will see how to install Tableau Desktop through the installation wizard.
Step 1: As soon as you open the downloaded exe file, the install wizard will launch. Click on **Run** to start the installation process.

You will get the first window as the welcome window. Check the box for license terms agreement and click on **Install**. This will start the install process.

Step 2: The next step will take a few minutes to install important files. The progress bar will indicate the progress in green.

Step 3: The next three steps are regarding the activation process. In this window, you have to choose from two options; **Start trial now** and **Activate**. The **Start trial now** option is the free 14-day trial option whereas, the **Activate** option is for installing the paid version in which you need to enter a product key.

We went for the free trial version as our purpose of the use is learning.
Step 4: Moving forward, it will ask you to fill up a registration form. It contains fields like Name, E-mail, Organization, Job title, etc. Fill the right details and click on **Register**.

The next window will confirm successful registration.

Step 5: Once the activation is complete, go to your desktop and click on the **Tableau** icon to open Tableau.
Step 6: Tableau Desktop welcome page will open. On its left, you can find a range of ***data source*** options you can connect to. And on its right panel, you can explore the training and resources option. To open a new workbook, click on the **Tableau star-like** icon given on the upper left corner.

Different types of charts ploted in our project are:

- Bar Charts
- Line Charts
- Pareto Charts
- Area Charts
- Histograms
- Pie Charts
- Tree Maps
- Scatter Plots
- Bubble Charts
- Heat Maps
- Maps
- Bullet Charts
- Gantt Charts
- Box and Whisker Plots
- Waterfall Charts
- Motion Charts

→**Bar charts** are definitely one of the most, if not the most common data visualizations across all BI platforms. You can quickly highlight differences between categories, show trends and outliers, and reveal historical highs and lows at a glance.

→**line chart**, or line graph, is another familiar method for displaying data. It connects several distinct data points, presenting them as one continuous evolution.

→**A Pareto chart** consists of both bar and line graph and the same measure is used to create the graphs but the *Measure* shelf values are manipulated differently.

→**Area charts** represent any quantitative data over various periods of time. It is basically a line graph where the area between line and axis is generally filled with color.

→**Histograms** show how your data is distributed across distinct groups. By grouping your data into specific categories (also known as “bins”), then plotting the number of records in a category as a vertical bar, you can quickly see which bins the majority of your data falls in.

→**Pie charts** are powerful for adding detail to other visualizations. The angle of the pie determines the measured value.

→**Treemaps** relate different segments of your data to the whole. By nesting rectangles within others, treemaps show how individual data points fit in a hierarchy.

→**Heat maps** are a great way to compare data across two or more categories using color. Patterns guide viewers around the chart, quickly showing them where the intersection of categories is strongest and weakest

CHAPTER-5

DASHBOARD

. A dashboard for data analytics is a tool used to multi-task, organize, visualize, analyze, and track data. The overall purpose of a data analytics dashboard is to make it easier for data analysts, decision makers, and average users to understand their data, gain deeper insights, and make better data-driven decisions.

Data dashboards are designed to connect and help extract important information from a wide variety of different data sources, services, and APIs. This information is displayed in a single, unified view via visuals such as charts, figures, graphs, and tables. An organization can have a different customizable dashboard for each department and even a dashboard for each individual project, which helps provide granular monitoring of very specific KPIs.

There are three types of dashboards: operational, strategic, and analytical.

- **Operational**:

  Operational analytics dashboards focus on the progress of work and show the current state of operations. Operational dashboards are real-time, are often used to understand where issues are, and are less often used for digging deeper into data.

- **Strategic:**

  Strategic analytics dashboards display KPIs and clear, at-a-glance, digestible data visualizations. The focus is on showing simple trends and progress, obtaining immediate answers, and helping users monitor and understand if KPIs are on track.

- **Analytical:**

  Analytical dashboards focus on flexibility and allowing the user to explore data points in as many ways as possible, through filters and comparisons, to reveal hidden insights. Common features include drill-downs, predictive models, and what-if parameters.

There are many different benefits to be gained from the many different kinds of data analytics dashboards. Some of the most common benefits include:

- data visibility
- accessibility
- measuring performance,
- business forecasting abilities
- agile responses

The best data analytics dashboard tools will offer:

- the ability to connect your data from multiple sources,
- embedding capabilities,
- self-service reporting,
- automated real-time updates,
- streaming and predictive analytics driven by AI,
- filtering across time and location, interactive visual analytics,
- full customization, and at-a-click exploration.

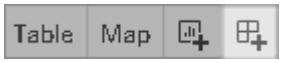
Some examples of popular enterprise analytics dashboard software include:

- HEAVY's visual analytics platform,
- Izenda, Periscope Data,
- Dundas BI analytics dashboard,
- Microsoft Power BI,
- IBM Cognos,
- TIBCO Spotfire, Looker, and
- Sisense.

Procedure for creating a dashboard using tableau desktop:

You create a dashboard in much the same way you create a new worksheet.

1. At the bottom of the workbook, click the **New Dashboard** icon:



2. From the **Sheets** list at left, drag views to your dashboard at right.

3. To replace a sheet, select it in the dashboard at right. In the Sheets list at left, hover over the replacement sheet, and click the **Swap Sheets** button.

or

Step 1: Install and launch the Tableau software

The first step to creating a dashboard is to download the Tableau software. You can do this from the Tableau website, selecting compatibility with your Mac or Windows operating system.

Follow the process and install the application on your device. Launch the software, and you're ready to begin.

Step 2: Connect to a data source

Before you can build a dashboard on Tableau, you'll need to connect it to a data source. First, launch Tableau and select the Connect to Data option. From there, choose the data source you want to connect.

Tableau supports various data sources, including file systems like Excel and CSV, relational databases like Oracle and SQL server, and various cloud systems.

Once you've selected your data source, you can move on to creating a worksheet.

Step 3: Create a new worksheet

Creating a new worksheet on Tableau takes very few steps. First, open Tableau and select the New Worksheet icon from the toolbar.

To rename the worksheet, right-click on the tab and select Rename.

Next, select the data source you want to use for your worksheet (you can follow along with our example by downloading this dataset). Once you've selected a data source, you'll need to choose which data fields you want to include in your worksheet.

Drag the desired fields from the data source pane onto the canvas. After you've chosen your data fields, you can begin creating your worksheet.

Click OK to close the dialog box and apply the changes. Your new worksheet will now be available in Tableau.

Step 4: Open a new dashboard

To create a new dashboard, click on the Dashboard button on the menu bar at the top of your screen. This will bring up a drop-down column. Select New Dashboard. You can rename it as you desire.

The new window comprises a blank workspace and a panel on the left side of the screen. The Dashboard pane on this panel has three sections.

- Size helps you set the dashboard dimensions and ensure it's responsive to mobile and desktop devices.
- Sheets allows you to add to or remove from the dashboard various visualization sheets.
- Objects are useful for adding images, texts, webpages, and extensions to the dashboard.

On the other hand, the Layout tab on the side panel helps provide more aesthetics in designing the dashboard. It allows you to adjust the border lines, padding, and background images.

Step 5: Configure the dashboard size

Select the Dashboard icon on the side pane. Under the Size section, click the drop-down button to select a dimension you'd like for your dashboard. Depending on your preference, you can choose a preset sizing option from the list or create a custom size by clicking the Custom button at the bottom of the drop-down menu.

Step 6: Add relevant sheets

Once you've set the dashboard size, you can include sheets. To do that, drag each worksheet and drop them on the canvas. The dashboard should now display the visualizations of the data present in that sheet.

You must organize the sheets to fit into the available work area. To do that, click on each graph or visual and select the Floating option on the drop-down menu at the right side of your screen. This allows you to move each visual around and organize the canvas.You can also enlarge or shrink the size of each visualization on your dashboard. Simply click on the desired image, click and hold any of the image vertices, and resize them to your liking.

Step 7: Add filters

A great dashboard should be interactive and give your reader more flexibility on how they view the data and explore the story behind the numbers—such as viewing a specific time range, region, or demography. Filters help you achieve this level of interactivity.

Add filters to a dashboard by following these steps:

1. Click on the sheet you'd like to add a filter to within your dashboard so the outline appears around it.
2. Select the small down arrow next to the visual and click on the Filter option.
3. Choose a filter you want to add.

Step 8: Add objects

You can add various buttons to the dashboard from the Objects section of the side panel and determine what they should do when you click on them. For example, you can include a website's company logo, project title, or screenshots.

Step 9: Change to presentation mode

Switching to presentation mode helps you interact with your data visualizations as your audience would. In presentation mode, you can hide the building tools and focus on the dashboard. That way, you can assess your data story, presentation layout, and filters to ensure everything is good before publishing.

To enter presentation mode, click the Presentation icon in the top-right corner of the Tableau interface

CHAPTER-6

STORY

what is a story in Tableau?

Well, it is a sequence of different charts that combine to provide a cohesive plot to its viewers. In essence, all these charts tell a story about the data which allows the viewers to form their conclusion. The story in Tableau contains story points, where each story point is either a worksheet or a dashboard

.

How to create a Story?

Let's see the various steps required to create a Story in Tableau. This story uses the

Step 1: Click on the new Story tab to create a new story. You can then add various sheets and dashboards to create a story point.

Step 2: You can double-click on the sheets and dashboards on the left to add them to a story point. You can also drag the sheets into your story point on the Tableau desktop. All the sheets and dashboards that are added to a story are connected to their original forms. So any changes made to the original sheets or dashboards are reflected in the story. For example, let's add a dashboard containing the relation between Discounted Sales and Profit by Category to the story.

Step 3: We can also add a caption to summarize the story point by clicking on "Add a caption" and then writing it. Let's add the caption "Relation between Discounted Sales and Profit by Category and Subcategory" to our example.

Step 4: It is possible to add another story point by 2 methods. You can either click on the Blank tab to use a blank sheet for the next story point or click on the Duplicate tab to obtain a duplicate sheet as the current story point. Let's click on the blank option.

Step 5: You can change the size of your story by clicking on the Size option in the lower-left corner. You can choose from one of the predefined sizes or set your custom size in pixels. You can also change the name of your story by right-clicking on your Story tab and choosing rename.

Step 6: Now, let's see a complete story on the relationship between the discounted sales and profit

→From a dashboard to story:

1. **Make a dashboard easier to read**

Many of the lessons taught in our books are specific to making explanatory communications better by using techniques like focusing and telling a story. Since dashboard visuals update when filters are applied and data is refreshed, it is difficult to add specific text or focus attention on a particular data point to tell a compelling story. However, some of the *storytelling with data* lessons do apply.

2. **Keep your audience in mind**

When creating a monitoring report capable of exploratory analysis, one should be thoughtful of the audience, and the insights they need to glean from the data. To improve the overall user experience, leverage white space, alignment, and grid layouts. This will make the view easier to read and navigate at all levels of exploration. For more tips on designing effective dashboards, check out The Big Book of Dashboards.

3. **Choose effective visuals**

To enable quick discovery, select the most appropriate chart type for the data being depicted. Some of the charts in our dashboard would be easier to interpret as a different visual. For example, a line graph would show the trends for registrations and sessions better than bars. A dot plot would enable a simpler view of trainer sessions, while stacked bars would provide a relative comparison of each fitness center location.

4. **Identify and eliminate clutter**

Because dashboards are naturally busy, we certainly want to take steps to reduce the cognitive burden for users by removing items that do not add information value, like gridlines and borders. The amount of color used can also contribute to a cluttered feeling.

In this simplified version of the dashboard, we've made it easier for our audience to see the data clearly by minimizing the number of data labels, borders, and colors.

Exploring data is different than explaining data

Dashboard interactivity makes data discovery much easier, but to drive meaningful change, it is usually more effective to craft a communication specific to the story we want to tell. Creating a presentation tailored to our audience lets us deliver our findings in a way that will resonate with them, and frees us from having to compete for attention with unrelated charts, filters, and text.

5. **Isolate data from your explorations that support your specific message**

In a presentation, we don't need to share every bit of data we looked at in dashboards throughout the exploratory process. The point of looking at all that data was so that we could isolate the critical insights, and share those directly with decision-makers. Providing too much data runs the risk of overwhelming our audience.

A more effective approach is to include only the most meaningful information needed to support our main message, or what we like to call the Big Idea. After crafting our Big Idea, we can use it to identify what data is necessary for our story—and what we can leave behind on the dashboard.

Imagine we oversee the personal training programs across all of the fitness center locations, and after reviewing the dashboard, we realize two important things.

1. **Registrations and sessions have tapered off since launching in May.** Our summer promotion was effective at getting clients to register for the program, but if we want to continue to grow the number of people we help through the program, we likely need another marketing push leading into the holiday season.
2. **Some centers are doing a better job** at getting registered clients to attend multiple sessions and there are likely some strategies we could learn from these locations.

As a result of these findings, we want to meet with the head of marketing to discuss how to drive more awareness and engagement for the program. Our Big Idea might be*:* To help more clients meet their fitness goals, we should develop a new marketing strategy for our personal training program and apply learnings from successful engagement tactics across all locations. Since we want to have a targeted conversation with the head of marketing, we do not want to share the entire dashboard, as it includes additional data that could distract from our key takeaways. Instead, we want to isolate just the pertinent information. Using our Big Idea as a filter to identify the data needed as evidence, we focus on the two graphs highlighted below.

Now that we've identified our audience, articulated our Big Idea, and found the data needed to support our message, we are well-positioned to create our communication to the head of marketing.

6. **Use words and color strategically**

Words and color are a powerful combination for highlighting key points in our communications. By taking the data out of the monitoring report, we can more easily get our audience to notice the most important information. Check out how we can immediately call attention to the lower number of weekly registrations by using color sparingly paired with words.

CHAPTER-7

PERFORMANCE TESTING

Our project has been generating a large amount of data in textual format. As the data is available on the web, a web-based visualization tool is desirable for the project rather than a standalone tool. This research analyzes the processing mechanisms of four popular web-based data visualization tools, that is, Google Charts, Flex, OFC, D3, and compares their performances.

A standalone visualization tool, JfreeChart, have been also used for comparison. The processing times have been divided into three segments, layout time, data transformation time, and rendering time, and separately measured. The actual temperature data from the Nevada Nexus project has been used for testing in different scales ranging from 100 to 100,000 data points. The result shows that each visualization tool has its own ideal environment.

