DEPARTMENT OF COMPUTER SCIENCE,
FACULTY OF SCIENCE,
THE UNIVERSITY OF IBADAN, NIGERIA.

CSC499 INDUSTRIAL TRAINING

ADIM SOLOMON CHIMAOBI

MATRIC NO: 222455

A TECHNICAL REPORT OF THE WORK DONE DURING MY INDUSTRIAL TRAINING UNDERTAKEN

AT

PEAK INFOTECH SYSTEM, OFF POLYTECHNIC ROAD, SANGO, IBADAN, OYO STATE

IN PARTIAL FULFILMENT OF THE REQUIREMENTFOR THE AWARD OF A BACHELOR OF SCIENCE (B.Sc.) DEGREE IN COMPUTER SCIENCE

24^{TH} January – 28^{TH} June, 2024.

Department of Computer Science
The University of Ibadan,
Ibadan, Oyo State.
July 2024.

The Director,

Industrial Training Coordinating Centre,

The University of Ibadan,

Ibadan,

Oyo State.

Dear Sir,

LETTER OF SUBMISSION OF SIWES REPORT

I, ADIM SOLOMON CHIMAOBI, have completed my training and am writing to submit my work report for evaluation per your instructions formally. This report provides a comprehensive account of the experiences I gained during my SIWES training.

I certify that I am the author of this work report. The training took place from January 24th to June 28th, 2024, at Peak Infotech Systems, located off Polytechnic Road, Sango, Ibadan, Oyo State.

Thanks

Yours sincerely
ADIM SOLOMON
222455

CERTIFICATION

I, ADIM SOLOMON CHIMAOBI, enrolling under the number 222455, certify that I completed the SIWES Program at PEAK INFOTECH Systems (Polytechnic Road, Sango Ibadan, Oyo state).
Moreover, I am the author of the report, writing it using the practical knowledge I acquired during the training to the best of my ability.

_____ _____

Student's Name Sign & Date

ACKNOWLEDGEMENT

I extend my heartfelt gratitude to the Almighty for His continuous blessings and guidance throughout my academic journey. Furthermore, I would like to express my profound appreciation to my parents, Mr. Adim Odoemenam and Mrs. Adim Nkechinyere, for their unwavering financial and moral support, particularly during my SIWES program.

I am also deeply thankful to the entire staff at Peak Infotech Systems for their invaluable assistance during my learning phase. I acknowledge and appreciate my supervisor, ENGR. Afenifere Yusuf, for his consistent encouragement and unwavering support during my tenure there.

Finally, I want to express my sincere gratitude to the University of Ibadan, the Industrial Training Coordinating Centre (ITCC), and the Department of Computer Science for giving me the opportunity to gain invaluable corporate experience.

God bless you all (Amen)

ABSTRACT

The Students Industrial Work Experience Scheme (SIWES) was established to expose students to the industrial environment and equip them with essential professional skills. This program aims to enable recent graduates to make meaningful contributions to the nation's development in their respective fields. During the early stages of science and technology education in Nigeria, graduates often needed more practical knowledge and work experience.

This technical report comprehensively accounts for my SIWES experience at Peak Infotech Systems. The report commences with an overview of the company's history, location, and mission. It delves into the company's goals and underscores the significance of technology in relation to the company's history and mission.

Following this introduction, the report provides an extensive overview of the various technologies I was exposed to during my SIWES placement. Subsequently, it is divided into distinct chapters, each dedicated to a specific aspect of my I.T. experience, encompassing software, hardware, and other technologies I had the privilege to engage with.

TABLE OF CONTENTS

CHAPTER 1

1.0 INTRODUCTION

The Student Industrial Work Experience Scheme (SIWES) provides students with practical work experience during their academic years. Also known as the Student Intern Work Experience Scheme, SIWES is a globally recognized program implemented in countries such as Japan, Australia, Europe, and Africa. In Europe, it is often called "sandwich education" and is commonly referred to as the "Student Work Experience Scheme." This program typically lasts six months and involves students working in various establishments relevant to their fields of study.

In Nigeria, SIWES was established by Decree 47 of 1972, enacted by General Yakubu Gowon's military council. This decree's primary goal was to reduce foreign participation in Nigeria's economic activities by promoting the use of locally skilled labor in various sectors of the economy.

According to the national education policy, students pursuing diplomas or degrees must meet specific requirements, which may vary between educational institutions. However, participation in the Industrial Work Experience Scheme (SIWES) is a common requirement for all institutions. This program is carefully designed to familiarize students with the industrial work environment through practical job training. Successful completion of SIWES is a mandatory prerequisite for obtaining a diploma or degree in Nigeria

1.1 BRIEF HISTORY OF SIWES

In 1973, the Federal Government of Nigeria established the Student Industrial Work Experience Scheme (SIWES) to enhance the nation's technological, physical, and social skills. This program allows students to gain practical experience in their fields of study before earning their Bachelor of Science (BSc) degrees.

1.2 AIMS AND OBJECTIVES OF SIWES

SIWES Offer students in higher education institutions the chance to acquire practical knowledge and hands-on experience within their approved academic programs while facilitating interactions with experts in their chosen fields. Prepare students for the post-graduation workplace by familiarizing them with work processes and imparting the necessary skills to operate machinery and equipment.

1.3 ROLES OF STUDENTS

Before starting the attachment, participate in the SIWES orientation program.

- Follow the rules and regulations of the hosting establishment.
- Arrange suitable accommodations for the duration of the attachment.
- Keep a comprehensive logbook, recording all training activities and tasks.
- Complete the Student's Evaluation of Industrial Work Experience (SPEI) using FORM 8 from ITF and get the employer's endorsement before submitting it to ITF.

- Engage actively in all tasks and projects, showing a commitment to learning and professional growth.
- Communicate regularly with your academic supervisor to update them on your progress and address any challenges.
- Attend all meetings and training sessions organized by the hosting establishment.
- Maintain professional conduct and ethics, positively representing your educational institution.
- Continuously seek feedback from supervisors and colleagues to improve your performance and skills.

1.4 OBJECTIVES OF THE REPORT

The objectives of this SIWES report are:

- To provide a comprehensive overview of the tasks and experiences encountered during my four-month industrial training.
- To fulfil the requirements for obtaining a national diploma in computer science.
- To contribute to the collective knowledge base and enhance the writer's understanding of similar roles.

1.5 THE LOGBOOK

The logbook provided by the institution to the attached student served as the primary tool for documenting daily activities throughout the attachment period. It underwent regular checks and received endorsements from both industry-based and institution-based supervisors and from the ITF during supervision sessions.

.

1.6 HISTORICAL BACKGROUND OF PEAK INFOTECH SYSTEM

Engr Raheem Ogundowole founded PEAK INFOTECH SYSTEMS in 2015. The firm, located off Polytechnic Road, Sango, Ibadan, is a tech solutions firm that designs, develops, and maintains applications, networks, and digital tools.

Initially started by a team of just three people, the company has since grown to include over 14 staff members, not including temporary staff and interns.

1.7 ORGANIZATIONAL DEPARTMENTS AND THEIR FUNCTIONS

- **Networking Department**: Responsible for designing, configuring, and managing network infrastructures, ensuring data flows efficiently and securely.
- **Security Department:** Focused on safeguarding the network and data from cyber threats and unauthorized access, implementing robust security measures.
- **Technical Support Department:** Providing timely assistance and troubleshooting to clients or internal users with network-related issues, ensuring smooth operations.
- **Sales and Marketing Department**: Responsible for acquiring clients, promoting networking services, and managing client relationships to drive business growth.

- **Solutions Department**: Responsible for acquiring clients, promoting I.T. solutions services, and managing client relationships to drive business growth.

1.8 SERVICES RENDERED BY THE COMPANY

- Network Design and Implementation
- It Support and Helpdesk Services
- Cybersecurity Solutions
- Software Development and Customization
- Cloud Services and Management
- Data Science and Analysis
- Robust and Integrated Business Solutions

1.9 MISSION AND VISIONS

1.9.1 MISSION

At PEAK INFOTECH SYSTEMS, our mission is to empower individuals and businesses through innovative tech solutions that streamline processes, enhance productivity, and drive growth.

