Water Quality Analysis

**OBJECTIVE:**

To Access to safe drinking-water is essential to health, a basic human right and a component of effective policy for health protection. This is important as a health and development issue at a national, regional and local level. In some regions, it has been shown that investments in water supply and sanitation can yield a net economic benefit, since the reductions in adverse health effects and health care costs outweigh the costs of undertaking the interventions. comprehensively evaluate water quality by assessing its potability, detecting deviations from established standards, and explaining the interrelationships among key parameters.

**Data visualization:**

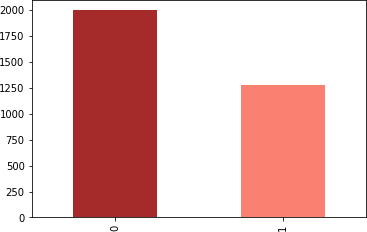
Data visualization plays a crucial role in data analysis, enabling the extraction of meaningful insights from complex datasets. This abstract outlines the process of creating informative and visually appealing data visualizations using Python's Matplotlib library within the Jupyter Anaconda environment.

The dataset chosen for this project is various parameter used for analysing water quality for domestic purpose, and the goal is to effectively communicate key trends, patterns, and relationships within the data. Matplotlib, a versatile and widely used data visualization library, offers a wide range of tools for creating various types of plots, such as line charts, bar graphs, scatter plots, heatmaps, and more.

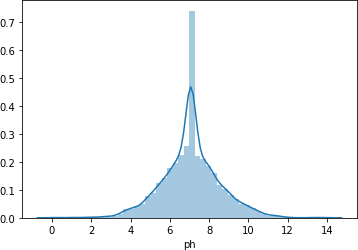
In this project, we will walk through the step-by-step process of data visualization, including data pre- processing, selecting appropriate plot types, customizing aesthetics (colours, labels, legends, etc.), and adding context to the visualizations with titles, annotations, and captions. We will also explore how to handle common data visualization challenges, such as handling missing data, creating subplots, and generating interactive visualizations.

As per the code done in jupyter the analysed data’s are:

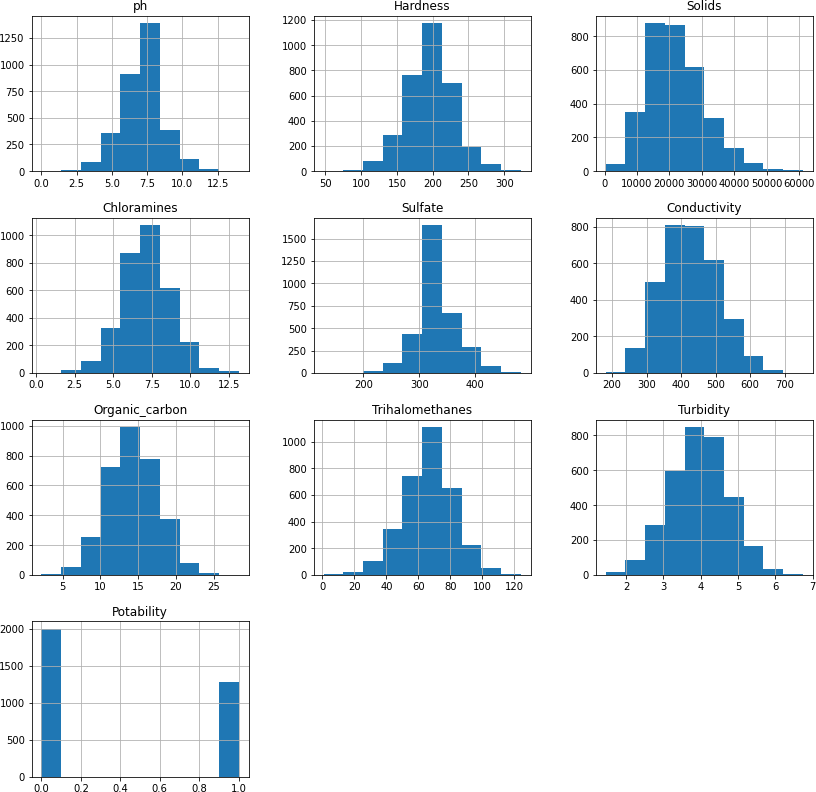
1. Water potability after replacing the zero parameters with mean value of respective data:



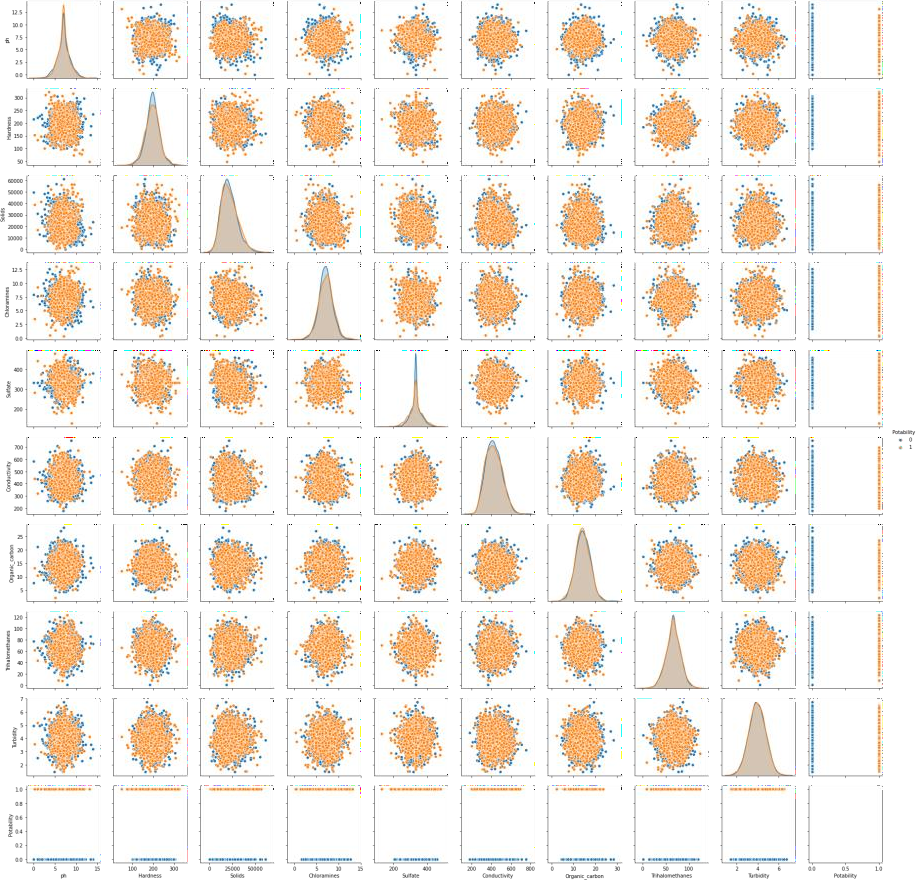
1. General displaying or analysis of ph with axes sup lot using matplot lib:



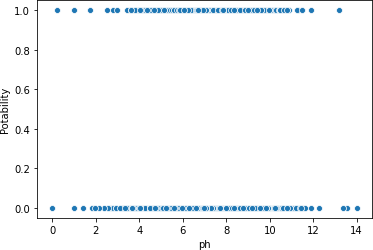
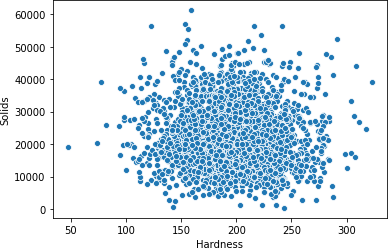
1. Representation of data’s of water parameter in histogram:



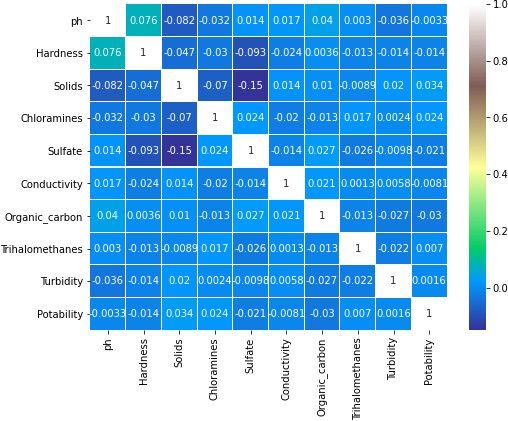
1. Representation of data’s of water parameter in pair plot by fixing hue as potability:



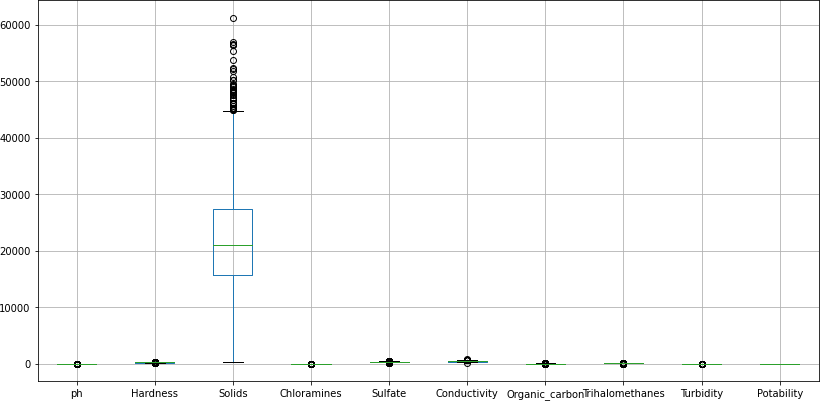
1. Representation of data’s of water parameter in scatter plot:



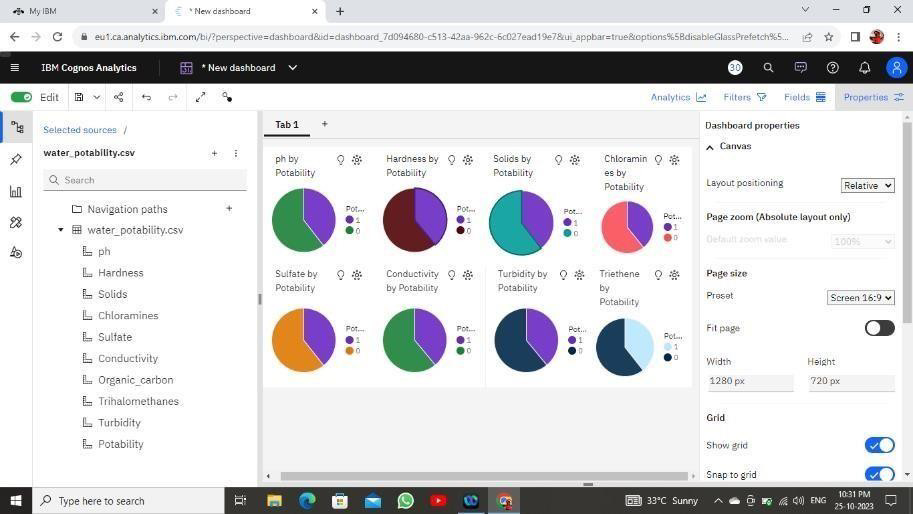
1. Examining the corelation between the given parameter:

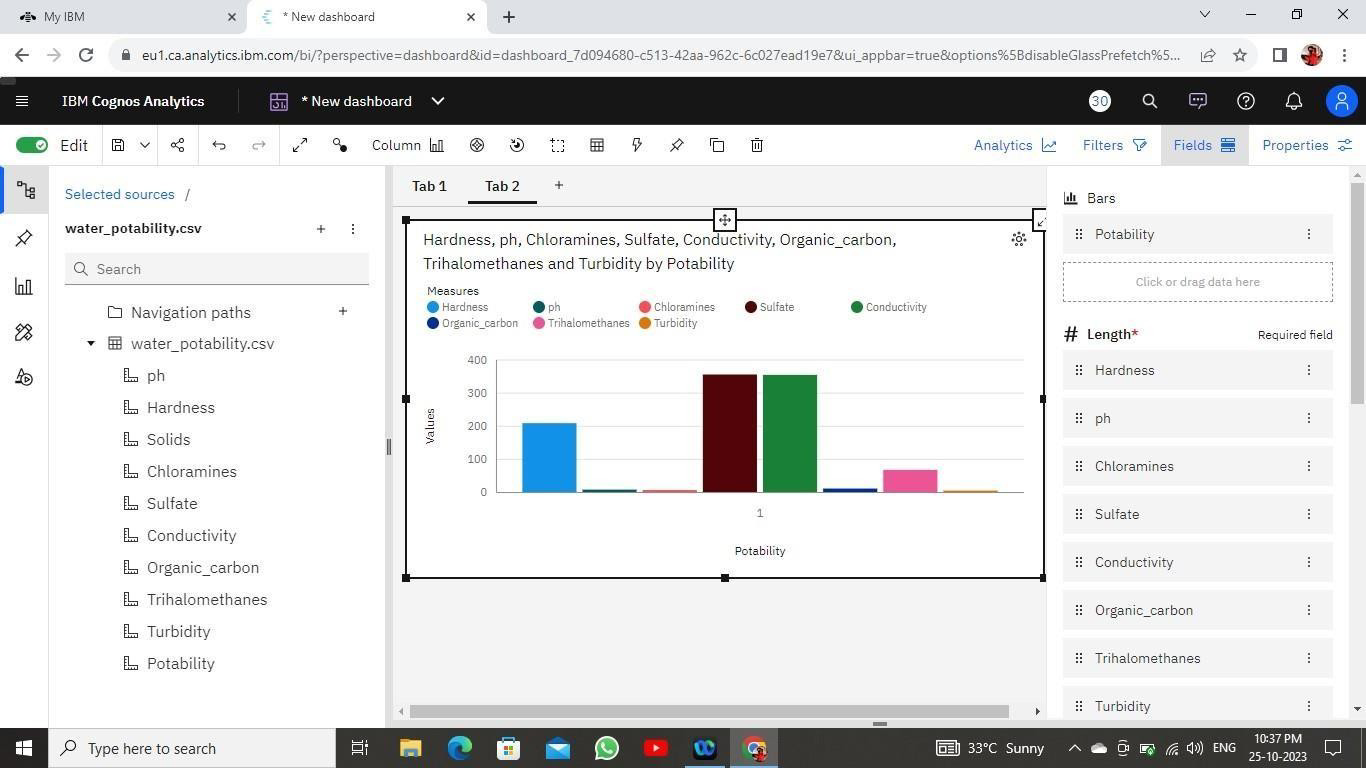


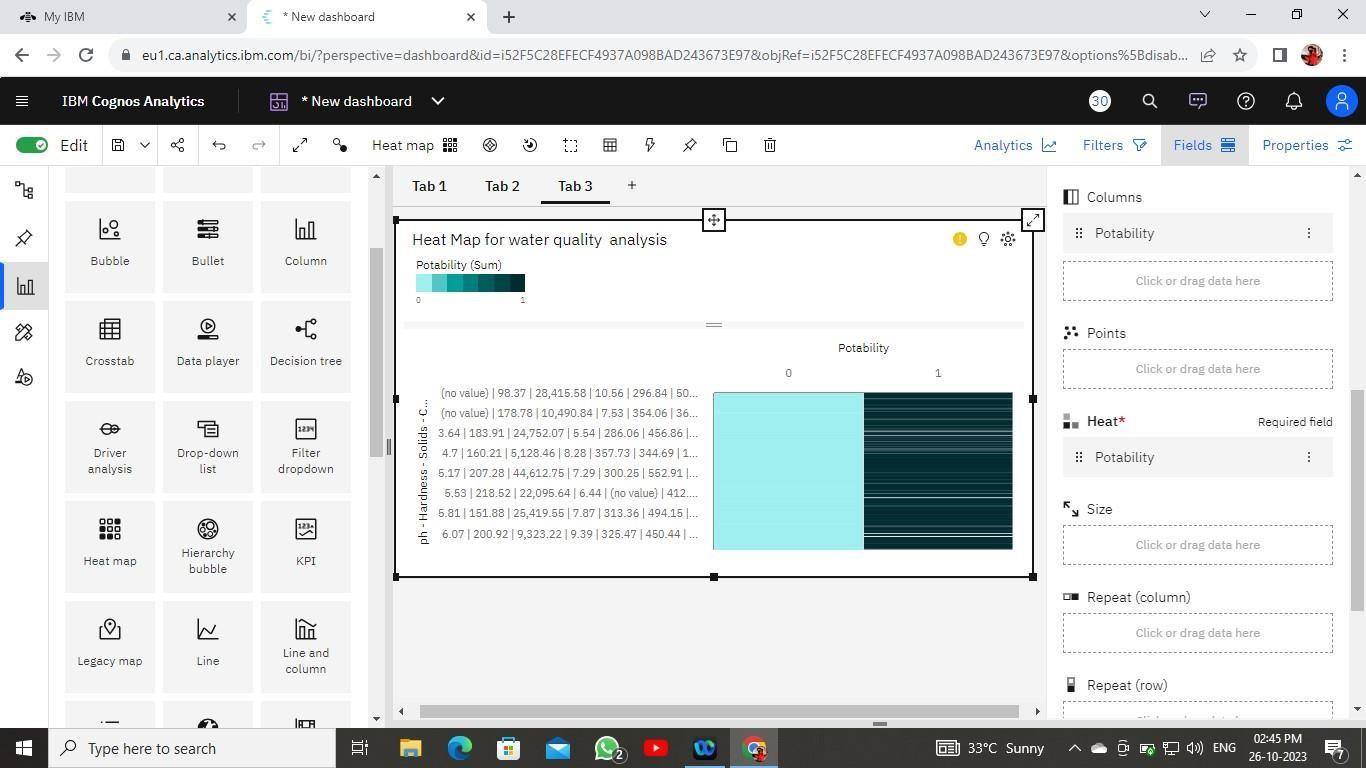
1. Representation of data’s of water parameter in box plot:

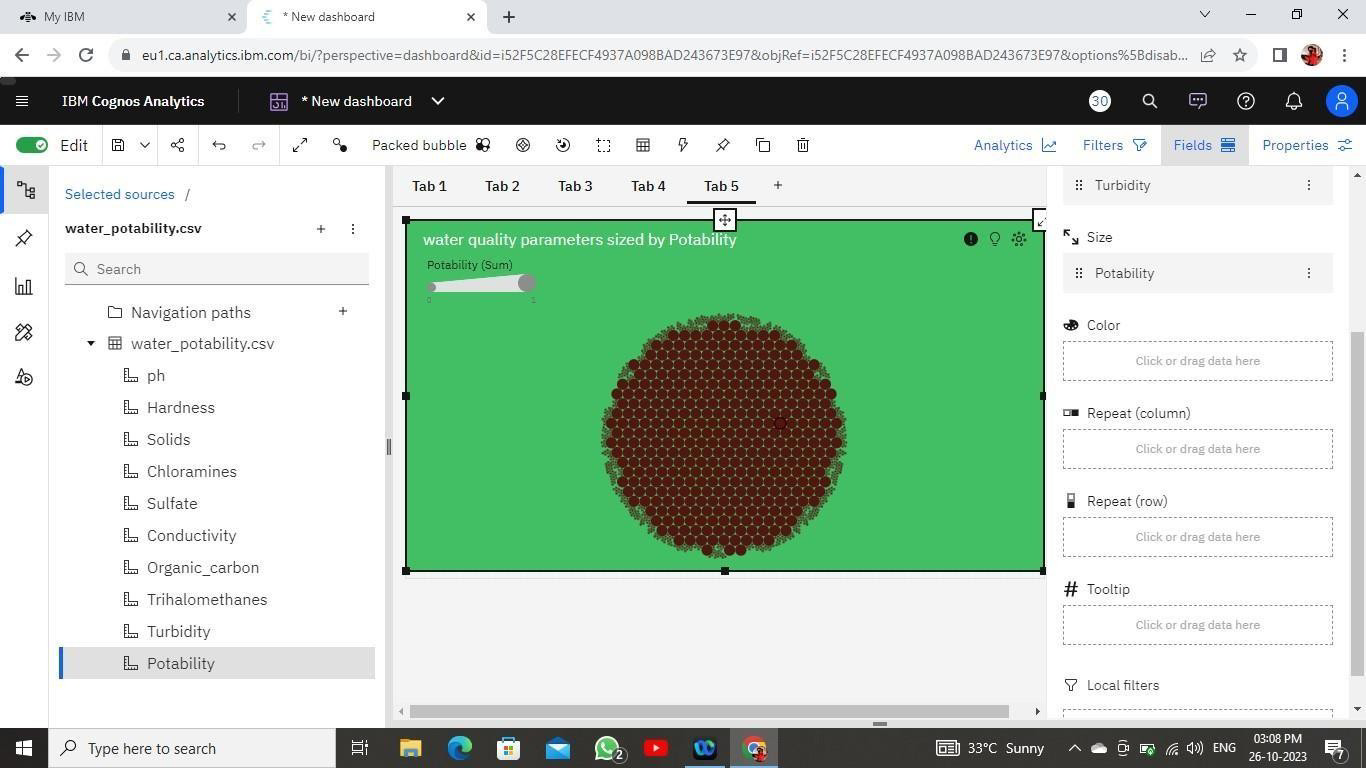


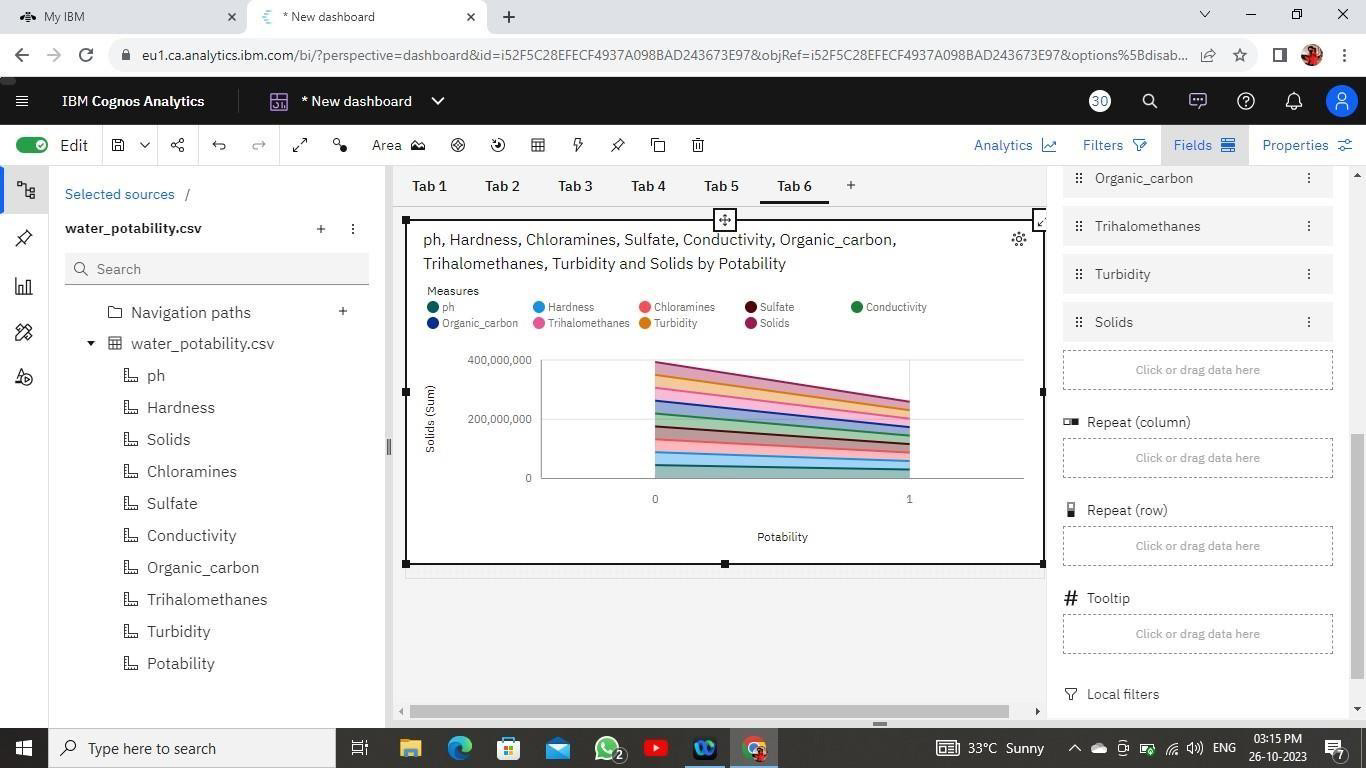
# DATA Visualisation Using IBM Cognos Analytics(Creation of dash board):