Test our data visualization and dashboard for usability and performance

Once you have designed your data visualization and dashboard, you need to test them for usability and performance. Usability testing evaluates how easy it is for users to interact with and understand your visualization and dashboard, while performance testing measures how fast and reliable your visualization and dashboard are in terms of loading, rendering, and updating the data. You can use various methods and tools for testing, such as user feedback surveys or interviews, user testing in person or remotely, analytics tools like Google Analytics or Hotjar, and testing tools like Lighthouse or WebPageTest. User feedback can provide satisfaction, comprehension, preferences, and suggestions for improvement. User testing can record actions, reactions, comments, and questions while measuring time, accuracy, and errors. Analytics tools can measure page views, bounce rate, session duration, click-through rate, scroll depth, and heat maps. Testing tools can measure load time, speed index, render time, resource size, and performance score.

Optimize our data visualization and dashboard based on the test results

Based on the results of your testing, you can optimize your data visualization and dashboard for usability and performance. To do this, you may need to make changes to your design, data, or code. This can help improve clarity, engagement, accuracy, speed, functionality, and usability. For example, simplifying or modifying the visualization type or format can make it easier to read and understand. Adding or removing data elements can also help highlight important information or reduce clutter. Additionally, adjusting colors, fonts, symbols, and layout can make them more consistent and meaningful. Titles, captions, annotations, and tooltips can provide more context while compressing or resizing images and files can reduce loading time. Lastly, fixing errors or bugs will increase the reliability of your visualization and dashboard.

Calculate the number of fields:

A field is a column in a database table. A calculated field is a field that uses existing database fields and applies additional logic — it allows you to create new data from your existing data. A calculated field either:

- performs some calculation on database fields to create a value that is not directly stored in the database or
- selects values in database fields based on some customized criteria

CHAPTER-8
WEB INTEGRATION

Web integration goes beyond traditional web scraping to provide hidden insights to businesses and analysts that are not easily readable by human end users. Web data integration is the process of acquiring and transforming data from multiple websites into one cohesive workflow.

web integration process:

1. **Identify.** Before any data extraction can occur, you must identify what type and data sources might provide you with business insights—for example, web sources where information is located. Identify the URL where your data is located.
2. **Extract.** Once you have identified and targeted specific data sources, you can begin the web data extraction process or web scraping. Relatedly, prior to starting the web data integration process, you must decide whether or not to process your data in-house or choose to outsource this process.
3. **Prepare.** The preparation step encompasses a larger group of smaller data quality adjacent processes, such as cleansing, normalization, and standardization. This step is vital in ensuring that data quality is maintained throughout the remaining integration process.
4. **Integrate.** After the data has been normalized and a certain standard of data quality has been achieved, you can now integrate the data into APIs.
5. **Utilize.** Lastly, with fully integrated data, you can begin discovering powerful insights from your newly processed data. For example, companies are able to integrate this data for machine learning and other AI business processes.

How to web Integrate dashboard with Flask:

STEP-1: Create a dashboard using tableau desktop.

STEP-2:Flask installation ,code gerneration for embed the dashboard with website ,generation of IP address .

STEP-3:Import necessary libraries, initialize the flask app, and load our dashboard.

STEP-4: Define the app route for the default page of the web-app here route refers to URL.

STEP-5:Create a webpage using html ,css and javascript for embed the dashboard to it .

STEP-6:Run the code and flask server started and it generates the IP address .

S

TEP-7:Copy the generated ip address and paste in the web browser to see the embed dashboard in web page.

3 benefits of web integration:

1. Increased accessibility

The API capability and integration within the larger web data integration process provide quick connections and more accessibility. For example, structured and normalized datasets available through APIs can provide investors with up-to-date insights during funding periods and throughout the investment evaluation stage.

2. Enhanced insights

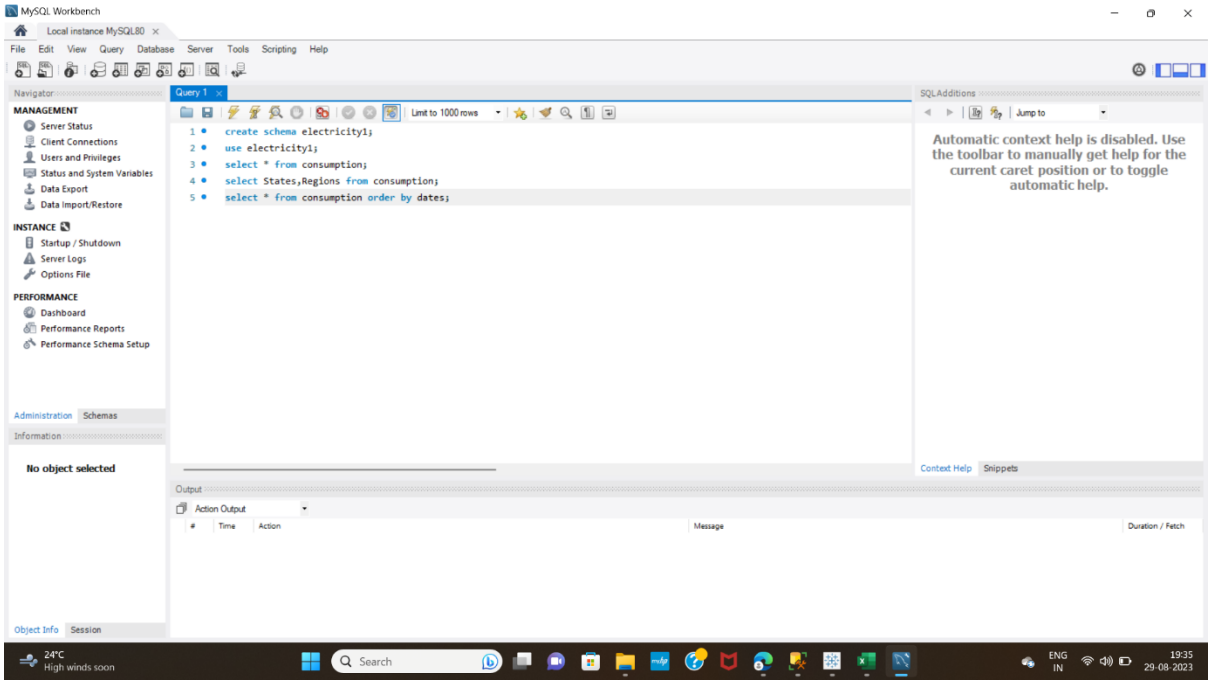
Manual web scraping can often miss data and therefore doesn't provide a full picture for analysis. The iterative and enhanced features of the web data integration process allow for retrieving hidden data in HTML files that aren't necessarily readable or accessible to human end users.

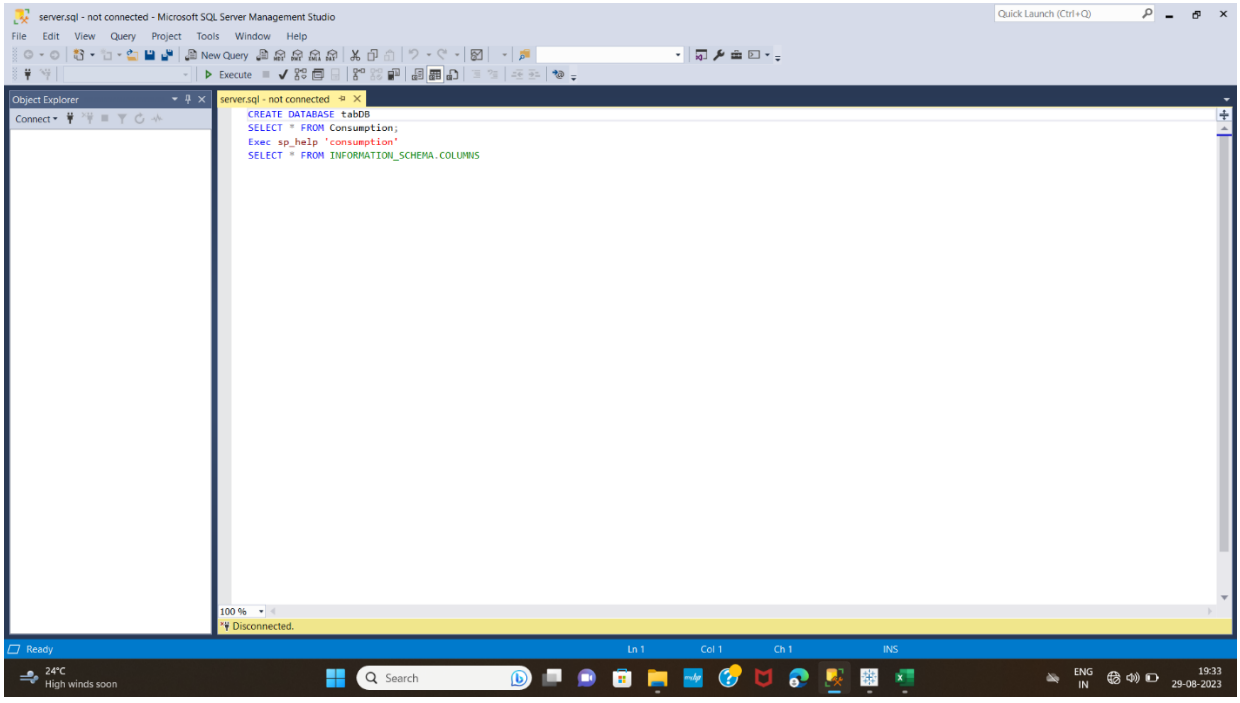
3. Improved data quality

The identifying and preparation stages of the web data integration process are centered around achieving data quality. For instance, data quality can be gained during the identify stage simply due to the targeted approach of selecting the appropriate sources for enhanced insights.

CONCLUSION:

We conclude that three major groups of determinants influence residential electricity consumption segmentation: physical characteristics of a dwelling especially year of construction and total floor area; electrical heating/cooling equipment and fireplaces ownership and use; and occupants profiles .





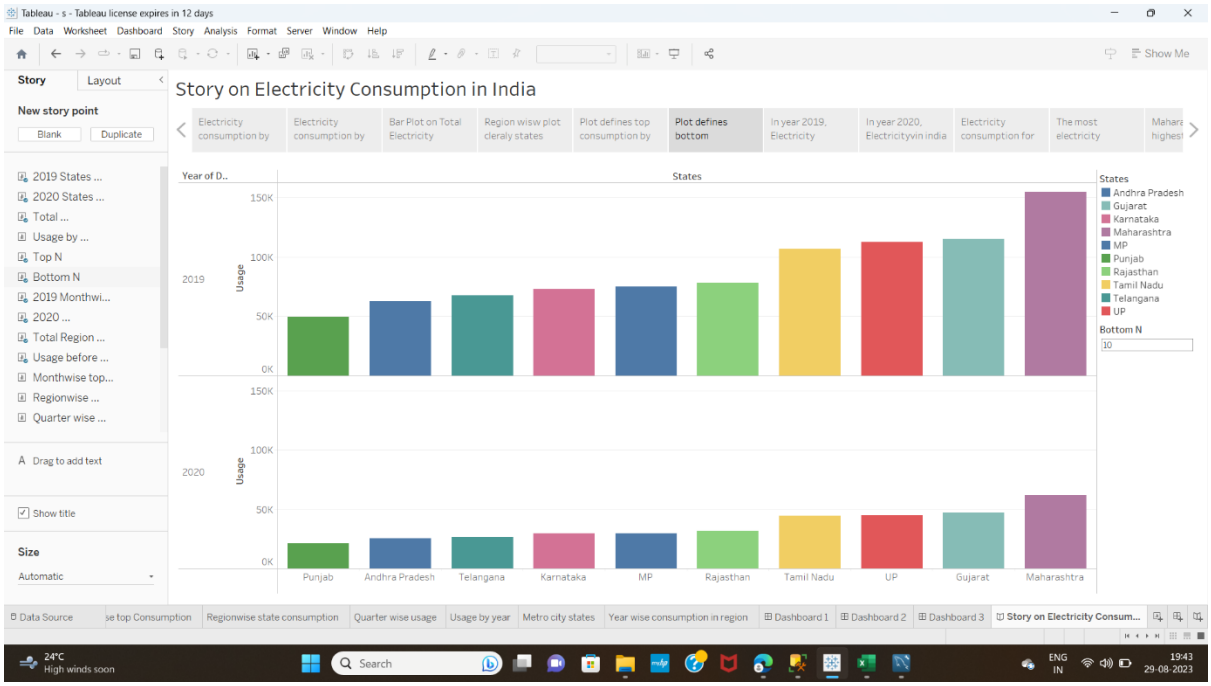


Tableau - s - Tableau license expires in 12 days
File Data Worksheet Dashboard Story Analysis Format Server Window Help
Story
Layout
New story point
Blank
Duplicate
2019 States ...
2020 States ...
Total ...
Usage by ...
Top N
Bottom N
2019 Monthwi...
2020 ...
Total Region ...
Usage before ...
Monthwise top...
Regionwise ...
Quarter wise ...
Drag to add text
Show title
Size
Automatic
Story on Electricity Consumption in India
Electricity consumption by
Electricity consumption by
Bar Plot on Total Electricity
Region wisw plot clerely states
Plot defines top consumption by
Plot defines bottom
In year 2019, Electricity
In year 2020, Electricityvin india
Electricity consumption for
The most electricity
Year of D..
States
2019
2020
Usage
150K
100K
50K
0K
Punjab
Andhra Pradesh
Telangana
Karnataka
MP
Rajasthan
Tamil Nadu
UP
Gujarat
Maharashtra
States
Andhra Pradesh
Gujarat
Karnataka
Maharashtra
MP
Punjab
Rajasthan
Tamil Nadu
Telangana
UP
Bottom N
10
Data Source
Regionwise state consumption
Quarter wise usage
Usage by year
Metro city states
Year wise consumption in region
Dashboard 1
Dashboard 2
Dashboard 3
Story on Electricity Consum...
24°C
High winds soon
Search
ENG
IN
19:43
29-08-2023