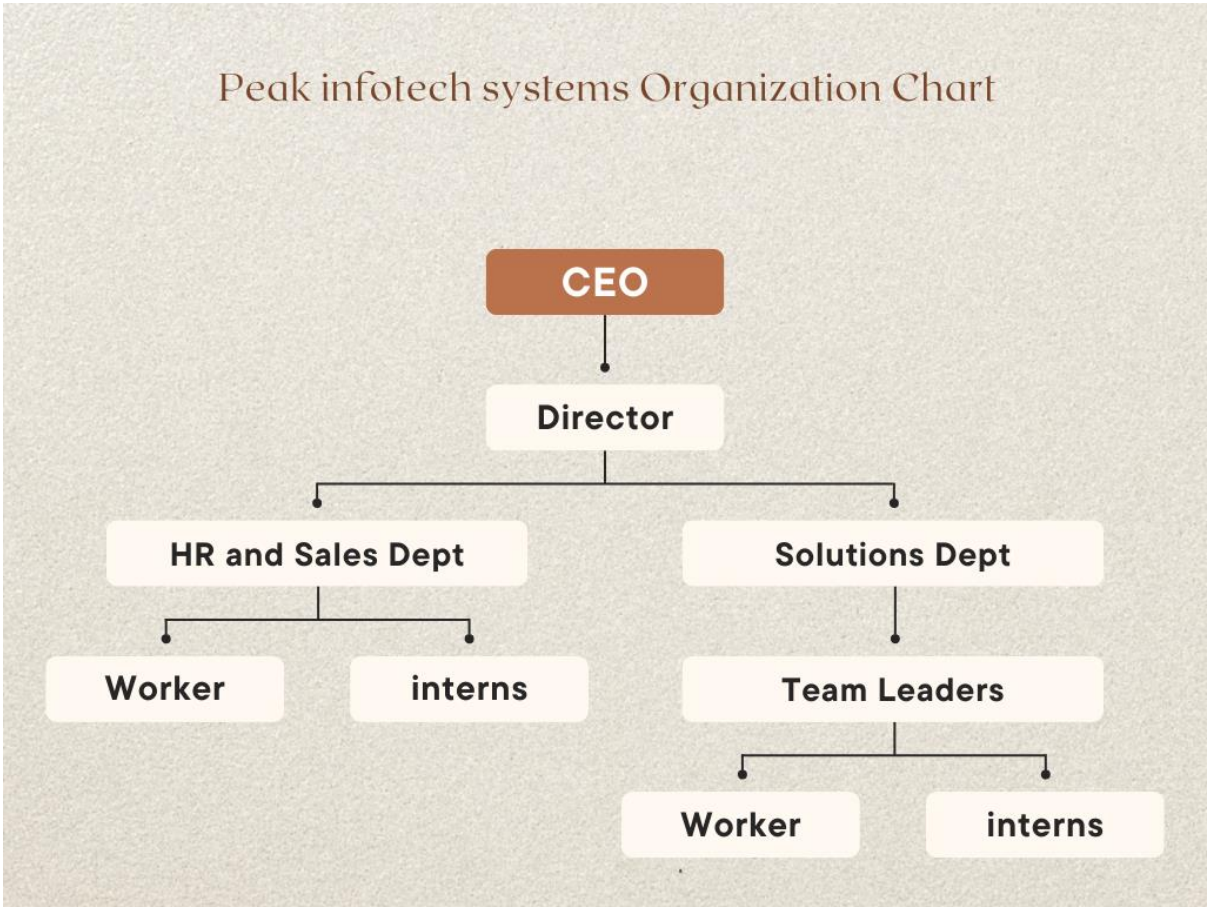
We are dedicated to delivering high-quality applications, networks, and digital tools tailored to meet the unique needs of our clients while fostering a culture of continuous improvement and excellence

1.9.2 VISION

Making you a better you

In order to achieve our vision above, we constantly strive to be the foremost provider of cutting-edge solutions, recognized for our unwavering commitment to client success, technological excellence, and global impact.

1.10 ORGANOGRAM OF THE ORGANIZATION



CHAPTER 2

2.0 DESCRIPTION OF TECHNOLOGIES/TOOLS USED

Numerous projects can be conceptualized and executed in the era of advancing technologies. Understanding the right technology for different projects plays a pivotal role in ensuring the software's effectiveness, dependability, and scalability.
This chapter delves into a range of technologies that I had the opportunity to acquaint myself with during my Industrial Training, enabling me to undertake diverse projects. The chapter is organized into the following subtopics.

TOOLS USED FOR DATA ANALYSIS

- Microsoft Excel
- Power Bi
- Rapid Miner
- Talend
- Tableau
- Knime
- Statistical Analysis System (SAS)
- Python
- R
- Apache spark
- Matplot lib
- Pandas
- Numpy
- Scipy
- Ggplot
- Seaborn
- Dplyr
- SQL
- Jupyter notebook
- Google collab
- Github

2.1 MICROSOFT EXCEL

Microsoft Excel is a powerful tool widely used in data science for various tasks, including data collection, cleaning, analysis, and visualization. Despite the advent of more sophisticated data science tools, Excel remains valuable due to its accessibility, user-friendly interface, and robust feature set.

2.1.1 Key Features and Application in Data Science

Data Collection and Entry:
Excel allows for easy data entry and organization. Users can manually input data or import data from various sources like CSV files, databases, and web pages.

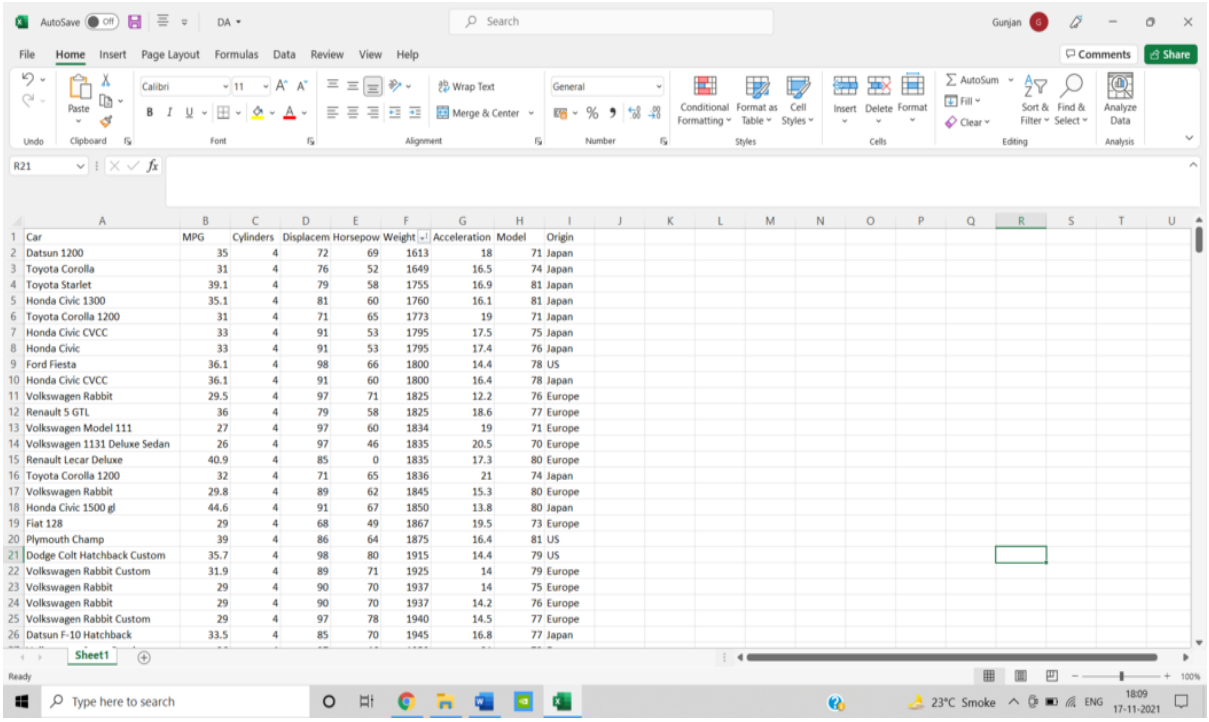
Data Cleaning:
Excel provides numerous functions and tools to clean and preprocess data, such as text manipulation functions, find and replace, data validation, and conditional formatting.

Data Analysis:
Formulas and Functions: Excel supports a wide range of mathematical, statistical, and logical functions, which are essential for data analysis.
Pivot Tables: Pivot tables allow users to summarize, analyze, explore, and present data insights dynamically.

Data Analysis Toolpak: This add-in provides tools for performing complex data analyses, including regression, histograms, and descriptive statistics.
Data Visualization:
Excel offers various chart types (e.g., bar, line, pie, scatter plots) to visualize data. Customizable options help in creating clear and informative visual representations of data.
Conditional Formatting: This feature helps highlight key data points and trends by applying formats to cells that meet specific criteria.
Automation:

Macros: Excel supports the creation of macros to automate repetitive tasks, enhancing efficiency and consistency in data handling.
Integration with Other Tools:
Excel can integrate with various data sources and other software tools, making it a flexible option for data scientists who need to combine data from multiple platforms.

2.1.2 Conclusion
Microsoft Excel is a crucial tool in the data science toolkit, especially for tasks that involve data manipulation, preliminary analysis, and visualization. Its ease of use and widespread availability make it an essential starting point for many data science projects.



Car	MPG	Cylinders	Displacem	Horsepow	Weight	Acceleration	Model	Origin
Datsun 1200	35	4	72	69	1613	18	71	Japan
Toyota Corolla	31	4	76	52	1649	16.5	74	Japan
Toyota Starlet	39.1	4	79	58	1755	16.9	81	Japan
Honda Civic 1300	35.1	4	81	60	1760	16.1	81	Japan
Toyota Corolla 1200	31	4	71	65	1773	19	71	Japan
Honda Civic CVCC	33	4	91	53	1795	17.5	75	Japan
Honda Civic	33	4	91	53	1795	17.4	76	Japan
Ford Fiesta	36.1	4	98	66	1800	14.4	78	US
Honda Civic CVCC	36.1	4	91	60	1800	16.4	78	Japan
Volkswagen Rabbit	29.5	4	97	71	1825	12.2	76	Europe
Renault 5 GTL	36	4	79	58	1825	18.6	77	Europe
Volkswagen Model 111	27	4	97	60	1834	19	71	Europe
Volkswagen 1131 Deluxe Sedan	26	4	97	46	1835	20.5	70	Europe
Renault Lecar Deluxe	40.9	4	85	0	1835	17.3	80	Europe
Toyota Corolla 1200	32	4	71	65	1836	21	74	Japan
Volkswagen Rabbit	29.8	4	89	62	1845	15.3	80	Europe
Honda Civic 1500 gl	44.6	4	91	67	1850	13.8	80	Japan
Fiat 128	29	4	68	49	1867	19.5	73	Europe
Plymouth Champ	39	4	86	64	1875	16.4	81	US
Dodge Colt Hatchback Custom	35.7	4	98	80	1915	14.4	79	US
Volkswagen Rabbit Custom	31.9	4	89	71	1925	14	79	Europe
Volkswagen Rabbit	29	4	90	70	1937	14	75	Europe
Volkswagen Rabbit	29	4	90	70	1937	14.2	76	Europe
Volkswagen Rabbit Custom	29	4	97	78	1940	14.5	77	Europe
Datsun F-10 Hatchback	33.5	4	85	70	1945	16.8	77	Japan

An interface of an Excel spreadsheet while working on data

2.2 POWER BI

Power B.I. is a powerful business analytics service by Microsoft that provides interactive visualizations and business intelligence capabilities. It is widely used in data science to transform raw data into meaningful insights through dashboards and reports.