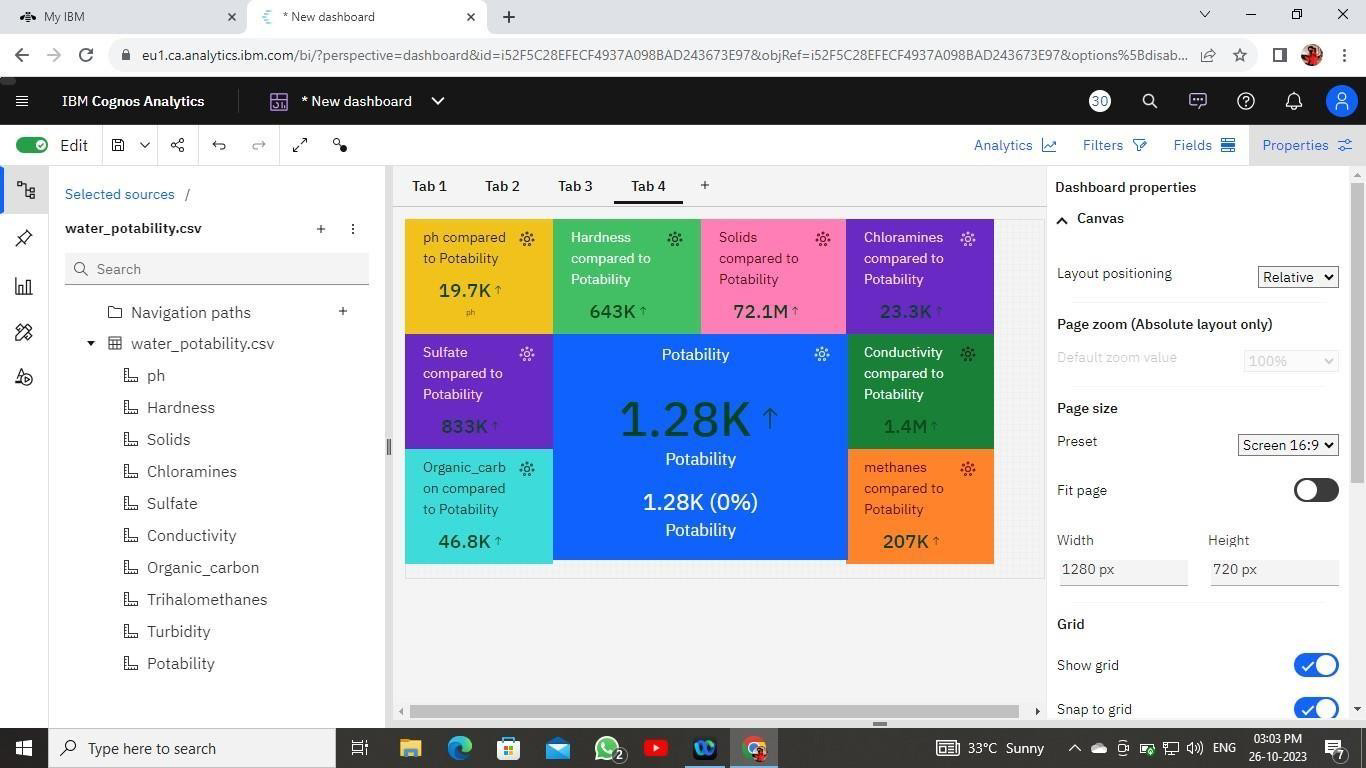
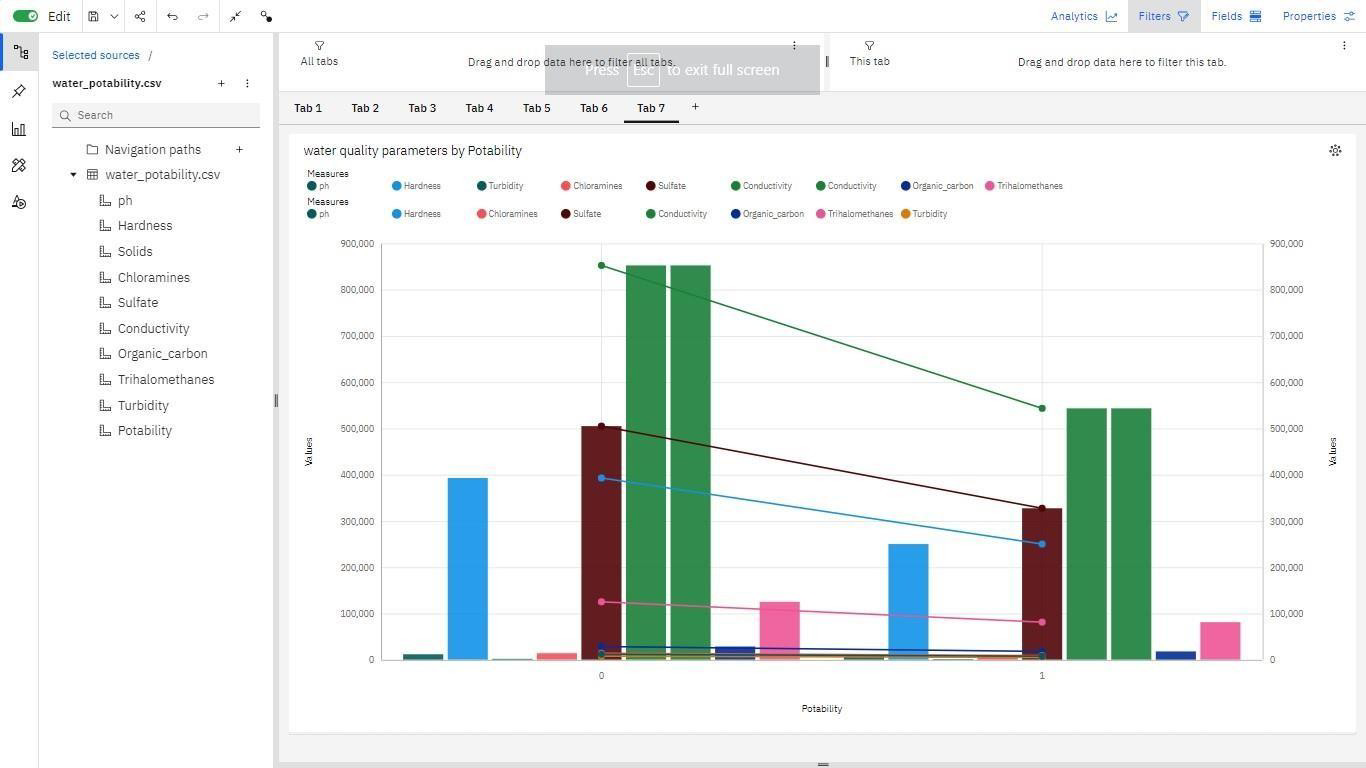