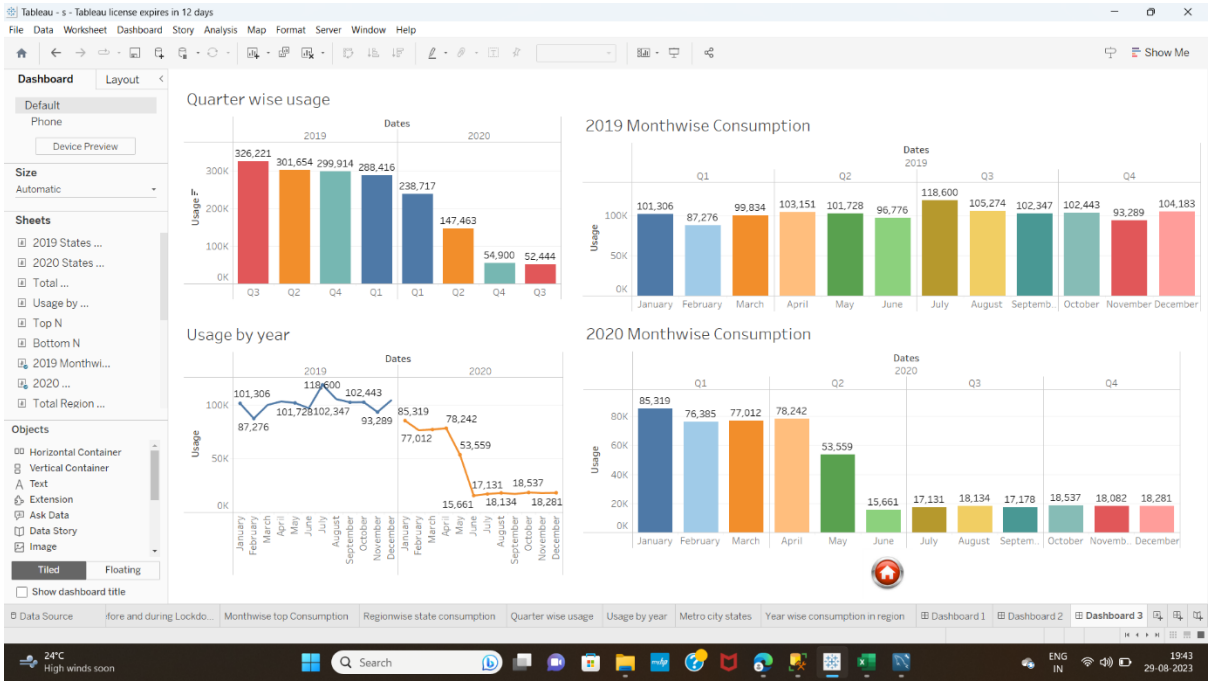
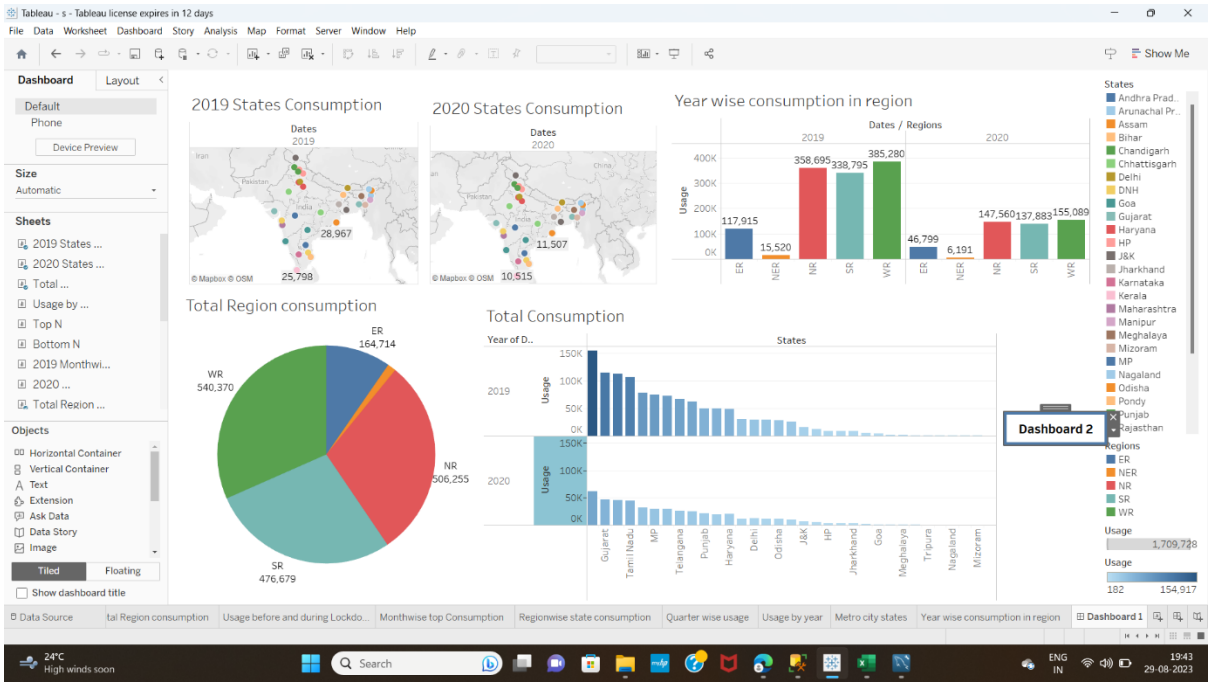
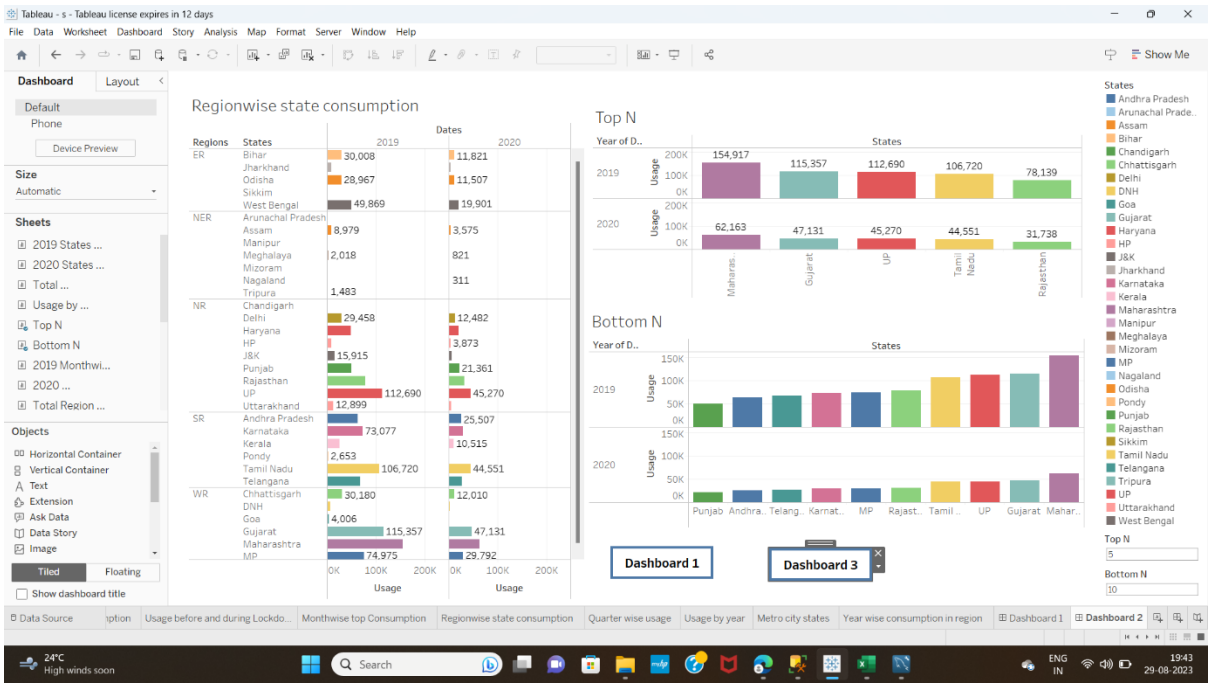


Tableau - s - Tableau license expires in 12 days
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Dashboard
Layout
Default
Phone
Device Preview
Size
Automatic
Sheets
2019 States ...
2020 States ...
Total ...
Usage by ...
Top N
Bottom N
2019 Monthwi...
2020 ...
Total Region ...
Objects
Horizontal Container
Vertical Container
Text
Extension
Ask Data
Data Story
Image
Tiled
Floating
Show dashboard title
Quarter wise usage
Dates
2019
2020
Usage F..
326,221
301,654
299,914
288,416
238,717
147,463
54,900
52,444
Q3
Q2
Q4
Q1
Q1
Q2
Q4
Q3
2019 Monthwise Consumption
Dates
2019
Q1
Q2
Q3
Q4
Usage
101,306
87,276
99,834
103,151
101,728
96,776
118,600
105,274
102,347
102,443
93,289
104,183
January
February
March
April
May
June
July
August
Septemb..
October
November
December
Usage by year
Dates
2019
2020
Usage
101,306
87,276
101,728
118,600
102,347
102,443
93,289
85,319
77,012
78,242
53,559
17,131
15,661
18,134
18,537
18,281
2020 Monthwise Consumption
Dates
2020
Q1
Q2
Q3
Q4
Usage
85,319
76,385
77,012
78,242
53,559
15,661
17,131
18,134
17,178
18,537
18,082
18,281
January
February
March
April
May
June
July
August
Septem..
October
Novemb..
December
Data Source
Monthwise top Consumption
Regionwise state consumption
Quarter wise usage
Usage by year
Metro city states
Year wise consumption in region
Dashboard 1
Dashboard 2
Dashboard 3
24°C
High winds soon
Search
ENG
IN
19:43
29-08-2023


2019 States Consumption
2020 States Consumption
Year wise consumption in region
Total Region consumption
Total Consumption
Dashboard 2


Regionwise state consumption
Top N
Bottom N
Dashboard 1
Dashboard 3

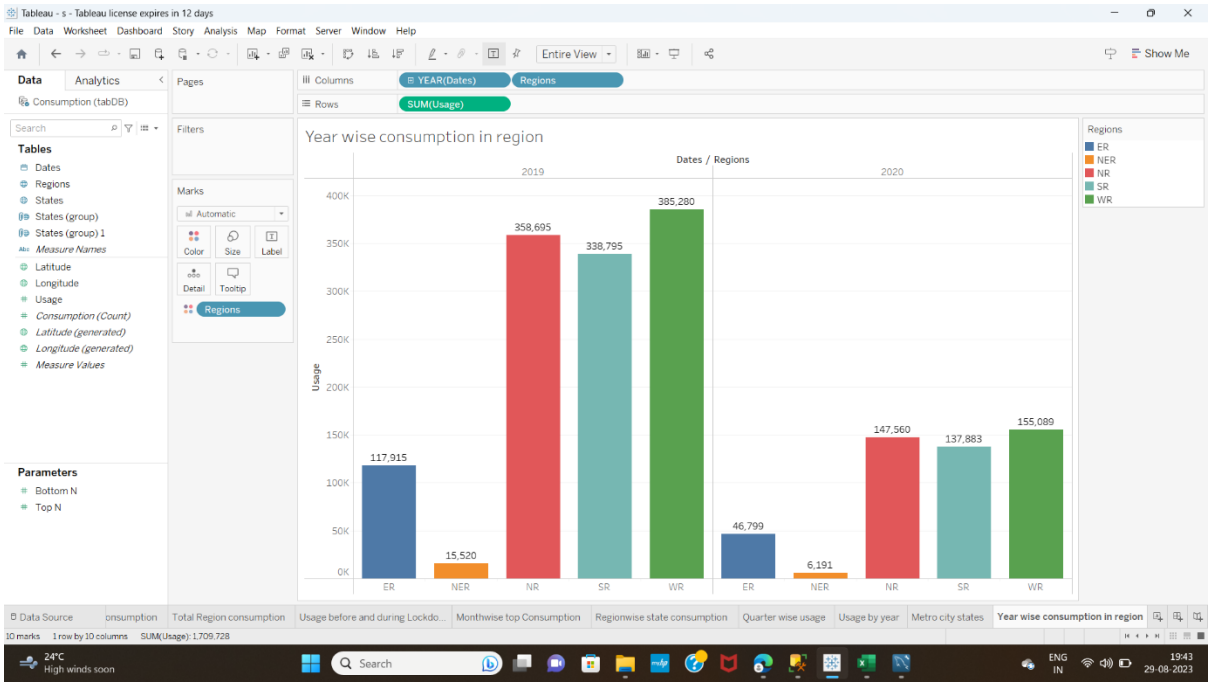


Tableau - s - Tableau license expires in 12 days
Year wise consumption in region
Dates / Regions
2019
2020
Usage
117,915
15,520
358,695
338,795
385,280
46,799
6,191
147,560
137,883
155,089
ER
NER
NR
SR
WR
Regions
Data Source
Total Region consumption
Usage before and during Lockdo...
Monthwise top Consumption
Regionwise state consumption
Quarter wise usage
Usage by year
Metro city states
Year wise consumption in region