2.2.1 Key Features and Uses in Data Science

Data Connectivity:

- **Wide Range of Data Sources**: Power B.I. can connect to a variety of data sources, including databases, spreadsheets, cloud services, and web APIs. This flexibility allows data scientists to integrate data from multiple platforms.

Data Transformation:

- **Power Query**: Power B.I. includes Power Query, a robust tool for data transformation and preparation. Users can clean, reshape, and combine data from different sources before analysis.

Data Modeling:

- **DAX (Data Analysis Expressions)**: Power B.I. uses DAX, a formula language, to create custom calculations and measures, enabling complex data modelling and analysis.

Visualization:

- **Interactive Dashboards**: Power B.I. allows users to create highly interactive and customizable dashboards. Visualizations such as charts, graphs, maps, and tables help users explore and present data insights effectively.
- **Real-Time Dashboards**: Power B.I. supports real-time data streaming, enabling the creation of dashboards that update live as data is received.

Reporting:

- **Paginated Reports**: Power B.I. can generate paginated reports that can be printed or shared as PDF documents, complementing interactive dashboards.

A.I. and Machine Learning Integration:

- **A.I. Insights**: Power B.I. integrates with Azure Machine Learning, allowing users to incorporate machine learning models directly into their Power B.I. reports for predictive analytics.
- **Cognitive Services**: Built-in A.I. capabilities, such as text analytics and image recognition, can be applied to data within Power B.I.

Collaboration and Sharing:

- **Power B.I. Service**: Users can publish reports and dashboards to the Power B.I. service, making sharing insights with colleagues and stakeholders easy.
- **Collaboration**: Features like workspaces, content packs, and app workspaces facilitate collaboration and report distribution across teams and organizations.

2.2.2 Applications

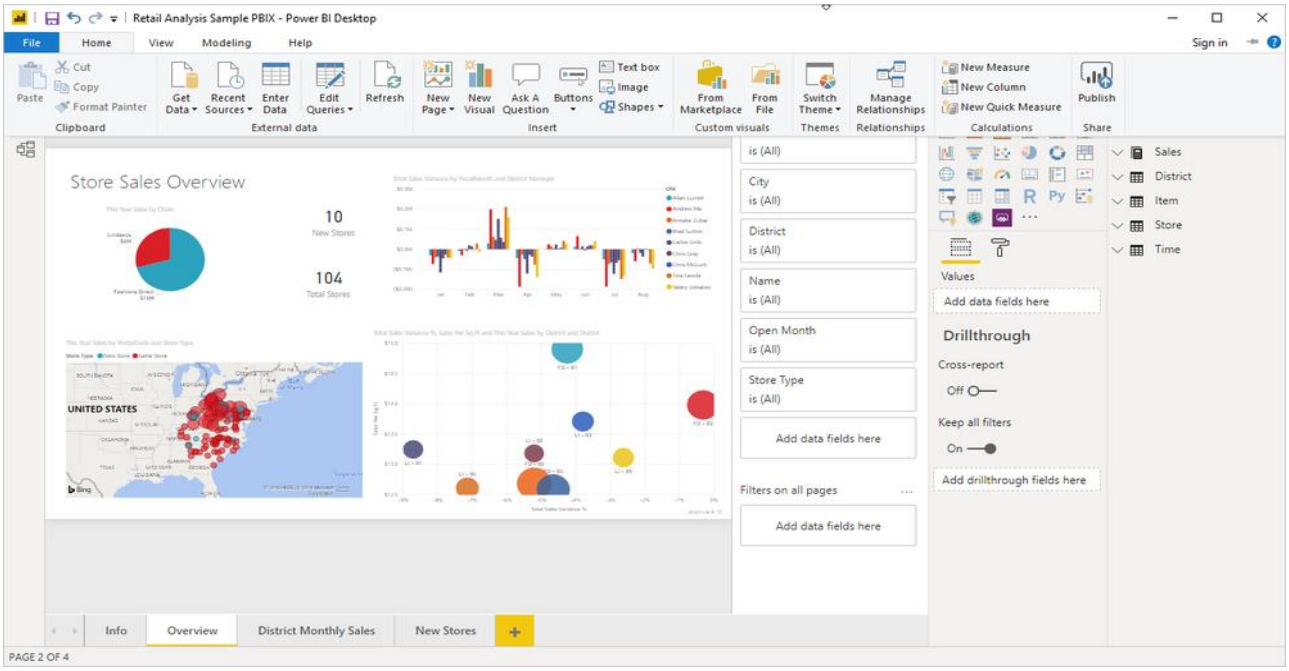
- **Business Intelligence**: Used extensively to create interactive dashboards and reports to support decision-making processes in organizations.
- **Financial Analysis**: This helps visualize and analyze financial data, including budget tracking, sales forecasting, and profitability analysis.
- **Customer Insights**: Provides insights into customer behaviour and preferences, aiding in targeted marketing and improving customer service.

2.2.3 Limitations

While Power B.I. is highly effective for visualization and reporting, it may not be suitable for handling massive datasets or complex data science workflows that require extensive machine learning or statistical analysis. In such cases, integration with tools like Python, R, or specialized machine learning platforms may be necessary.

2.2.4 Conclusion

Power B.I. is valuable in the data science toolkit, particularly for data visualization, reporting, and business intelligence. Its ease of use, extensive data connectivity, and powerful visualization capabilities make it famous for transforming data into actionable insights.



A Power B.I. interface displaying different data

2.3 RAPID MINER

RapidMiner is a powerful, open-source data science platform designed to streamline data preparation, machine learning, and predictive analysis. Due to its user-friendly interface and extensive capabilities, it is widely used in academia and industry.

2.3.1 Key Features and Uses in Data Science

Visual Workflow Design:

RapidMiner provides a drag-and-drop interface for designing data processing workflows, making it accessible to users without extensive programming knowledge.

Data Preparation:

The platform includes data cleaning, transformation, normalization, and aggregation tools, which are essential for preparing data for analysis.

Machine Learning:

RapidMiner supports a broad range of machine learning algorithms, including classification, regression, clustering, and association rules.

It offers automated machine learning (AutoML) features to streamline the model selection and optimization process.

Data Visualization:

RapidMiner provides various visualization tools to explore and understand data and present analytical results effectively.

Integration and Extensions:

The platform integrates with various data sources, such as databases, cloud storage, and APIs, allowing for seamless data import and export.

RapidMiner supports extensions and plugins, including those for Python and R, enabling advanced customization and functionality.

Deployment:

RapidMiner facilitates the deployment of predictive models into production environments, allowing businesses to operationalize their data science workflows.

Community and Support:

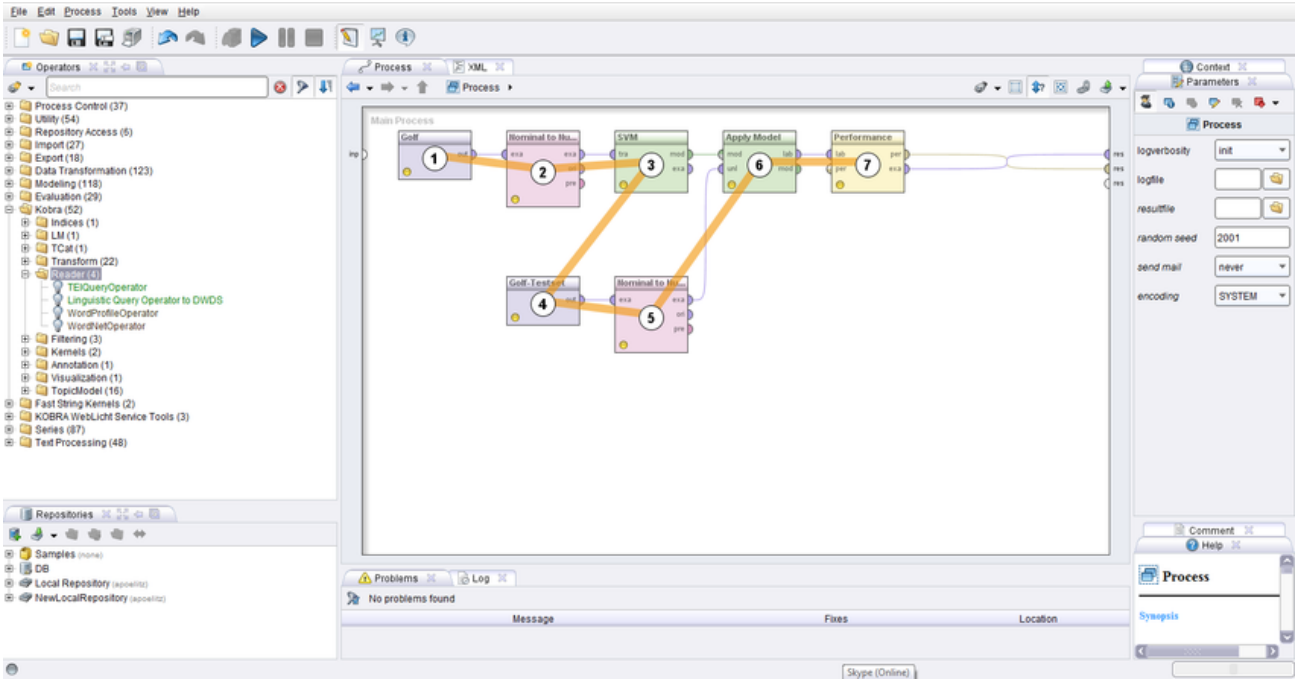
Being open-source, RapidMiner has a robust community of users and developers who contribute to its development and provide support through forums and tutorials.

2.3.2 Applications

- **Predictive Maintenance**: Used in industries like manufacturing to predict equipment failures and schedule maintenance.
- **Customer Analytics**: Helps businesses understand customer behaviour and preferences, enabling targeted marketing campaigns.
- **Fraud Detection**: Assists financial institutions in identifying and mitigating fraudulent activities.

2.3.3 Conclusion

RapidMiner is a comprehensive tool for data science, offering ease of use, flexibility, and powerful features for data preparation, machine learning, and model deployment. Its ability to integrate with other tools and its strong community support make it a valuable asset in a data scientist's toolkit.