**Data Processing Procedure:**

To create a data visualization and perform machine learning analysis on a water quality dataset using Python libraries like pandas, NumPy, matplotlib, seaborn, and scikit-learn for decision tree classification, follow these steps:

**Development Phase 1- Decision Tree Classifier**

* 1. **Data Collection and Pre-processing:** Obtain the water quality dataset with relevant parameters. Import necessary Python libraries: pandas, NumPy, matplotlib, seaborn, and scikit- learn. Load and pre-process the dataset using pandas to handle missing values, outliers, and data cleaning.
  2. **Correlation Analysis:** Use pandas to calculate the correlation between different parameters in the dataset. Create correlation matrices and visualize them using heatmap plots from seaborn to identify relationships between variables.
  3. **Machine Learning Preparation:** Select the target variable (e.g., water quality class) and features (water quality parameters) for the machine learning model. Split the dataset into training and testing sets.
  4. **Decision Tree Classifier:** Train a decision tree classifier using the Decision Tree Classifier from scikit-learn. Fit the model to the training data and evaluate its performance on the testing data.
  5. **Model Evaluation:** Calculate performance metrics such as accuracy, precision, recall, and F1- score to assess the model's classification performance.
  6. **Visualization of Decision Tree:** Visualize the decision tree structure using tools provided by scikit-learn, such as the plot tree function.
  7. **Interpretation:** Analyse the decision tree to understand the rules and features that are most important for classification.

# Development Phase 2 - KNN Model with Hyperparameter Tuning:

1. **Model Accuracy:** The implementation of a KNN model in the second development phase enhances the ability to classify water quality data accurately. KNN can make predictions based on the similarity of data points, making it a valuable tool for categorizing water quality.
2. **Hyperparameter Tuning**: Hyperparameter tuning ensures that the KNN model performs optimally. By fine-tuning parameters such as the number of neighbours (K), distance metrics, and weighting functions, the model can be customized to deliver the best results for the specific water quality dataset.
3. **Predictive Capabilities**: The KNN model's predictive capabilities provide valuable insights into future water quality trends. By analysing historical data, it can identify patterns and forecast potential water quality issues, allowing for proactive measures to be taken, The KNN model's results can be integrated with IBM Cognos for visualization and reporting. This further enhances the overall water quality monitoring system, as stakeholders can access predictive and historical data through the same platform.

# Data setup and Data analysis technique:

To perform water quality analysis using IBM Cognos and Anaconda Jupyter, you'll need to follow several steps, including dataset acquisition, database setup, analysis techniques, and visualization methods. Here's a general overview of the process:

# Dataset Acquisition:

* + Obtain a water quality dataset from a reliable source, such as a government agency, research institution, or environmental organization. These datasets typically contain various parameters like pH, temperature, dissolved oxygen, turbidity, and contaminant concentrations (e.g., heavy metals, bacteria).

# Data Pre-processing and Database Setup:

* + Import the water quality dataset into a structured format compatible with your data analysis tools.
  + You may choose to store the data in a relational database or a data warehouse for efficient querying. IBM Db2, MySQL, or PostgreSQL can be used for this purpose.
  + Perform data cleaning, which includes handling missing values, correcting data entry errors, and ensuring data consistency.

# Analysis Techniques:

* + Utilize Anaconda Jupyter for data analysis. Anaconda provides a Python environment with essential libraries for data analysis and visualization, including Pandas, NumPy, Matplotlib, Seaborn, and more.
  + Apply statistical and machine learning techniques to analyse the water quality data. Some possible analyses include:
    - Descriptive statistics to understand the dataset's basic properties.
    - Time series analysis to detect trends and seasonality.
    - Correlation analysis to identify relationships between different water quality parameters.
    - Predictive modelling to forecast future water quality conditions.
    - Anomaly detection to find unusual events or pollution incidents.

# Visualization Methods:

* Data visualization is crucial for conveying insights to stakeholders and decision-makers.
* Use libraries such as Matplotlib and Seaborn to create various types of plots and charts, including:
  + Line plots to display trends over time.
  + Scatter plots for showing correlations between parameters.
  + Heatmaps to visualize relationships among multiple water quality parameters.
  + Histograms for distribution analysis.
  + Geographic maps to display spatial variations in water quality.
* Dashboards and reports can be created using IBM Cognos. IBM Cognos provides a platform for designing and sharing interactive reports and dashboards.
* Integrate Anaconda Jupyter with Cognos by exporting your analysis results as data sources or visualizations that can be embedded in Cognos reports and dashboards.