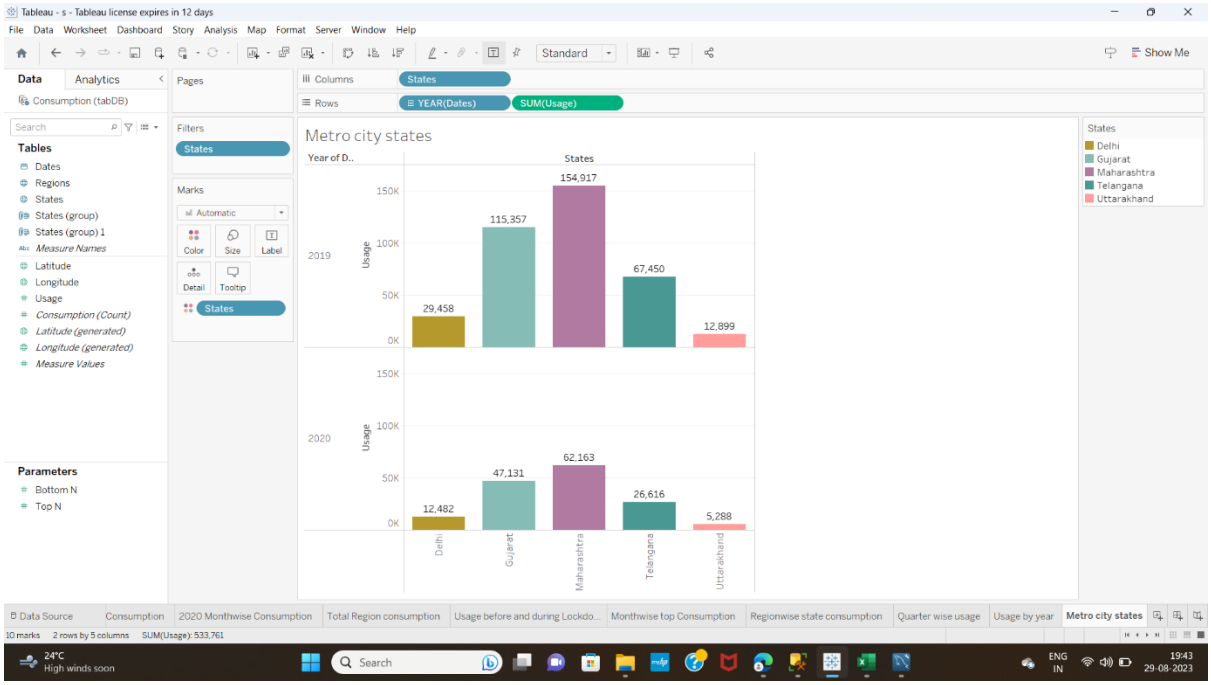
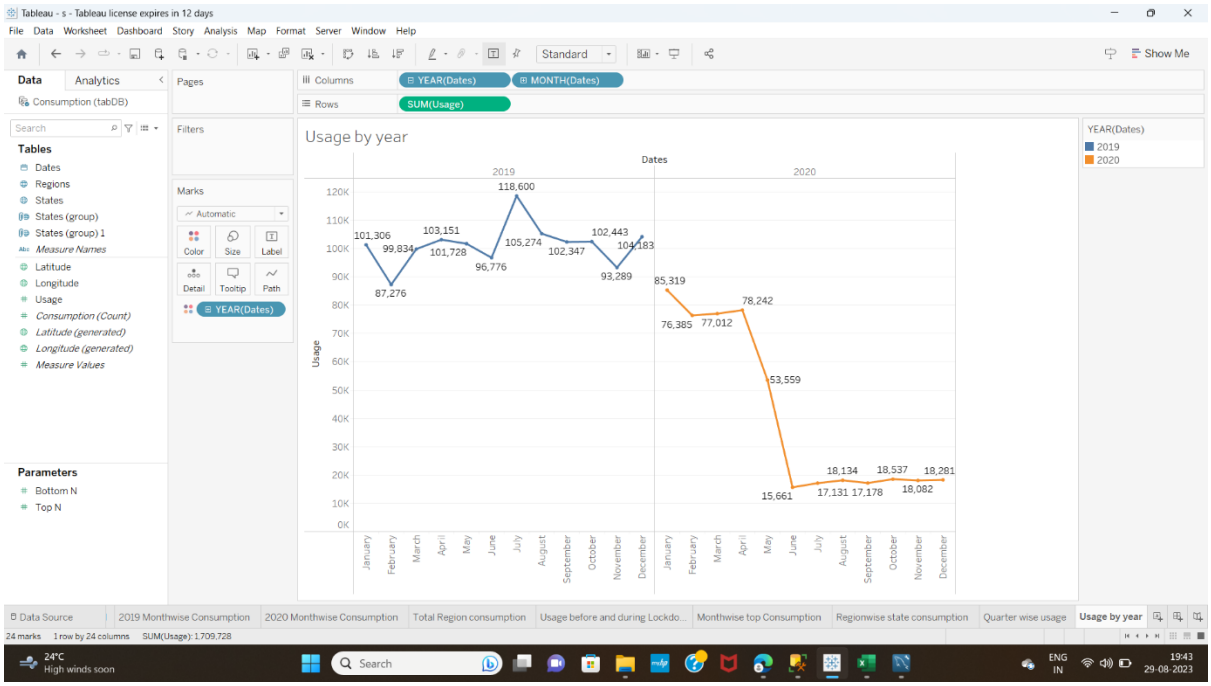
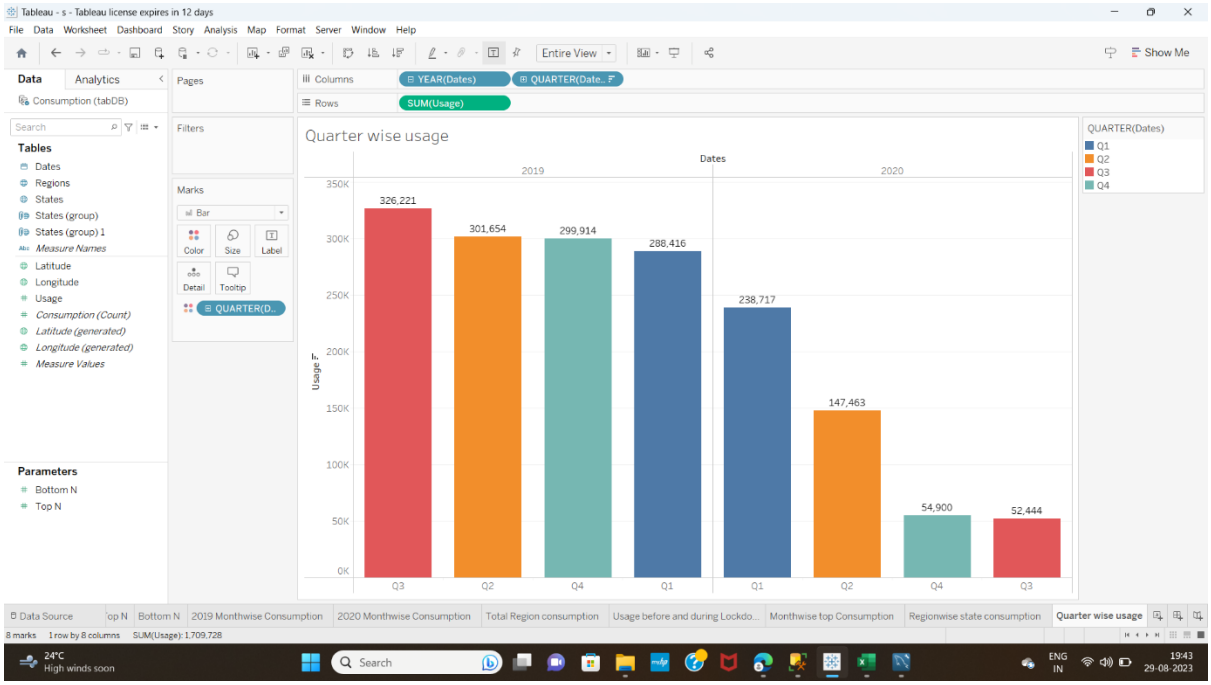
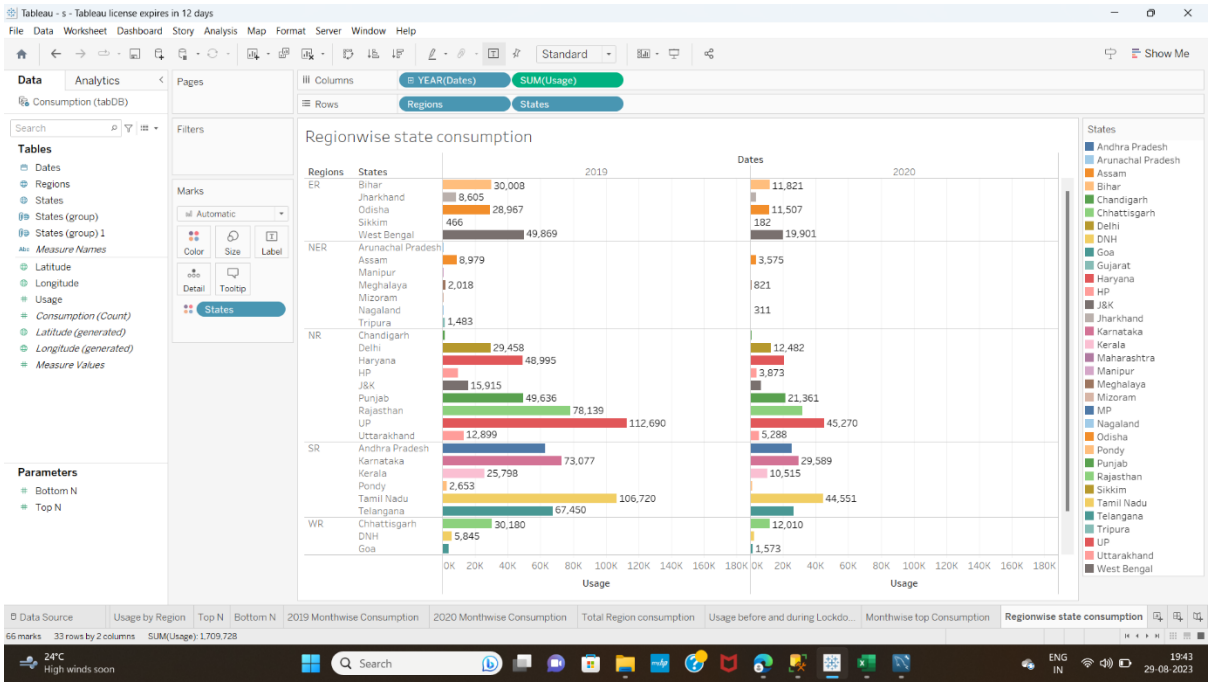
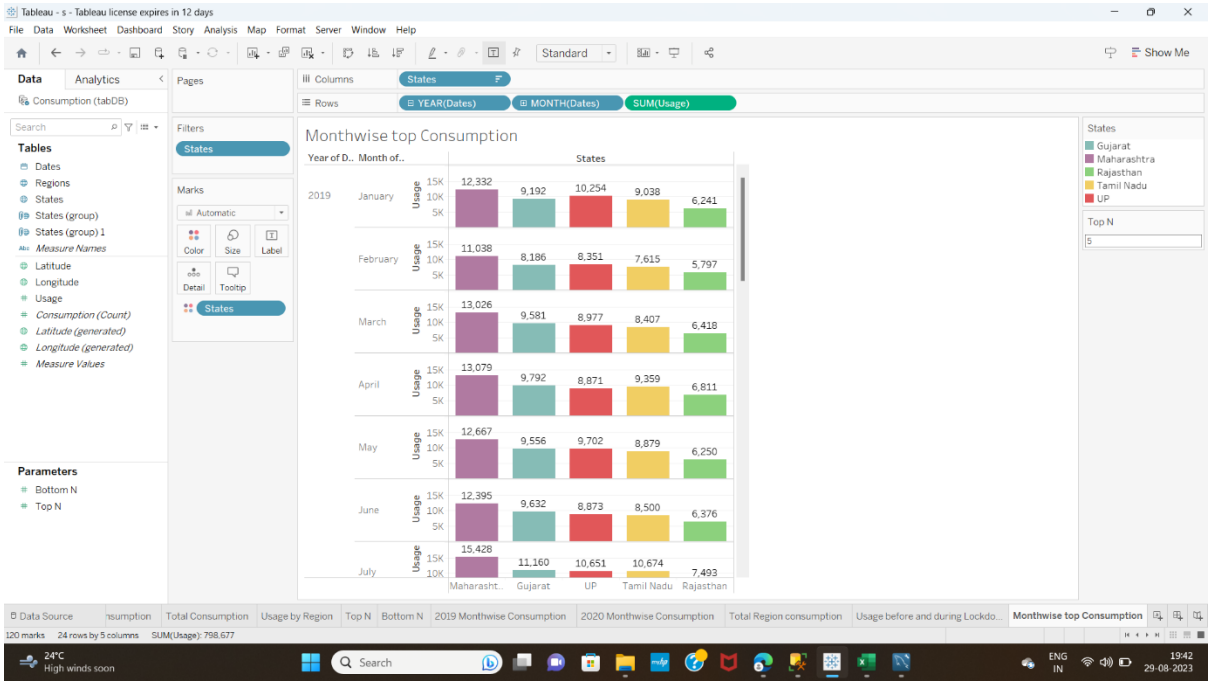


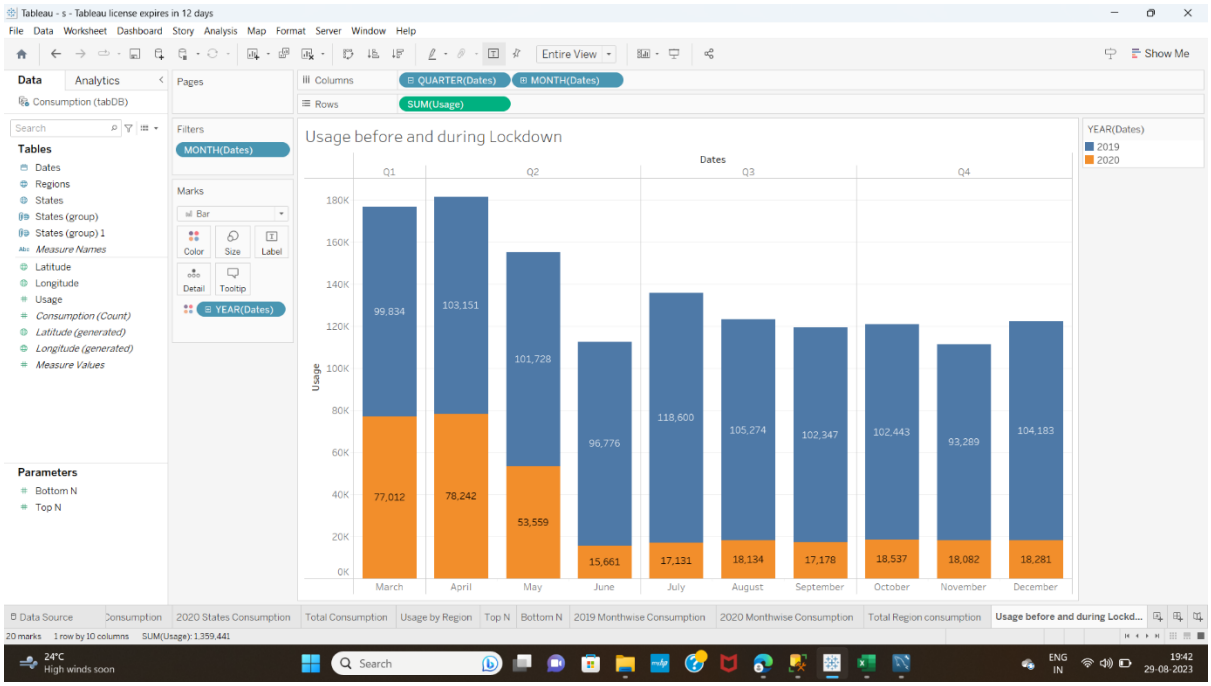
Tableau - s - Tableau license expires in 12 days
Metro city states
States
Year of D..
2019
2020
Usage
29,458
115,357
154,917
67,450
12,899
12,482
47,131
62,163
26,616
5,288
Delhi
Gujarat
Maharashtra
Telangana
Uttarakhand
Data Source
Consumption
2020 Monthwise Consumption
Total Region consumption
Usage before and during Lockdo...
Monthwise top Consumption
Regionwise state consumption
Quarter wise usage
Usage by year
Metro city states