Rapid Miner interface

2.4 TALEND

Talend is an open-source data integration platform widely used in data science for its robust ETL (Extract, Transform, Load) capabilities. It helps data scientists efficiently handle large volumes of data from various sources, ensuring data quality and consistency.

2.4.1 Key Features and Uses in Data Science

Data Integration

ETL Processes: Talend excels at ETL, enabling users to extract data from multiple sources, transform it as needed, and load it into target systems. This process is essential for preparing data for analysis.

Connectors: Talend provides numerous connectors to integrate data from databases, cloud services, APIs, flat files, and more, ensuring comprehensive data integration.

Data Quality:

Data Profiling: Talend includes tools for data profiling, which help in assessing data quality by identifying inconsistencies, duplicates, and anomalies.

Data Cleansing: Built-in data cleansing features enable users to standardize, deduplicate, and enrich data, ensuring high-quality data for analysis.

Data Transformation:

Graphical Interface: Talend's drag-and-drop interface makes designing complex data transformation workflows easy without extensive coding.

Custom Components: Users can create custom components and transformations using Java, extending Talend’s functionality to meet specific data processing needs.

Big Data Support:

Big Data Integration: Talend supports big data technologies like Hadoop, Spark, and NoSQL databases, making it suitable for handling large-scale data processing tasks.
Real-Time Data Processing: Talend can process data in real time, which is crucial for applications that require immediate data insights.
Cloud Integration:
Cloud Services: Talend integrates with major cloud platforms such as AWS, Google Cloud, and Azure, enabling seamless data integration and management in cloud environments.
Data Migration: The platform facilitates smooth data migration between on-premises systems and cloud services, supporting hybrid data architectures.

Machine Learning Integration:
Machine Learning Components: Talend offers components for integrating with machine learning libraries and platforms like Apache Spark MLlib, enabling data scientists to incorporate machine learning models into their data workflows.
Collaboration and Governance:

Collaborative Platform: Talend provides features for team collaboration, allowing multiple users to work on data integration projects simultaneously.

Data Governance: The platform includes data governance tools to ensure compliance with data regulations and policies and enhance data security and management.

2.4.2 Applications

- **Business Intelligence**: Talend integrates and prepares data for B.I. tools, ensuring that businesses have accurate and timely data for decision-making.
- **Data Warehousing**: It helps build and maintain data warehouses by automating ETL processes and ensuring data quality.
- **Big Data Analytics**: Talend's support for big data technologies makes it suitable for large-scale data analytics projects.

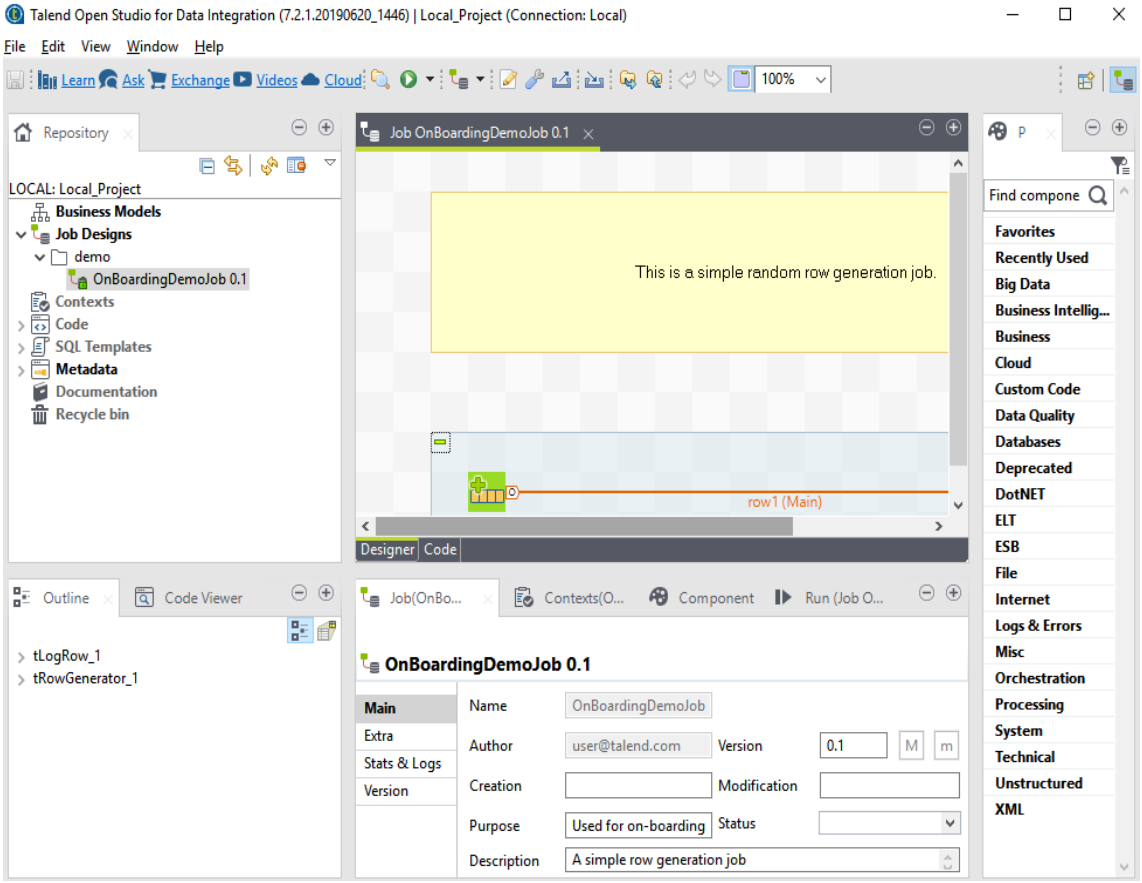
2.4.3 Limitations

While Talend is highly effective for data integration and transformation, it may require a learning curve for users new to ETL processes or its specific interface. Additionally, complex customizations necessitate some programming knowledge.

2.4.4Conclusion

Talend is a comprehensive data integration tool that plays a crucial role in data science by ensuring efficient data extraction, transformation, and loading. Its ability to handle big data, real-time processing, and cloud

integration makes it valuable for data scientists working on diverse data projects.



Talend Open Studio User Interface

2.5 TABLEAU

Tableau is a powerful data visualization tool widely used in data science. It transforms raw data into interactive and shareable dashboards and reports. Its user-friendly interface and robust capabilities make it a preferred choice for data analysis and visualization.

2.5.1 Key Features and Uses in Data Science

Data Connectivity:

Wide Range of Data Sources: Tableau can connect to various data sources, including databases (SQL, NoSQL), spreadsheets, cloud services (AWS, Google Cloud), and web data connectors, enabling seamless data integration.

Data Visualization:

Interactive Dashboards: Tableau allows users to create interactive and dynamic dashboards that can visualize Data through charts, graphs, maps, and other visual formats. Users can filter, drill down, and explore data interactively.

Advanced Visual Analytics: The tool supports advanced visual analytics such as trend lines, forecasting, clustering, and heat maps, which help in deriving insights from data.

Ease of Use:

Drag-and-Drop Interface: Tableau's intuitive drag-and-drop interface makes it accessible to users without extensive technical skills, enabling them to create complex visualizations quickly.

Customization: Users can customize visualizations with various formatting options, creating tailored and visually appealing reports.

Data Preparation:

Tableau Prep: This feature helps clean, shape, and combine data from multiple sources before analysis, ensuring that the data is accurate and ready for visualization.

Real-Time Data Processing: Tableau supports real-time data integration and visualization, which is crucial for applications that require up-to-date insights.

Collaboration and Sharing:

Tableau Server and Tableau Online: These platforms allow users to publish and share dashboards securely with colleagues and stakeholders. They support collaborative analytics and decision-making.

Embed Dashboards: Dashboards can be embedded into web applications, portals, or presentations, making it easy to share insights across different platforms.

Integration with Data Science Tools:

R and Python Integration: Tableau can integrate with R and Python, allowing users to leverage advanced statistical and machine learning models within their visualizations. This integration enhances Tableau's analytical capabilities.

2.5.2 Applications

Business Intelligence: Tableau is extensively used to create B.I. dashboards that provide actionable insights and support data-driven decision-making.

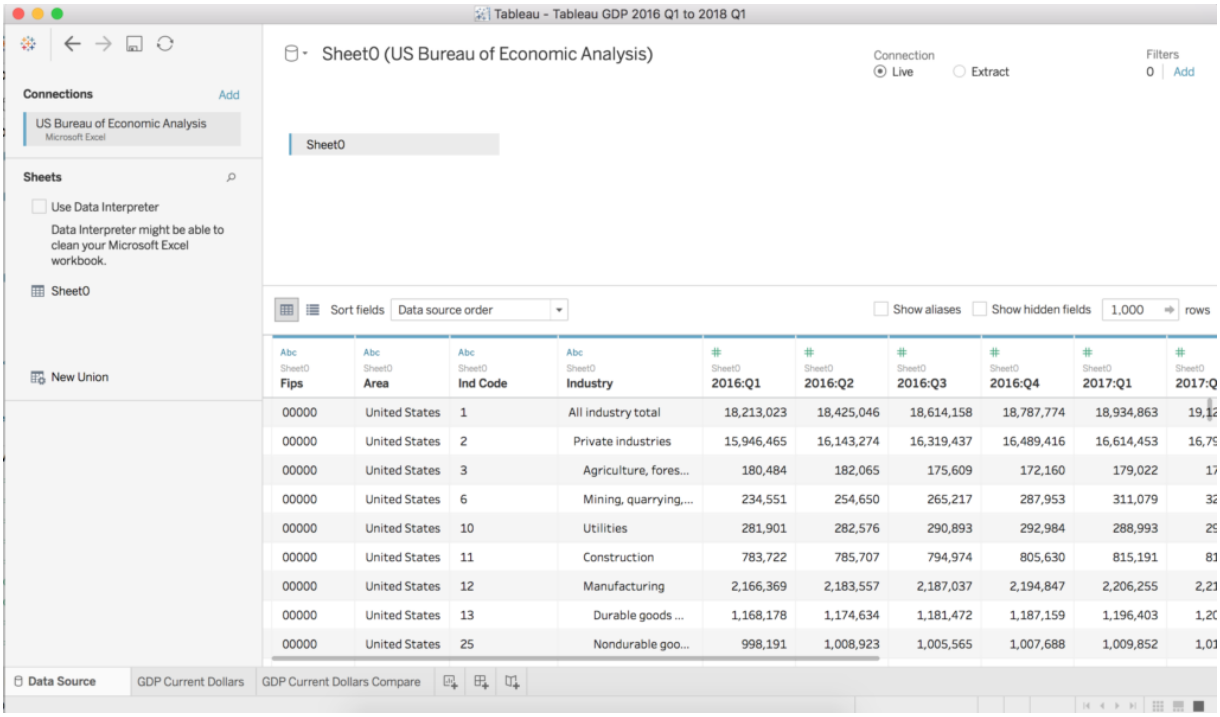
Sales and Marketing Analytics: This tool helps analyze sales performance, customer behavior, and marketing campaign effectiveness through visualizations.