By combining the strengths of IBM Cognos and Anaconda Jupyter, you can perform comprehensive water quality analysis, generate meaningful insights, and present these insights effectively to stakeholders for informed decision-making.

# Deploying a big data analysis:

Deploying a big data analysis solution using IBM Cloud Databases and performing data analysis involves several steps. In this example, we'll use IBM Db2 as the database system and Python for data analysis. Here are the instructions:

**Step 1:** Create an IBM Cloud Account

If you don't have an IBM Cloud account, sign up for one at [IBM Cloud]

**Step 2:** Set Up an IBM Db2 Database on IBM Cloud

1. Log in to your IBM Cloud account.
2. In the IBM Cloud Dashboard, click on the "Create Resource" button.
3. In the catalog, search for "Db2" and select the "Db2 on Cloud" service.
4. Follow the prompts to set up the Db2 database, including selecting a plan, specifying credentials, and configuring the database instance.
5. Once the Db2 database is provisioned, note down the connection details, such as the hostname, port, database name, and credentials. You'll need these to connect to the database from your analysis environment.

**Step 3**: Prepare Your Data for Analysis

1. Gather or obtain your big data dataset. Ensure it's in a format compatible with your analysis tools, such as CSV, JSON, or Parquet.
2. If needed, clean and preprocess the data, including addressing missing values, removing duplicates, and handling data quality issues.

**Step 4:** Install Required Software

1. Install Anaconda or Miniconda on your local machine or server where you'll perform data analysis. Anaconda provides a convenient Python environment for data analysis. You can download it from [Anaconda's website](https:/[/www.ana](http://www.anaconda.com/products/individual))c[onda.com/products/individual).](http://www.anaconda.com/products/individual))
2. Create a new Conda environment and install the necessary Python libraries for data analysis. You can use libraries like Pandas, NumPy, Matplotlib, Seaborn, and Jupyter Notebooks for data manipulation, analysis, and visualization.
3. Install additional libraries if your analysis requires them. For database connectivity, you'll need the `ibm\_db` or `ibm\_db\_sa` package to connect to the Db2 database.

**Step 5:** Connect to the IBM Db2 Database

In your Jupyter Notebook, you can use the `ibm\_db` or `ibm\_db\_sa` package to connect to the Db2 database using the credentials you obtained earlier:

Step 6: Perform Data Analysis

You can now use Python in your Jupyter Notebook to perform data analysis. Load your dataset into a Pandas Data Frame, run SQL queries, and utilize data analysis techniques as required for your specific use case. Visualize your results using Matplotlib, Seaborn, or other visualization libraries.

Step 7: Save and Share Your Analysis

Once you've completed your analysis, save your Jupyter Notebook and any visualization outputs. You can also create a report or dashboard using tools like IBM Cognos if necessary. Share your insights and findings with stakeholders or team members.

By following these steps, you can deploy a big data analysis solution using IBM Cloud Databases, connect to the database, and perform data analysis using Python in an Anaconda environment. This approach allows you to leverage IBM Cloud's infrastructure and services for database management while conducting robust data analysis on your dataset.

**Conclusion:**

The project's objectives were to comprehensively evaluate water quality, assess its potability, detect deviations from established standards, and explain the interrelationships among key parameters. The team utilized data visualization techniques to extract meaningful insights from the water quality dataset.

Key highlights of the project include:

1. Data Visualization: The team used Python's Matplotlib library within the Jupyter Anaconda environment to create informative and visually appealing data visualizations. They explored various plot types, customized aesthetics, and added context to the visualizations to effectively communicate key trends, patterns, and relationships in the water quality data.
2. Machine Learning Analysis: The project included a machine learning phase involving a Decision Tree Classifier to classify water quality. The team pre- processed data, performed correlation analysis, and evaluated the model's performance. The Decision Tree Classifier provided insights into the rules and features important for classification.
3. KNN Model with Hyperparameter Tuning: In the second phase, the project introduced a K-Nearest Neighbours (KNN) model to enhance classification accuracy. Hyperparameter tuning was performed to optimize the KNN model. This phase emphasized the predictive capabilities of the model, allowing for the identification of water quality trends and forecasting potential issues.
4. Data Processing Procedure: The project's data analysis was accomplished using Python libraries such as Pandas, NumPy, Matplotlib, Seaborn, and scikit- learn. The team followed a structured process that involved data collection, pre- processing, correlation analysis, machine learning, and model evaluation. Additionally, the project outlined the use of IBM Cognos Analytics for creating interactive dashboards and reports to visualize and share the analysis results. This approach allowed for effective communication of insights to stakeholders. In summary, this water quality analysis project combined data visualization and machine learning techniques to assess and classify water quality. It not only provided a deeper understanding of water quality parameters but also offered the potential for proactive measures to address water quality issues. The use of IBM Cloud Databases and Anaconda Jupyter further enhanced the data analysis capabilities.