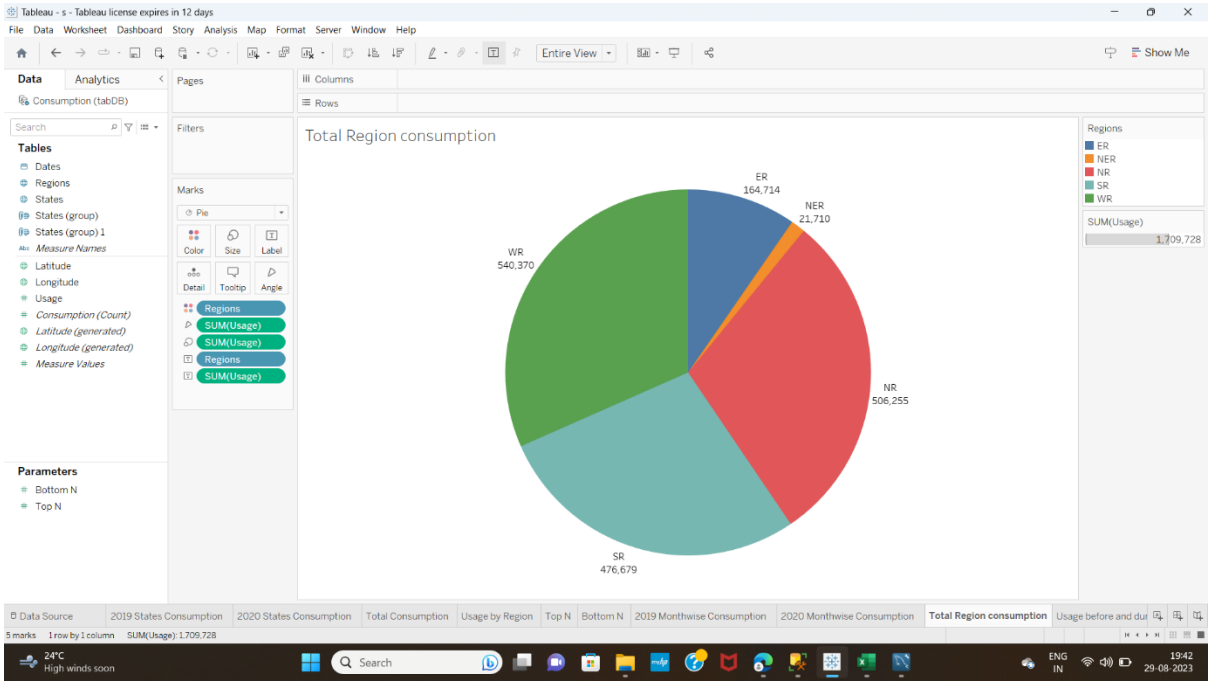

Usage by year
Dates
2019
2020
YEAR(Dates)
2019
2020
Usage
101,306
99,834
87,276
103,151
101,728
96,776
118,600
105,274
102,347
102,443
104,183
93,289
85,319
76,385
77,012
78,242
53,559
15,661
18,134
17,131
17,178
18,537
18,082
18,281
January
February
March
April
May
June
July
August
September
October
November
December


Quarter wise usage
Dates
2019
2020
QUARTER(Dates)
Q1
Q2
Q3
Q4
326,221
301,654
299,914
288,416
238,717
147,463
54,900
52,444
Q3
Q2
Q4
Q1
Q1
Q2
Q4
Q3


Regionwise state consumption
Dates
2019
2020
Regions
States
Usage
Data
Analytics
Pages
Filters
Marks
Columns
YEAR(Dates)
SUM(Usage)
Rows
Regions
States
Tables
Parameters
Bottom N
Top N
Data Source
Usage by Region
Top N
Bottom N
2019 Monthwise Consumption
2020 Monthwise Consumption
Total Region consumption
Usage before and during Lockdo...
Monthwise top Consumption
Regionwise state consumption


Monthwise top Consumption
Year of D..
Month of..
States
2019
January
February
March
April
May
June
July
Usage
Maharasht...
Gujarat
UP
Tamil Nadu
Rajasthan
Data
Analytics
Pages
Filters
Marks
Columns
States
Rows
YEAR(Dates)
MONTH(Dates)
SUM(Usage)
Top N
Data Source
Total Consumption
Usage by Region
Top N
Bottom N
2019 Monthwise Consumption
2020 Monthwise Consumption
Total Region consumption
Usage before and during Lockdo...
Monthwise top Consumption


Usage before and during Lockdown
Dates
Q1
Q2
Q3
Q4
Usage
180K
160K
140K
120K
100K
80K
60K
40K
20K
0K
99,834
77,012
103,151
78,242
101,728
53,559
96,776
15,661
118,600
17,131
105,274
18,134
102,347
17,178
102,443
18,537
93,289
18,082
104,183
18,281
March
April
May
June
July
August
September
October
November
December
YEAR(Dates)
2019
2020


Total Region consumption
ER
164,714
NER
21,710
NR
506,255
SR
476,679
WR
540,370
Regions
ER
NER
NR
SR
WR
SUM(Usage)
1,709,728

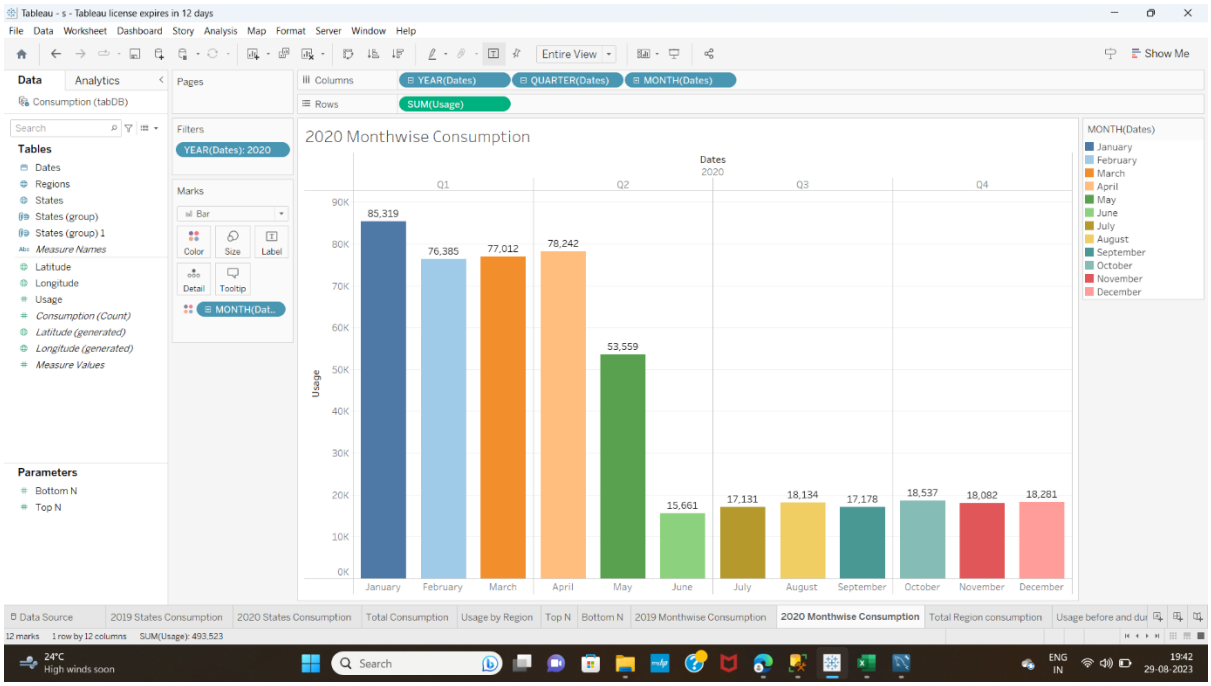


Tableau - s - Tableau license expires in 12 days
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Entire View
Show Me
Data
Analytics
Consumption (tabDB)
Tables
Dates
Regions
States
States (group)
States (group) 1
Measure Names
Latitude
Longitude
Usage
Consumption (Count)
Latitude (generated)
Longitude (generated)
Measure Values
Parameters
Bottom N
Top N
Pages
Filters
YEAR(Dates): 2020
Marks
Bar
Color
Size
Label
Detail
Tooltip
MONTH(Dat..
Columns
YEAR(Dates)
QUARTER(Dates)
MONTH(Dates)
Rows
SUM(Usage)
2020 Monthwise Consumption
Dates
2020
Q1
Q2
Q3
Q4
Usage
90K
80K
70K
60K
50K
40K
30K
20K
10K
0K
85,319
76,385
77,012
78,242
53,559
15,661
17,131
18,134
17,178
18,537
18,082
18,281
January
February
March
April
May
June
July
August
September
October
November
December
MONTH(Dates)
January
February
March
April
May
June
July
August
September
October
November
December
Data Source
2019 States Consumption
2020 States Consumption
Total Consumption
Usage by Region
Top N
Bottom N
2019 Monthwise Consumption
2020 Monthwise Consumption
Total Region consumption
Usage before and dur
12 marks
1 row by 12 columns
SUM(Usage): 493,523
24°C
High winds soon
Search
ENG
IN
19:42
29-08-2023

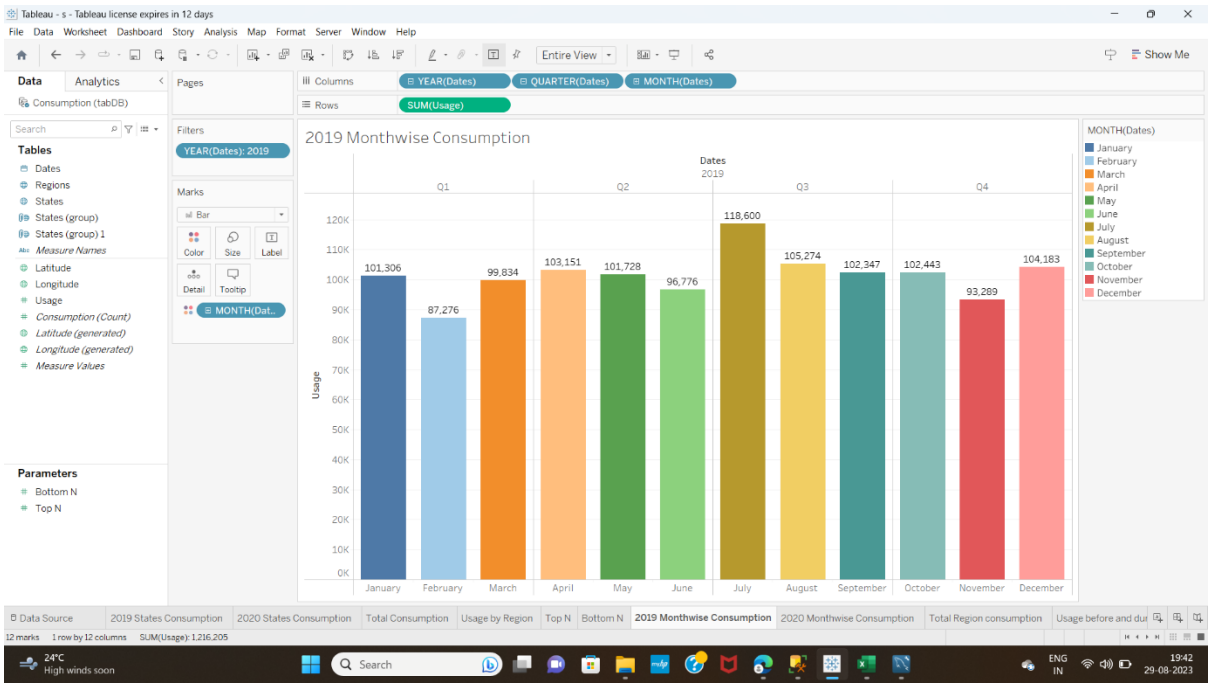


Tableau - s - Tableau license expires in 12 days
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Entire View
Show Me
Data
Analytics
Consumption (tabDB)
Tables
Dates
Regions
States
States (group)
States (group) 1
Measure Names
Latitude
Longitude
Usage
Consumption (Count)
Latitude (generated)
Longitude (generated)
Measure Values
Parameters
Bottom N
Top N
Pages
Filters
YEAR(Dates): 2019
Marks
Bar
Color
Size
Label
Detail
Tooltip
MONTH(Dat..
Columns
YEAR(Dates)
QUARTER(Dates)
MONTH(Dates)
Rows
SUM(Usage)
2019 Monthwise Consumption
Dates
2019
Q1
Q2
Q3
Q4
Usage
120K
110K
100K
90K
80K
70K
60K
50K
40K
30K
20K
10K
0K
101,306
87,276
99,834
103,151
101,728
96,776
118,600
105,274
102,347
102,443
93,289
104,183
January
February
March
April
May
June
July
August
September
October
November
December
MONTH(Dates)
January
February
March
April
May
June
July
August
September
October
November
December
Data Source
2019 States Consumption
2020 States Consumption
Total Consumption
Usage by Region
Top N
Bottom N
2019 Monthwise Consumption
2020 Monthwise Consumption
Total Region consumption
Usage before and dur
12 marks
1 row by 12 columns
SUM(Usage): 1,216,205
24°C
High winds soon
Search
ENG
IN
19:42
29-08-2023