Healthcare Analytics: Tableau analyzes patient data, tracks healthcare outcomes, and improves operational efficiency in healthcare institutions.

Financial Analytics: It visualizes complex financial data and supports financial analysis, risk management, and forecasting.

2.5.2 Conclusion

Tableau is a valuable tool in the data science toolkit, particularly for data visualization and business intelligence. Its ease of use, powerful visual analytics, and ability to integrate with various data sources and analytical tools make it an essential platform for transforming data into actionable insights.



A snapshot of Data dropped into Tableau

2.6 KNIME

KNIME (Konstanz Information Miner) is an open-source data analytics, reporting, and integration platform widely used in data science. It enables users to create data workflows, perform complex data manipulations, and execute advanced analytics through a visual programming interface.

2.6.1 Key Features and Uses in Data Science

Visual Workflow Interface:

Drag-and-Drop Functionality: KNIME provides an intuitive drag-and-drop interface for designing data workflows without requiring extensive coding knowledge.

Node-based Workflows: KNIME workflows are constructed using nodes, each representing a specific data operation, which can be connected to form a comprehensive data processing pipeline.

Data Integration:

Diverse Data Sources: KNIME can integrate data from various sources, including databases, spreadsheets, cloud services, and big data platforms.

ETL Capabilities: It excels in ETL (Extract, Transform, Load) processes, allowing for efficient data extraction, transformation, and loading into target systems.

Data Transformation and Preparation:

Comprehensive Data Manipulation: KNIME provides extensive tools for data cleaning, transformation, normalization, and aggregation, which is essential for preparing data for analysis.

Interactive Data Views: Users can explore and interact with data directly through various views and visualizations within the platform.

Advanced Analytics and Machine Learning:

Built-In Algorithms: KNIME includes numerous built-in machine learning and statistical algorithms for tasks such as classification, regression, clustering, and text mining.

Integration with R and Python: KNIME workflows can incorporate custom scripts written in R and Python, leveraging advanced analytics and machine learning libraries from these languages.

Big Data and Cloud Integration:

Big Data Connectors: KNIME supports big data technologies such as Hadoop, Spark, and Hive, enabling the processing of large datasets.
Cloud Integration: It can connect to cloud platforms like AWS, Azure, and Google Cloud, facilitating cloud-based data analytics.
Visualization and Reporting:
Interactive Dashboards: KNIME provides tools to create interactive dashboards and visual reports, helping to communicate insights effectively.
Data Reporting: Users can generate comprehensive reports in various formats, including PDF and HTML, directly from their workflows.
Extensibility and Community:
Node Extensions: The platform supports extensions and plugins, allowing users to add new functionalities and integrate with additional tools and services.
Active Community: KNIME has a robust community of users and developers who contribute to its development and provide support through forums and shared workflows.

2.6.2 Applications

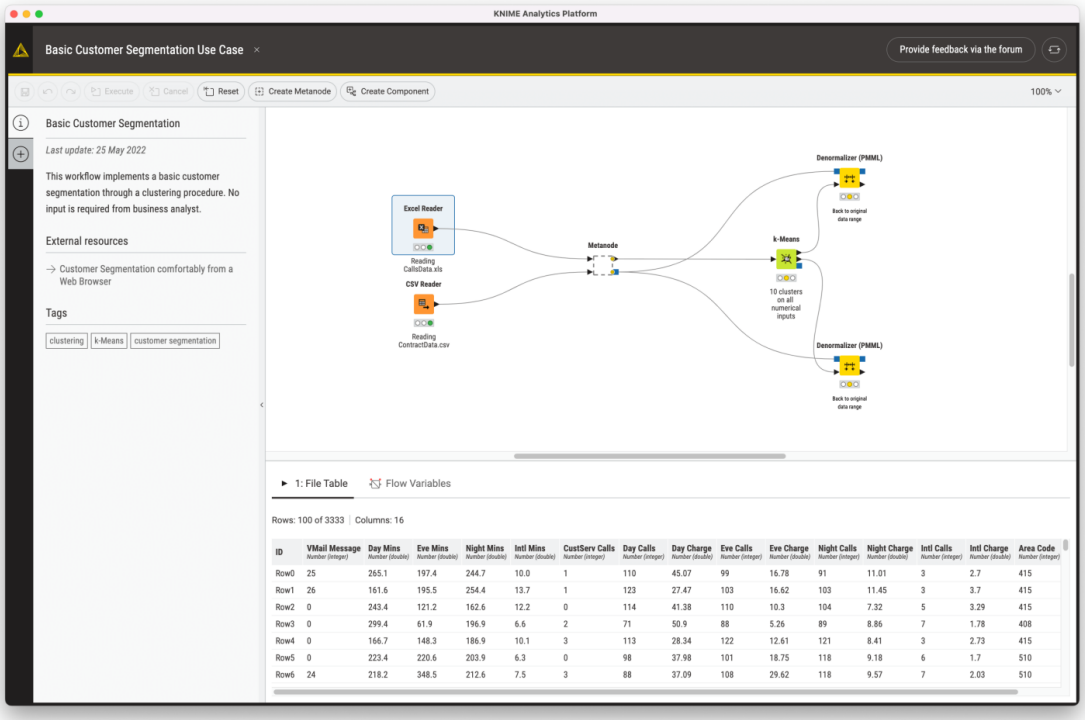
Predictive Analytics: KNIME is used to build predictive models that help businesses forecast trends and make data-driven decisions.
Customer Analytics: It helps analyze customer data to understand behaviour, preferences, and segmentation, aiding in targeted marketing strategies.
Pharmaceutical Research: KNIME is extensively used in bioinformatics and cheminformatics for drug discovery and research.

2.6.3 Limitations

While KNIME is highly versatile, users unfamiliar with its node-based interface may experience a learning curve. Additionally, complex customizations might require knowledge of scripting languages like Python or R.

2.6.4 Conclusion

KNIME is a comprehensive and flexible platform for data science. It offers powerful data integration, transformation, and analytics capabilities through an intuitive visual interface. Its ability to handle diverse data sources and advanced analytics makes it a valuable tool for data scientists in various industries.



Knime interface

2.7 STATISTICAL ANALYSIS SYSTEM(SAS)

SAS (Statistical Analysis System) is a comprehensive software suite developed by the SAS Institute for advanced analytics, multivariate analysis, business intelligence, data management, and predictive analytics. It is widely used in data science for its powerful statistical capabilities and robust data handling features.

2.7.1 Key Features and Uses in Data Science

Data Management:

Data Access: SAS can access data from various sources, including databases, spreadsheets, and flat files, ensuring seamless data integration.

Data Cleaning and Preparation: SAS provides extensive tools for data cleaning, transformation, and preparation, which are essential for accurate analysis.

Statistical Analysis:

Descriptive Statistics: SAS offers a wide range of procedures to summarize and describe data, such as mean, median, variance, and frequency distributions.

Inferential Statistics: It supports complex statistical analyses, including hypothesis testing, ANOVA, regression analysis, and survival analysis.

Advanced Analytics and Machine Learning:

Predictive Modeling: SAS develops predictive models using linear and logistic regression techniques, decision trees, and neural networks.

Machine Learning: SAS provides tools for machine learning algorithms, including clustering, classification, and association rule mining.

Data Visualization:

Graphical Representation: SAS offers extensive data visualization capabilities, including the ability to create charts, graphs, and plots, which help in exploring and presenting data insights effectively.

Interactive Dashboards: SAS Visual Analytics allows users to create interactive and shareable dashboards that facilitate data exploration and reporting.

Reporting and B.I.:

Automated Reporting: SAS enables the creation of automated reports that can be scheduled and distributed to stakeholders.

Business Intelligence: SAS integrates with B.I. tools to provide comprehensive business intelligence solutions, supporting decision-making processes.

Integration and Extensibility:

Programming Interfaces: SAS supports integration with other programming languages, such as Python, R, and SQL, enhancing its analytical capabilities.

APIs: It offers APIs to integrate with various applications and platforms, facilitating seamless workflow integration.

Text Analytics and Natural Language Processing:

Text Mining: SAS provides text mining and natural language processing tools, enabling the analysis of unstructured data such as social media content and customer reviews.

2.7.2 Applications

Healthcare Analytics: Used for clinical trial analysis, patient data management, and predictive modelling for disease outbreaks.

Financial Services: Applied in risk management, fraud detection, and customer segmentation.

Marketing Analytics: Helps in customer behavior analysis, campaign effectiveness measurement, and targeted marketing.

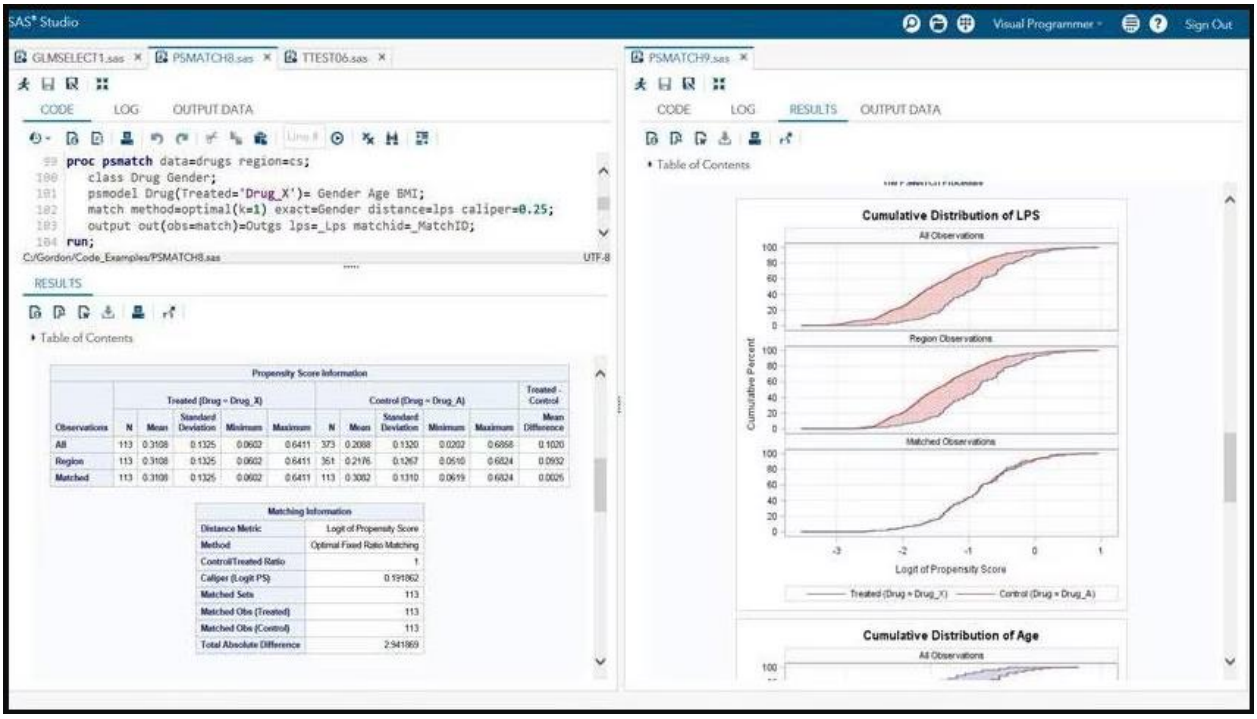
Government and Public Sector: Used for policy analysis, public health monitoring, and resource allocation.

2.7.3 Limitations

While SAS is robust and comprehensive, it can be costly, which may be a barrier for small organizations or individual users.

2.7.4 Conclusion

SAS is a robust and versatile tool in the data science toolkit. It offers extensive capabilities for statistical analysis, advanced analytics, data management, and visualization. Its wide range of features and integration options make it a valuable platform for data-driven decision-making in various industries.



SAS interface

2.8 PYTHON

Python is a dominant programming language in data science due to its simplicity, extensive libraries, and strong community support. Its versatility allows data scientists to perform a wide range of tasks, from data manipulation and analysis to machine learning and visualization.

2.8.1 Key Features and Uses in Data Science

Ease of Use and Readability:

Simple Syntax: Python's syntax is straightforward, making it easy to learn and use, which is crucial for data scientists who need to focus on solving complex problems rather than dealing with language intricacies.

Rapid Development: Python allows for quick prototyping and development, enabling data scientists to experiment with different approaches and iterate rapidly.

Extensive Libraries:

NumPy provides support for large, multidimensional arrays and matrices and a collection of mathematical functions to operate on them.

Pandas: Essential for data manipulation and analysis, pandas provide data structures like DataFrames, simplifying working with structured data.

SciPy: Builds on NumPy and provides additional tools for scientific and technical computing, including modules for optimization, Integration, and statistics.

Matplotlib and Seaborn: These libraries are used for data visualization, and they allow the creation of a wide variety of static, animated, and interactive plots.

Scikit-learn: A key library for machine learning in Python, providing simple and efficient data mining and analysis tools.

TensorFlow and PyTorch: Widely used for building and deploying machine learning and deep learning models, supporting extensive neural network operations.

Data Manipulation and Analysis:

Data Cleaning: Python's libraries, particularly pandas, offer functions for handling missing data, filtering, and transforming datasets.
Statistical Analysis: Libraries like SciPy and stats models provide comprehensive tools for performing statistical tests and building statistical models.

Machine Learning and AI:

Model Building: Scikit-learn offers a range of supervised and unsupervised learning algorithms, making it easy to implement machine learning models.
Deep Learning: TensorFlow and PyTorch enable the construction of deep learning models, supporting tasks like image recognition, natural language processing, and more.
Automation: Python's flexibility allows for automating repetitive tasks, such as data collection, preprocessing, and model evaluation.

Data Visualization:
Customizable Plots: Matplotlib and Seaborn enable the creation of detailed and customizable visualizations, helping in the exploratory data analysis (EDA) and presentation of results.
Interactive Dashboards: Libraries like Plotly and Bokeh allow the creation of interactive dashboards that can be shared with stakeholders.
Integration and Extensibility:

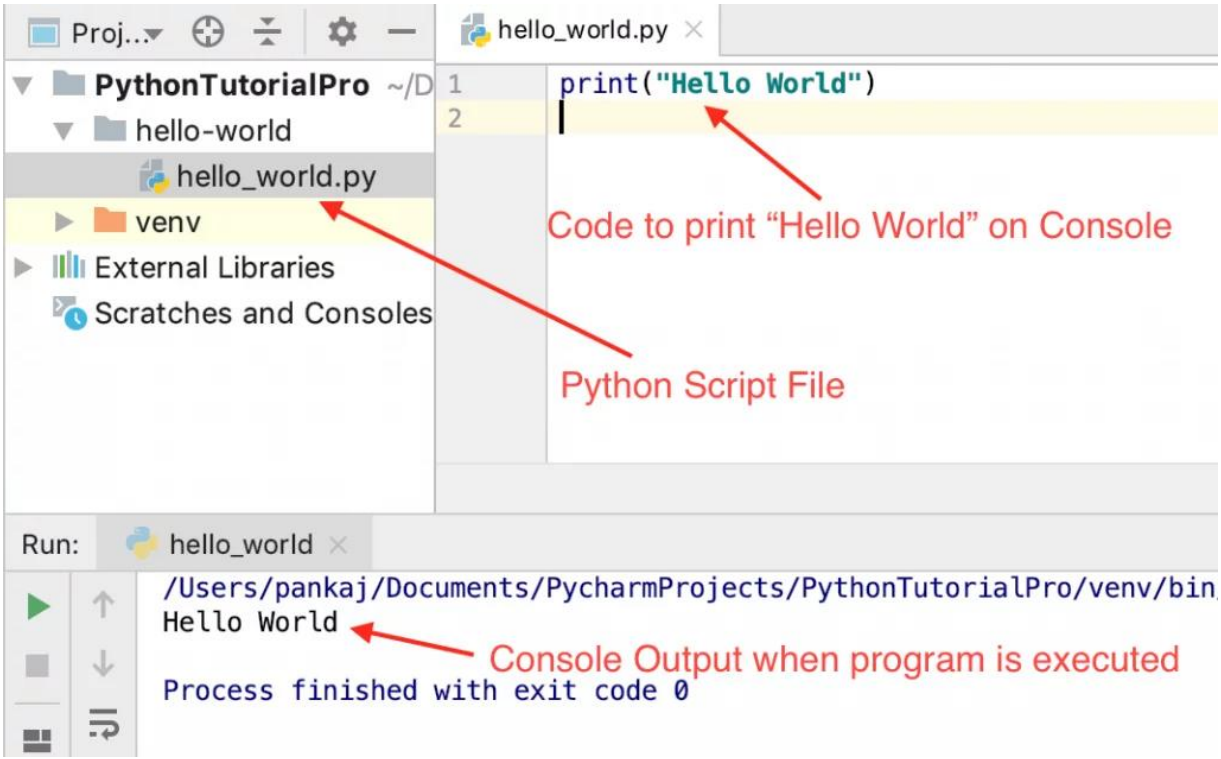
APIs and Data Sources: Python can easily connect to various APIs and databases, facilitating seamless data integration from multiple sources.
Jupyter Notebooks: Widely used for data exploration and sharing, Jupyter Notebooks provide an interactive environment for writing and executing Python code, visualizing data, and documenting analysis.

2.8.2 Applications

Predictive Analytics: Python is used to develop models that predict future trends based on historical data, helping businesses make informed decisions.
Natural Language Processing: Libraries like NLTK and spaCy enable text analysis, sentiment analysis, and other NLP tasks.
Image and Video Processing: OpenCV and deep learning frameworks support tasks like image classification, object detection, and video analysis.
Financial Modeling: Python's statistical and analytical capabilities are leveraged for risk assessment, portfolio management, and financial forecasting.

2.8.3 Conclusion

Python is a fundamental tool in data science. It provides a rich ecosystem of libraries and tools that streamline data manipulation, analysis, machine learning, and visualization. Its ease of use and flexibility make it an indispensable language for data scientists across various domains.



Python interface showing a simple Basic Code

2.9 PYTHON LIBRARIES

2.9.1 Numpy

NumPy (Numerical Python) is a foundational library for numerical computing in Python. It provides support for arrays, matrices, and many mathematical functions to operate on these data structures.

Key Features

1. **N-Dimensional Array Object**: The primary feature of NumPy is its powerful N-dimensional array object, `ndarray`, which allows for efficient storage and manipulation of large datasets.
2. **Mathematical Functions**: NumPy offers a wide array of mathematical functions, including linear algebra, random number generation, and Fourier transforms.
3. **Broadcasting**: This feature allows operations to be performed on arrays of different shapes, making it easier to perform element-wise operations without explicit looping.
4. **Integration**: NumPy arrays are used as the primary data structure in other data science libraries, such as pandas, SciPy, and sci-kit-learn.