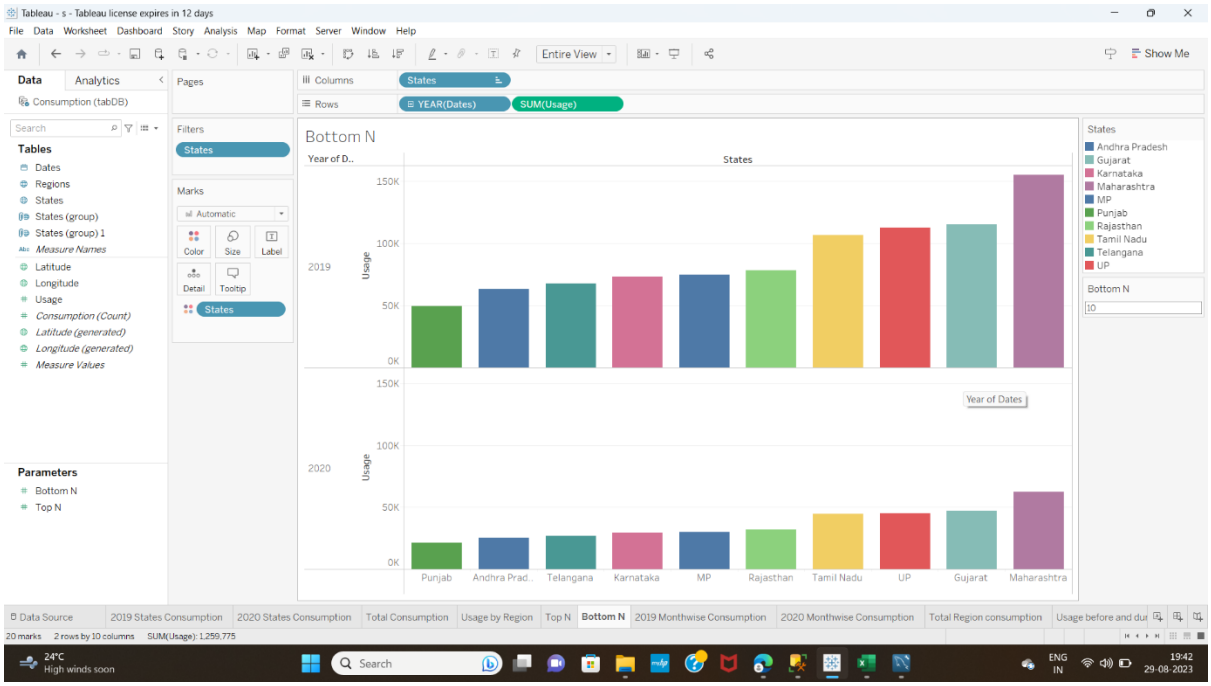


Tableau - s - Tableau license expires in 12 days
Bottom N
Year of D..
States
2019
2020
Usage
Year of Dates
Punjab
Andhra Prad..
Telangana
Karnataka
MP
Rajasthan
Tamil Nadu
UP
Gujarat
Maharashtra
Andhra Pradesh
Gujarat
Karnataka
Maharashtra
MP
Punjab
Rajasthan
Tamil Nadu
Telangana
UP
Bottom N
10

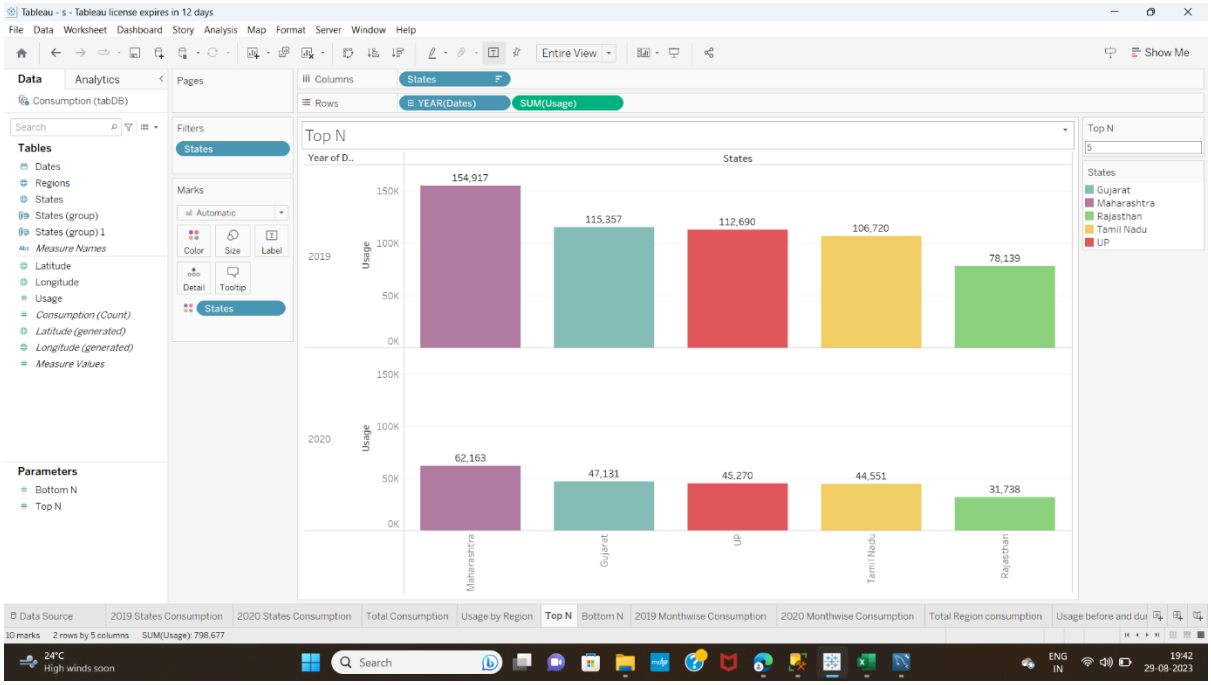
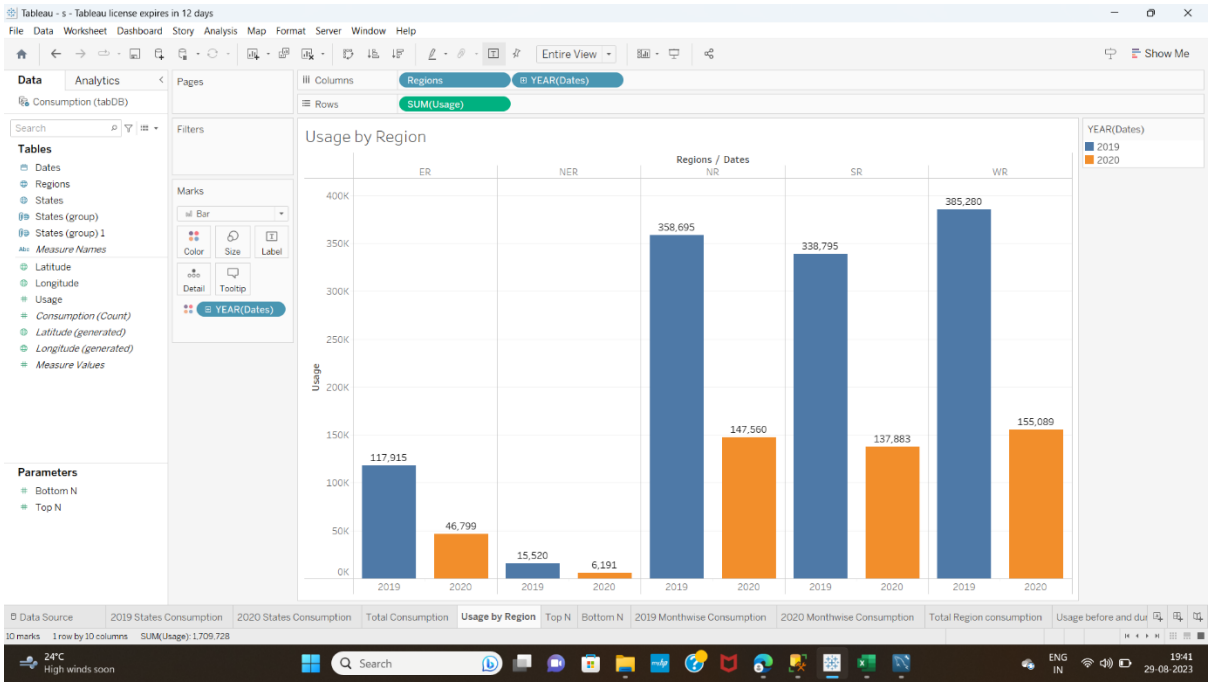
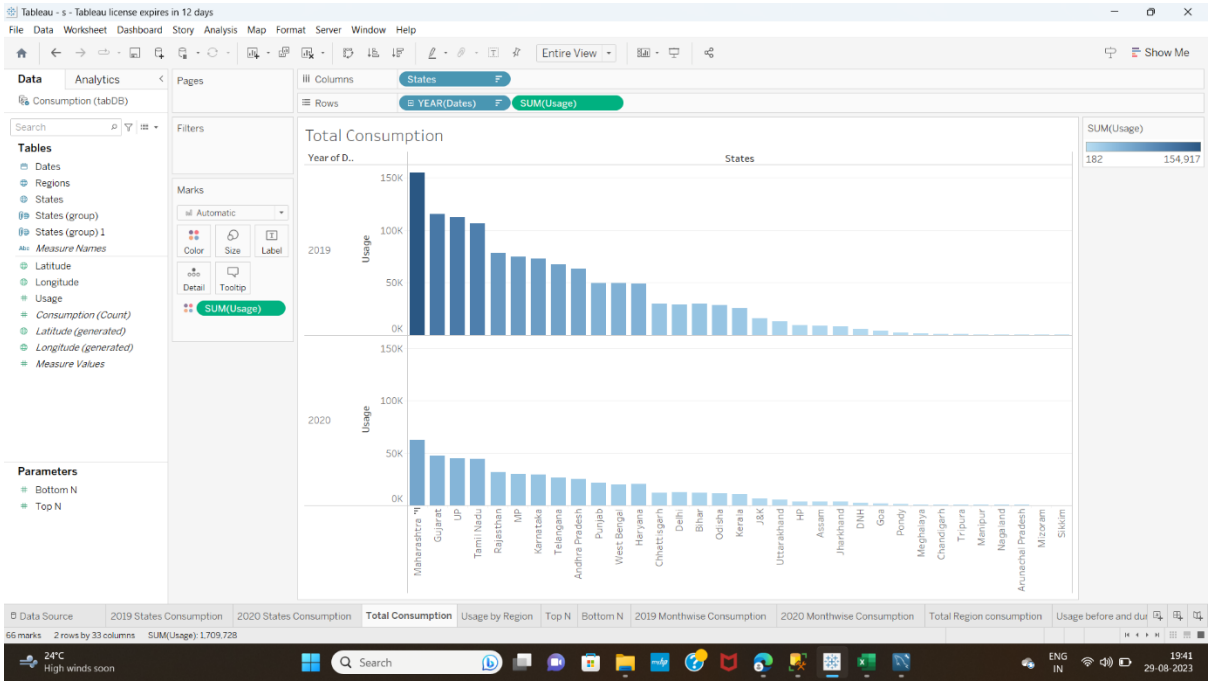


Tableau - s - Tableau license expires in 12 days
Top N
Year of D..
States
2019
2020
Usage
154,917
115,357
112,690
106,720
78,139
62,163
47,131
45,270
44,551
31,738
Maharashtra
Gujarat
UP
Tamil Nadu
Rajasthan
Top N
5


Usage by Region
Regions / Dates
ER
NER
NR
SR
WR
Usage
117,915
46,799
15,520
6,191
358,695
147,560
338,795
137,883
385,280
155,089
2019
2020
YEAR(Dates)


Total Consumption
States
SUM(Usage)
2019
2020
Usage
Maharashtra
Gujarat
UP
Tamil Nadu
Rajasthan
MP
Karnataka
Telangana
Andhra Pradesh
Punjab
West Bengal
Haryana
Chhattisgarh
Delhi
Bihar
Odisha
Kerala
J&K
Uttarakhand
HP
Assam
Jharkhand
DNH
Goa
Pondy
Meghalaya
Chandigarh
Tripura
Manipur
Nagaland
Arunachal Pradesh
Mizoram
Sikkim

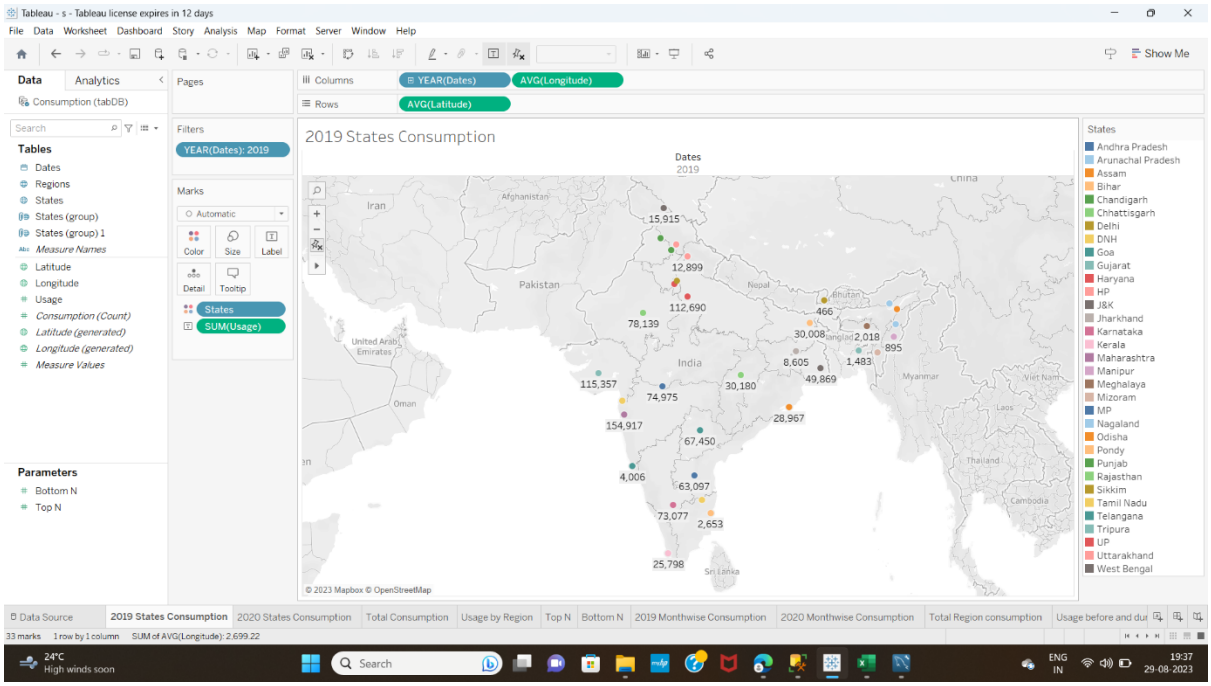
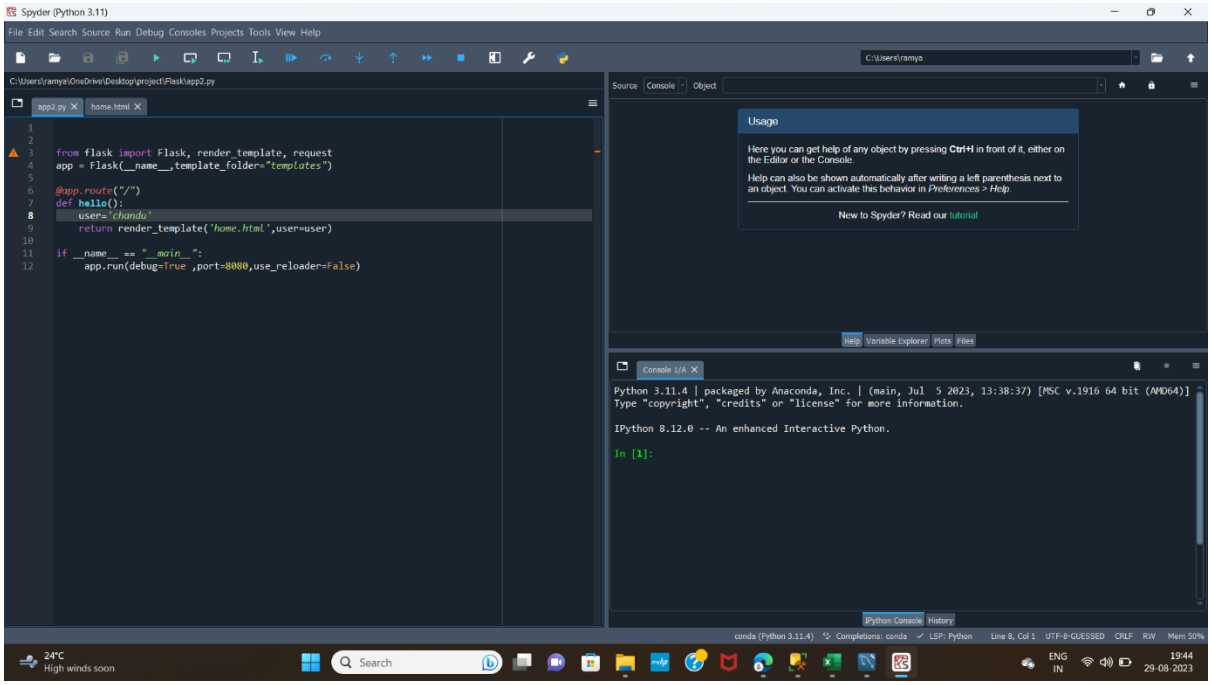
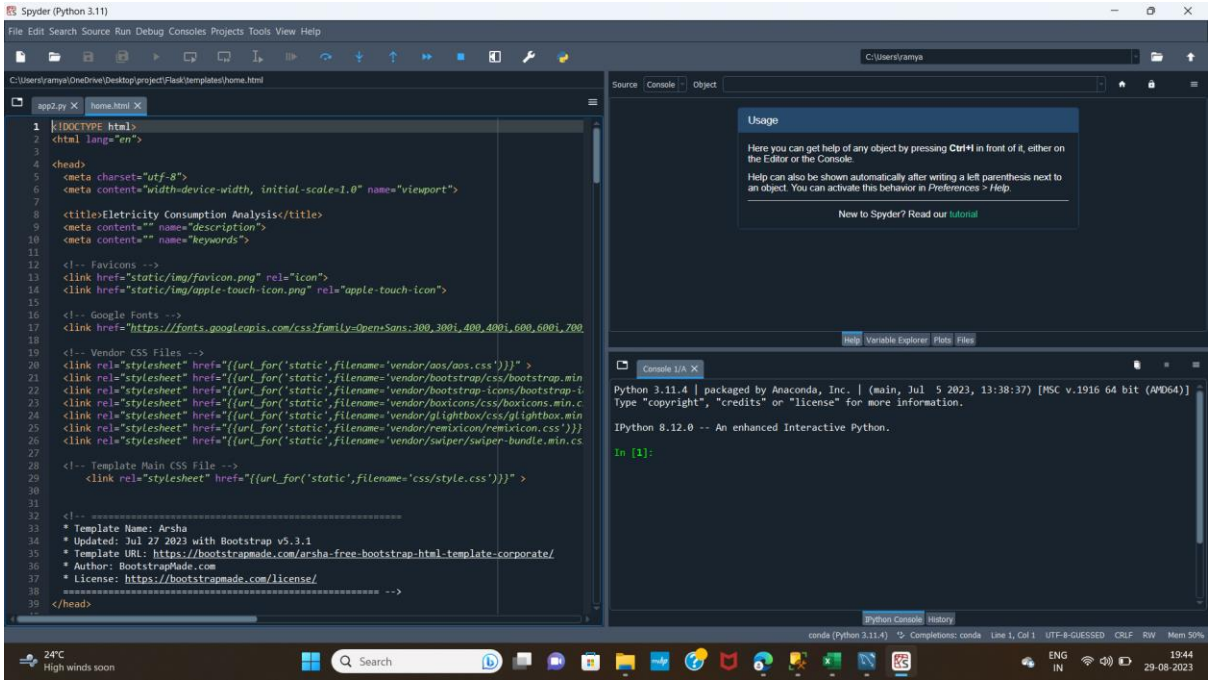
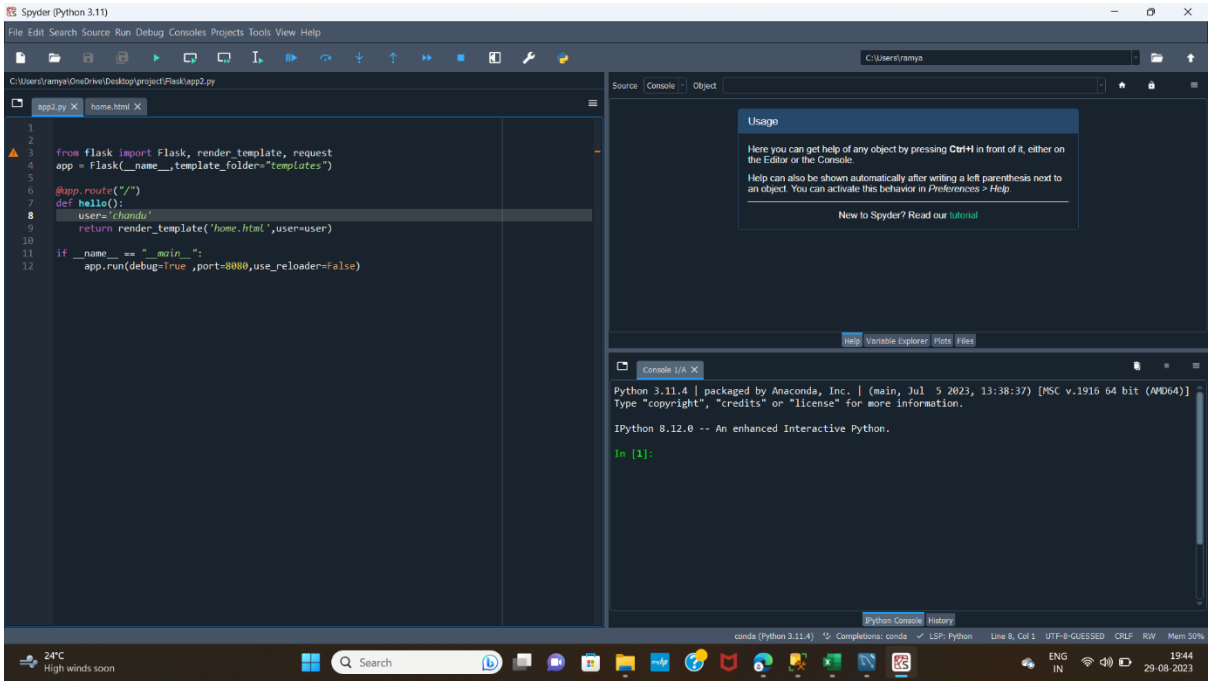
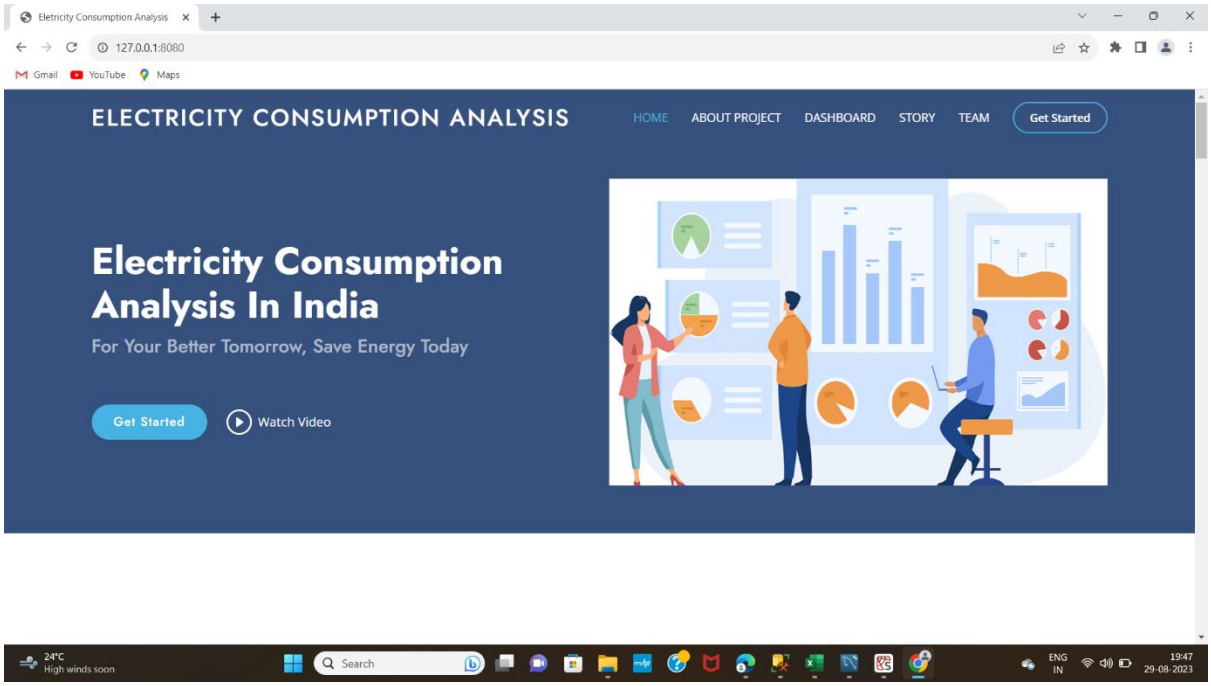


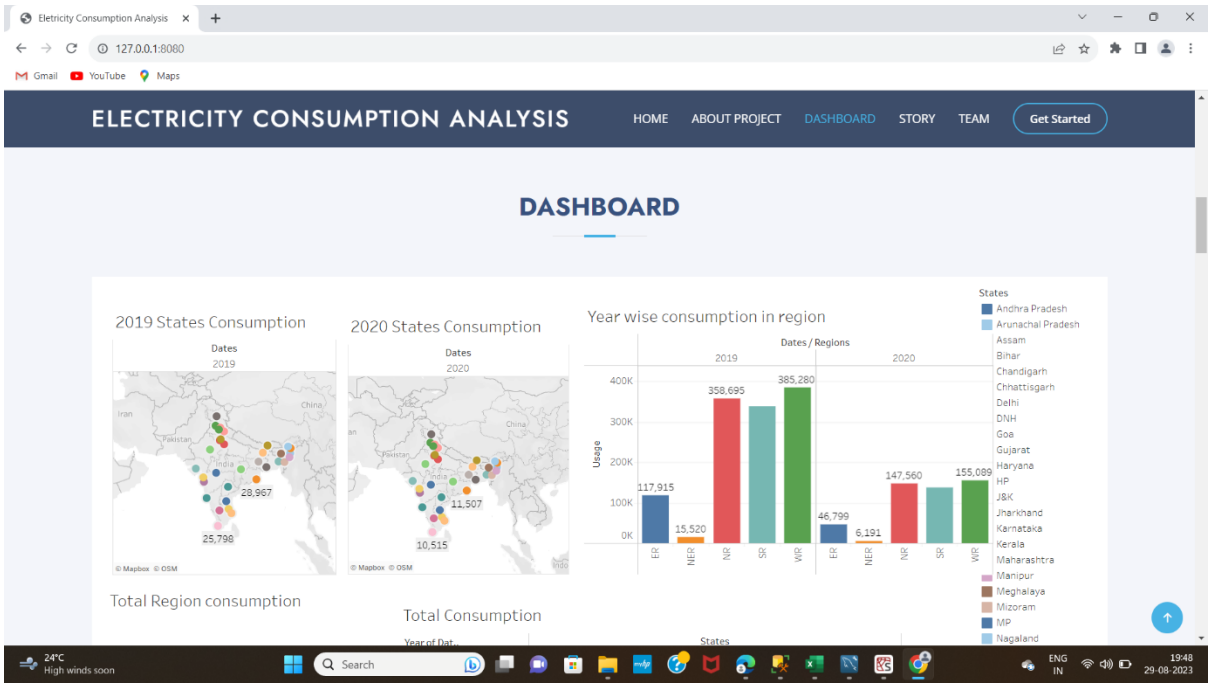
Tableau - s - Tableau license expires in 12 days
File Data Worksheet Dashboard Story Analysis Map Format Server Window Help
Data
Analytics
Consumption (tabDB)
Tables
Dates
Regions
States
States (group)
States (group) 1
Measure Names
Latitude
Longitude
Usage
Consumption (Count)
Latitude (generated)
Longitude (generated)
Measure Values
Parameters
Bottom N
Top N
Pages
Filters
YEAR(Dates): 2019
Marks
Automatic
Color
Size
Label
Detail
Tooltip
States
SUM(Usage)
Columns
YEAR(Dates)
AVG(Longitude)
Rows
AVG(Latitude)
Show Me
2019 States Consumption
Dates
2019
15,915
12,899
112,690
78,139
30,008
466
2,018
895
1,483
8,605
49,869
30,180
115,357
74,975
154,917
28,967
67,450
4,006
63,097
73,077
2,653
25,798
© 2023 Mapbox © OpenStreetMap
States
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chandigarh
Chhattisgarh
Delhi
DNH
Goa
Gujarat
Haryana
HP
J&K
Jharkhand
Karnataka
Kerala
Maharashtra
Manipur
Meghalaya
Mizoram
MP
Nagaland
Odisha
Pondy
Punjab
Rajasthan
Sikkim
Tamil Nadu
Telangana
Tripura
UP
Uttarakhand
West Bengal
Data Source
2019 States Consumption
2020 States Consumption
Total Consumption
Usage by Region
Top N
Bottom N
2019 Monthwise Consumption
2020 Monthwise Consumption
Total Region consumption
33 marks
1 row by 1 column
SUM of AVG(Longitude): 2,699.22
24°C
High winds soon
Search
ENG
IN
19:37
29-08-2023