Applications

- Data preprocessing and cleaning
- Performing mathematical and statistical operations
- Handling large datasets efficiently
- Basis for more complex data science and machine learning libraries

2.9.2 Pandas

Pandas

Pandas is a powerful data manipulation and analysis library built on top of NumPy. It provides two primary data structures: Series (1-dimensional) and DataFrame (2-dimensional).

Key Features

1. **DataFrames** are two-dimensional, size-mutable, and potentially heterogeneous tabular data structures with labelled axes.
2. **Data Alignment**: Pandas automatically aligns data by labels, simplifying joining, merging, and concatenating datasets.
3. **Data Cleaning**: Functions for handling missing data, filtering, transforming, and aggregating data.
4. **Time Series Analysis**: Robust tools for working with time series data, including date range generation, frequency conversion, moving window statistics, and more.

Applications

- Data wrangling and cleaning
- Exploratory data analysis (EDA)
- Time series analysis
- Preparing data for machine learning models

2.9.3 Seaborn

Seaborn is a data visualization library based on Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

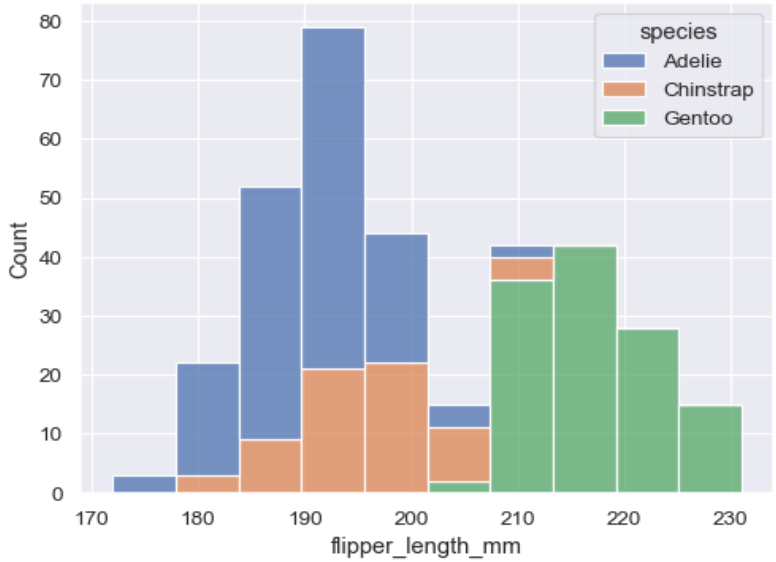
Key Features

1. **Statistical Plotting**: Seaborn simplifies creating complex visualizations such as violin plots, box plots, and heat maps.
2. **Themes and Color Palettes**: Provides default themes and colour palettes to make plots more visually appealing.
3. **Integration with Pandas**: Easily handles DataFrame objects and integrates well with other data science libraries.
4. **Facet Grids**: Allows for the creation of multi-plot grids to visualize relationships across multiple variables.

Applications

- Visualizing distributions of data
- Exploring relationships between variables
- Creating informative and attractive statistical plots
- Enhancing the interpretability of complex data analysis

```
penguins = sns.load_dataset("penguins")
sns.histplot(data=penguins, x="flipper_length_mm", hue="species", multiple="stack")
```



A chart showing how seaborn is used to display data

2.9.4 Plotly

Plotly is a graphing library that makes interactive, publication-quality graphs online. It is particularly known for its ability to create interactive visualizations.

Key Features

1. **Interactive Plots**: Unlike static plots from libraries like Matplotlib, Plotly visualizations are interactive and can be embedded in web applications.
2. **Wide Range of Plots**: It supports a wide variety of plot types, including scatter plots, bar charts, line charts, pie charts, bubble charts, box plots, and many more.
3. **Dash**: Plotly's Dash framework builds analytical web applications without requiring JavaScript.
4. **Customization**: Highly customizable visualizations allow for detailed control over the aesthetics and behaviour of the plots.

Applications

- Creating interactive dashboards and visual analytics
- Presenting data insights in an engaging manner
- Building web-based data visualization applications
- Real-time data visualization in web apps

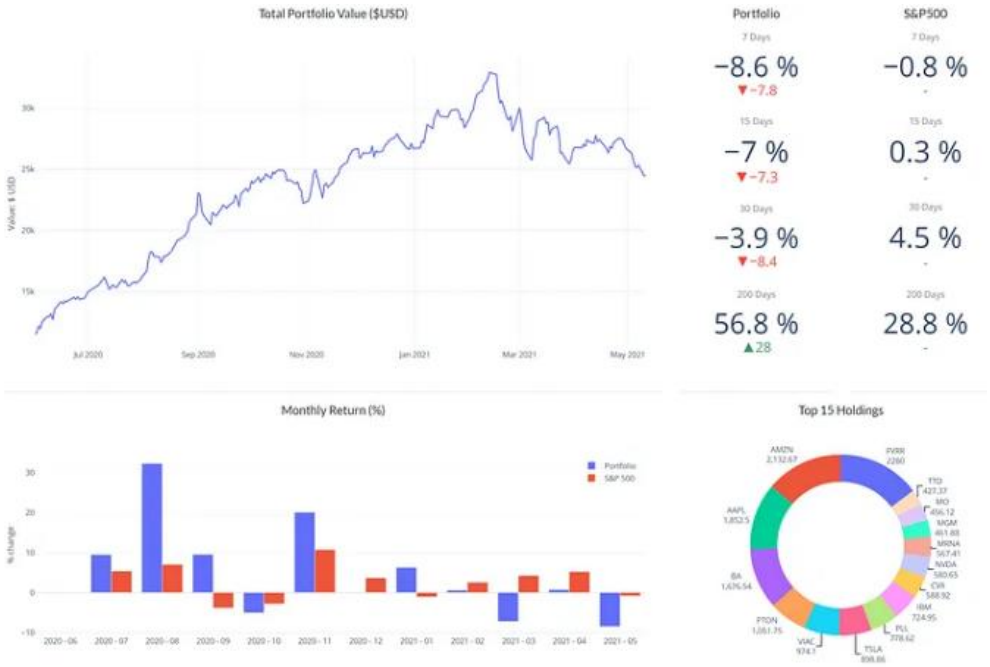


Image showing the help of Plotly library in Python to track different investment portfolio

2.9.4 Scipy

SciPy (Scientific Python) is an open-source library that builds on NumPy and provides a collection of algorithms and high-level commands for scientific and technical computing. It is a fundamental library in the Python data science ecosystem, offering a broad range of mathematics, science, and engineering tools.

Key Features

Scientific and Technical Computing:

Optimization: Functions for finding the minimum or maximum of a function, curve fitting, and root finding.

Integration: Numerical integration routines, including single, double, and multiple integrals.

Linear Algebra: Solvers for linear systems, eigenvalue problems, matrix decompositions, and more.

Statistics: Statistical functions for probability distributions, descriptive statistics, and hypothesis testing.

Signal Processing: Tools for filtering, spectral analysis, and signal transformations.

Modules and Functions:

scipy.optimize Optimization algorithms, including minimization and root-finding routines.

scipy.integrate Integration and ordinary differential equation solvers.

scipy.linalg: Linear algebra routines and matrix decompositions.

scipy.stats: Statistical distributions and functions for descriptive and inferential statistics.

scipy.signal: Signal processing tools, including filtering, spectral analysis, and wavelet transforms.

scipy.spatial: Spatial data structures and algorithms, including nearest neighbour searches and distance computations.

scipy.ndimage: Multidimensional image processing functions.

Interoperability with NumPy:

Seamless Integration: SciPy is built on NumPy, ensuring seamless Integration and efficient handling of NumPy arrays.

Enhanced Functionality: SciPy extends NumPy's capabilities by adding more sophisticated algorithms and functions for scientific computing.

Applications

Optimization:

Curve Fitting: Use scipy.optimize.curve_fit to fit data to a model function.

Root Finding: Solving equations using functions like scipy.optimize.root.

Integration:

Numerical Integration: Functions like scipy.integrate.quad for single integrals and scipy.integrate.dblquad for double integrals.

ODE Solvers: Solving ordinary differential equations using scipy.integrate.solve_ivp.

Linear Algebra:

Matrix Decompositions: Performing LU, QR, and singular value decompositions.

Linear Systems: Solving linear systems using scipy.linalg.solve.

Statistics:

Probability Distributions: Working with continuous and discrete distributions.

Hypothesis Testing: Functions for t-tests, chi-square tests, and other statistical tests.

Signal Processing:

Filtering: Designing and applying filters to signals.

Spectral Analysis: Computing Fourier transforms and power spectra.

Image Processing:

Multidimensional Image Processing: Functions for image filtering, morphology, and transformations.

SciPy is an essential library for data science, providing powerful tools for scientific and technical computing. Its extensive modules and functions allow data scientists to perform complex mathematical operations, statistical analysis, signal processing, and more, all while leveraging the efficiency and interoperability of NumPy arrays. By extending NumPy's capabilities, SciPy enables more advanced and sophisticated data analysis and modelling, making it a crucial Python data science ecosystem component.