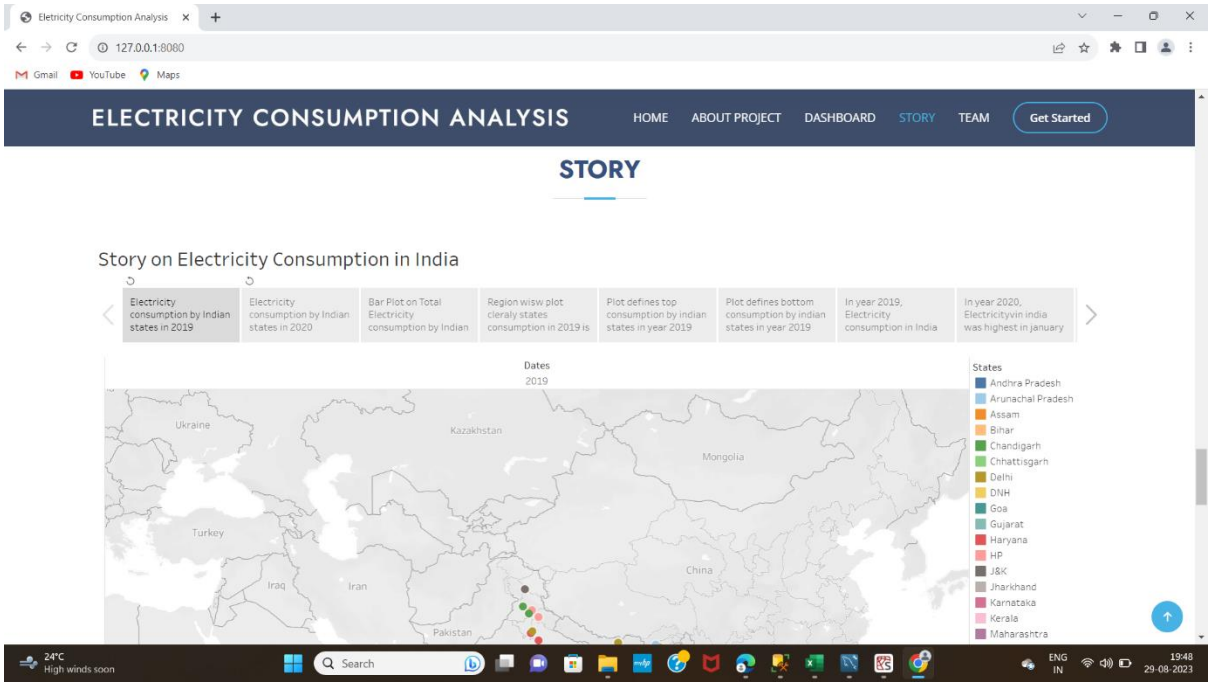

Spyder (Python 3.11)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\ramya\OneDrive\Desktop\project\Flask\app2.py
app2.py
home.html
from flask import Flask, render_template, request
app = Flask(__name__,template_folder="templates")
@app.route("/")
def hello():
user='chandu'
return render_template('home.html',user=user)
if __name__ == "__main__":
app.run(debug=True ,port=8080,use_reloader=False)
C:\Users\ramya
Source
Console
Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Help
Variable Explorer
Plots
Files
Console 1/A
Python 3.11.4 | packaged by Anaconda, Inc. | (main, Jul 5 2023, 13:38:37) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 8.12.0 -- An enhanced Interactive Python.
In [1]:
IPython Console
History
conda (Python 3.11.4)
Completions: conda
LSP: Python
Line 8, Col 1
UTF-8-GUESSED
CRLF
RW
Mem 50%
24°C
High winds soon
Search
ENG
IN
19:44
29-08-2023

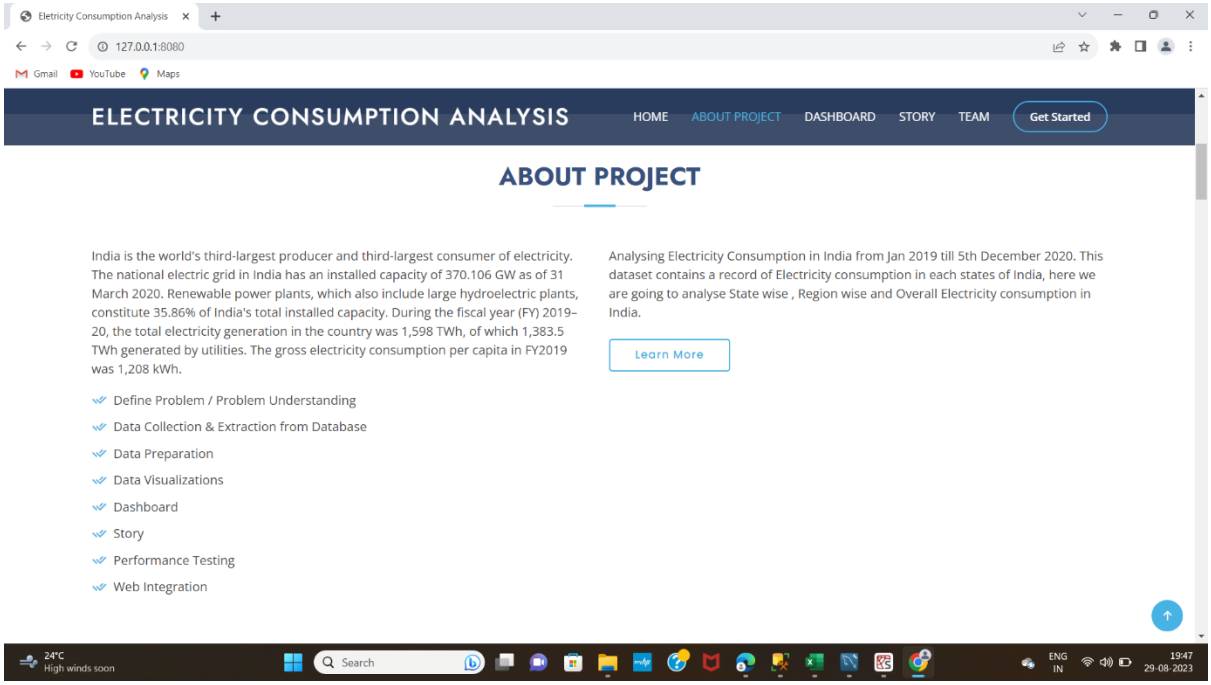

Spyder (Python 3.11)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\ramya\OneDrive\Desktop\project\Flask\templates\home.html
app2.py
home.html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta content="width=device-width, initial-scale=1.0" name="viewport">
<title>Eletricity Consumption Analysis</title>
<meta content="" name="description">
<meta content="" name="keywords">
<!-- Favicons -->
<link href="static/img/favicon.png" rel="icon">
<link href="static/img/apple-touch-icon.png" rel="apple-touch-icon">
<!-- Google Fonts -->
<link href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700
<!-- Vendor CSS Files -->
<link rel="stylesheet" href="{{url_for('static',filename='vendor/aos/aos.css')}}" >
<link rel="stylesheet" href="{{url_for('static',filename='vendor/bootstrap/css/bootstrap.min
<link rel="stylesheet" href="{{url_for('static',filename='vendor/bootstrap-icons/bootstrap-i
<link rel="stylesheet" href="{{url_for('static',filename='vendor/boxicons/css/boxicons.min.c
<link rel="stylesheet" href="{{url_for('static',filename='vendor/glightbox/css/glightbox.min
<link rel="stylesheet" href="{{url_for('static',filename='vendor/remixicon/remixicon.css')}}
<link rel="stylesheet" href="{{url_for('static',filename='vendor/swiper/swiper-bundle.min.cs
<!-- Template Main CSS File -->
<link rel="stylesheet" href="{{url_for('static',filename='css/style.css')}}" >
<!--
* Template Name: Arsha
* Updated: Jul 27 2023 with Bootstrap v5.3.1
* Template URL: https://bootstrapmade.com/arsha-free-bootstrap-html-template-corporate/
* Author: BootstrapMade.com
* License: https://bootstrapmade.com/license/
-->
</head>
Source Console Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Help Variable Explorer Plots Files
Console 1/A
Python 3.11.4 | packaged by Anaconda, Inc. | (main, Jul 5 2023, 13:38:37) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 8.12.0 -- An enhanced Interactive Python.
In [1]:
IPython Console History
conda (Python 3.11.4) Completions: conda Line 1, Col 1 UTF-8-GUESSED CRLF RW Mem 50%
24°C
High winds soon
Search
ENG
IN
19:44
29-08-2023


Spyder (Python 3.11)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\ramya\OneDrive\Desktop\project\Flask\app2.py
app2.py
home.html
from flask import Flask, render_template, request
app = Flask(__name__,template_folder="templates")
@app.route("/")
def hello():
user='chandu'
return render_template('home.html',user=user)
if __name__ == "__main__":
app.run(debug=True ,port=8080,use_reloader=False)
Source Console Object
Usage
Here you can get help of any object by pressing Ctrl+I in front of it, either on the Editor or the Console.
Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in Preferences > Help.
New to Spyder? Read our tutorial
Help Variable Explorer Plots Files
Console 1/A
Python 3.11.4 | packaged by Anaconda, Inc. | (main, Jul 5 2023, 13:38:37) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.
IPython 8.12.0 -- An enhanced Interactive Python.
In [1]:
IPython Console History
conda (Python 3.11.4) Completions: conda LSP: Python Line 8, Col 1 UTF-8-GUESSED CRLF RW Mem 50%
24°C
High winds soon
Search
ENG
IN
19:44
29-08-2023


Eletricity Consumption Analysis
127.0.0.1:8080
Gmail
YouTube
Maps
ELECTRICITY CONSUMPTION ANALYSIS
HOME
ABOUT PROJECT
DASHBOARD
STORY
TEAM
Get Started
Electricity Consumption Analysis In India
For Your Better Tomorrow, Save Energy Today
Get Started
Watch Video
24°C
High winds soon
Search
ENG
IN
19:47
29-08-2023


Eletricity Consumption Analysis
127.0.0.1:8080
Gmail
YouTube
Maps
ELECTRICITY CONSUMPTION ANALYSIS
HOME
ABOUT PROJECT
DASHBOARD
STORY
TEAM
Get Started
DASHBOARD
2019 States Consumption
Dates
2019
28,967
25,798
2020 States Consumption
Dates
2020
11,507
10,515
Year wise consumption in region
Dates / Regions
2019
2020
Usage
400K
300K
200K
100K
0K
117,915
15,520
358,695
385,280
46,799
6,191
147,560
155,089
ER
NER
NR
SR
WR
States
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chandigarh
Chhattisgarh
Delhi
DNH
Goa
Gujarat
Haryana
HP
J&K
Jharkhand
Karnataka
Kerala
Maharashtra
Manipur
Meghalaya
Mizoram
MP
Nagaland
Total Region consumption
Total Consumption
24°C
High winds soon
Search
ENG
IN
19:48
29-08-2023


Eletricity Consumption Analysis
127.0.0.1:8080
Gmail
YouTube
Maps
ELECTRICITY CONSUMPTION ANALYSIS
HOME
ABOUT PROJECT
DASHBOARD
STORY
TEAM
Get Started
STORY
Story on Electricity Consumption in India
Electricity consumption by Indian states in 2019
Electricity consumption by Indian states in 2020
Bar Plot on Total Electricity consumption by Indian
Region wisw plot clerely states consumption in 2019 is
Plot defines top consumption by indian states in year 2019
Plot defines bottom consumption by indian states in year 2019
In year 2019, Electricity consumption in India
In year 2020, Electricityvin india was highest in january
Dates
2019
States
Andhra Pradesh
Arunachal Pradesh
Assam
Bihar
Chandigarh
Chhattisgarh
Delhi
DNH
Goa
Gujarat
Haryana
HP
J&K
Jharkhand
Karnataka
Kerala
Maharashtra
Ukraine
Kazakhstan
Mongolia
Turkey
Iraq
Iran
China
Pakistan
24°C
High winds soon
Search
ENG
IN
19:48
29-08-2023


Eletricity Consumption Analysis
127.0.0.1:8080
Gmail
YouTube
Maps
ELECTRICITY CONSUMPTION ANALYSIS
HOME
ABOUT PROJECT
DASHBOARD
STORY
TEAM
Get Started
ABOUT PROJECT
India is the world's third-largest producer and third-largest consumer of electricity. The national electric grid in India has an installed capacity of 370.106 GW as of 31 March 2020. Renewable power plants, which also include large hydroelectric plants, constitute 35.86% of India's total installed capacity. During the fiscal year (FY) 2019–20, the total electricity generation in the country was 1,598 TWh, of which 1,383.5 TWh generated by utilities. The gross electricity consumption per capita in FY2019 was 1,208 kWh.
Analysing Electricity Consumption in India from Jan 2019 till 5th December 2020. This dataset contains a record of Electricity consumption in each states of India, here we are going to analyse State wise , Region wise and Overall Electricity consumption in India.
Learn More
Define Problem / Problem Understanding
Data Collection & Extraction from Database
Data Preparation
Data Visualizations
Dashboard
Story
Performance Testing
Web Integration
24°C
High winds soon
Search
ENG
IN
19:47
29-08-2023