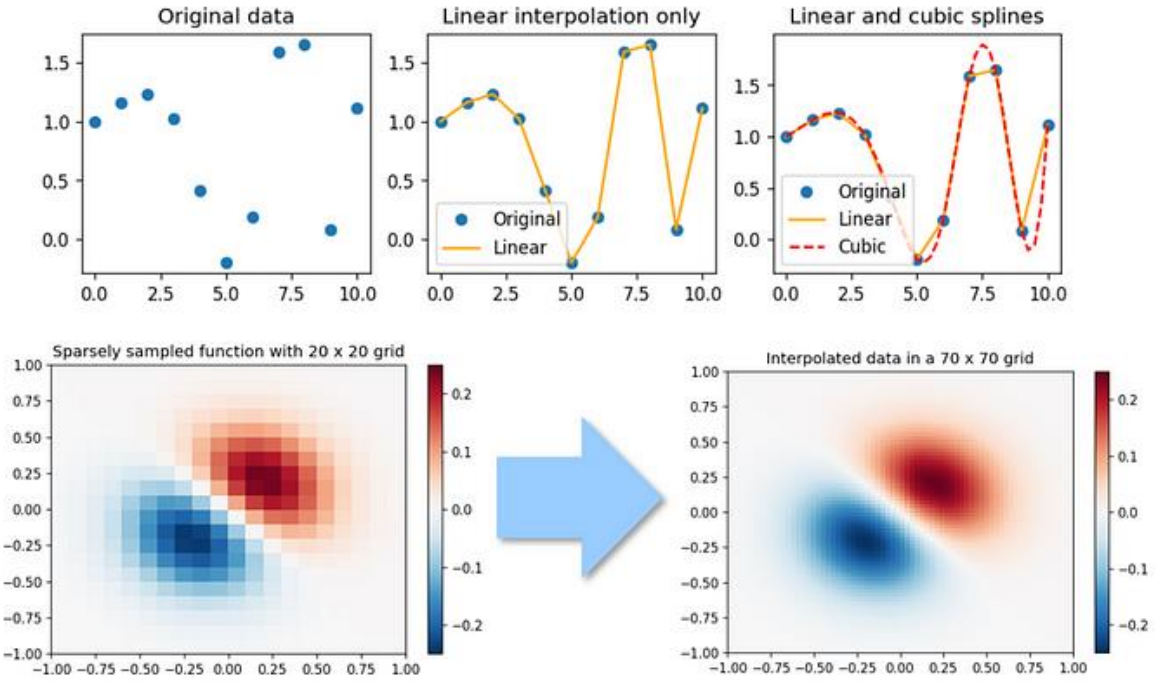


Image showing Interpolation Done with Scipy

Conclusion

These libraries—NumPy, pandas, Seaborn, Plotly, and Scipy—form a robust ecosystem for data science in Python. They provide the necessary tools for data manipulation, analysis, and visualization, each with its unique strengths:

- **NumPy**: Foundation for numerical computing
- **Pandas**: Data manipulation and analysis
- **Seaborn**: Statistical data visualization
- **Plotly**: Interactive and web-based visualizations
- **Scipy:** powerful tools for scientific and technical computing

Together, they enable data scientists to efficiently process, analyze, and visualize data, facilitating deeper insights and more effective results communication.

2.10 R PROGRAMMING LANGUAGE

R is a comprehensive programming language and software environment primarily used for statistical computing, data analysis, and graphical representation. It has gained immense popularity in the data science community due to its rich ecosystem of packages, robust statistical capabilities, and powerful data visualization tools.

Key Features and Benefits

Statistical Analysis:

R is designed specifically for statistical analysis, making it highly suitable for data scientists who need to perform complex statistical tests, build models, and analyze data distributions.

Data Visualization:

R offers advanced data visualization capabilities through packages like ggplot2, lattice, and others, enabling the creation of detailed and aesthetically pleasing graphs and plots.

Data Manipulation:

Packages like dplyr and tidyr provide powerful data manipulation tools, allowing users to clean, transform, and prepare data efficiently.

Integration with Other Tools:

R can be integrated with databases, web applications, and other programming languages (e.g., Python, SQL), enhancing its versatility in data science projects.

Applications in Data Science

Data Wrangling and Cleaning:

R's robust data manipulation packages include dplyr, tidy, and data. Tables are widely used to clean and transform raw data into an analysis-ready format.

Exploratory Data Analysis (EDA):

R excels in EDA, offering numerous functions and packages to summarize, visualize, and understand data distributions and relationships.

Statistical Modeling:

R provides a comprehensive suite of tools for building and evaluating statistical models, including linear regression, logistic regression, and various machine learning algorithms.

Data Visualization:

The ggplot2 package, based on the grammar of graphics, is a powerful tool for creating complex and customizable visualizations. It helps data scientists effectively communicate their findings.

Machine Learning:

R supports machine learning through packages like caret, randomForest, and xgboost, facilitating predictive model development, training, and evaluation.

Time Series Analysis:

Specialized packages like Forecast and Zoo provide tools for analyzing and forecasting time series data, widely used in financial modelling, economics, and environmental studies.

Example Usage

Here is an example of how R can be used for data manipulation and visualization:

```
# Load necessary libraries
library(dplyr)
library(ggplot2)

# Sample data frame
data <- data.frame(
  id = 1:10,
  score = c(8, 7, 6, 9, 10, 5, 7, 6, 9, 8),
  group = rep(c("A", "B"), each = 5)
)

# Data manipulation with dplyr
data_summary <- data %>%
  group_by(group) %>%
  summarise(mean_score = mean(score), sd_score = sd(score))
```

```
print(data_summary)

# Data visualization with ggplot2
ggplot(data, aes(x = group, y = score)) +
  geom_boxplot() +
  labs(title = "Score Distribution by Group", x = "Group", y = "Score")
```

This code snippet demonstrates how to use dplyr to summarize data and ggplot2 to create a boxplot to visualize the distribution of scores by group.

2.11 SEQUENCE QUERY LANGUAGE

SQL is a crucial tool in data science for managing and manipulating relational databases. It allows data scientists to efficiently query, update, and manage large datasets stored in databases. SQL is used for data extraction, which is the first step in the data analysis process, making it possible to retrieve the exact data needed for analysis. It enables complex operations such as joins, aggregations, and subqueries, providing a powerful way to organize and preprocess data before performing more advanced analytics.

2.12 JUPYTER NOTEBOOK

Jupyter Notebook is an open-source web application extensively used in data science to create and share documents containing live code, equations, visualizations, and narrative text. It supports interactive data exploration and analysis, which is essential for data scientists to test hypotheses, visualize data, and document the analysis process. Jupyter Notebooks facilitate reproducible research by allowing data scientists to share their entire workflow, including code and results, in a single, accessible document.

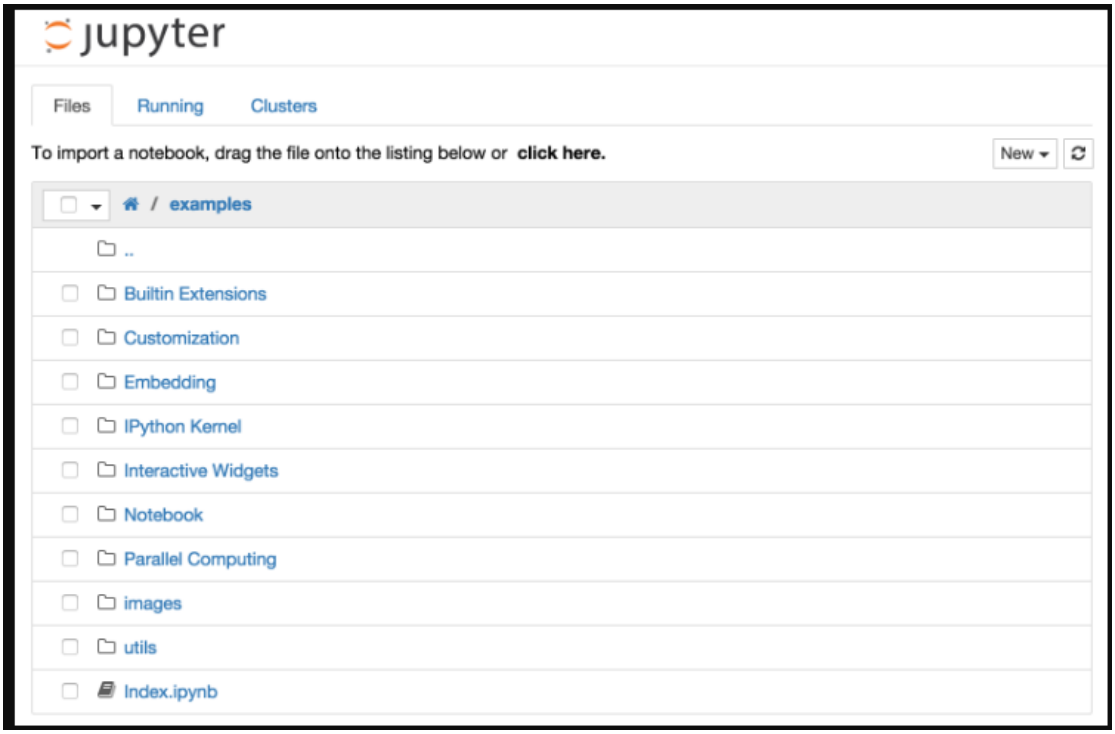


Image of jupyter notebook

2.12 GOOGLE COLAB

Google Colab (Collaboratory) is a free cloud-based environment that provides Jupyter notebooks hosted by Google. It is widely used in data science for its accessibility and computational power, allowing data scientists to leverage Google's infrastructure, including GPUs and TPUs, for faster and more efficient computation. Google Colab is particularly useful for collaborative projects and notebook sharing, as it integrates seamlessly with Google Drive and supports real-time collaboration.

2.13 GITHUB

GitHub is an essential platform for version control and collaboration in data science. It allows data scientists to store, manage, and track changes to their code repositories, making it easier to collaborate on projects and maintain a history of code versions. GitHub supports open-source contributions, enabling data scientists to share their code, datasets, and tools with the community. It also serves as a repository for numerous libraries and resources that can accelerate the development and deployment of data science projects.

CHAPTER 3

3.0 PROJECTS UNDERTAKEN

The Student Industrial Work Experience Scheme (SIWES) aims to allow students to acquire new skills and put them to good use. In this chapter, I will outline the various data science projects I was involved in during my time at Peak Info Tech Systems. I will focus on the key initiatives I contributed to during my tenure and highlight some personal projects that I undertook to enhance my understanding of data analysis and machine learning.

Throughout my internship, I had the chance to delve into a wide range of topics that piqued my interest, primarily centred around data analysis, data visualization, and introductory machine learning. I will provide specific details on the different aspects of these tasks, which greatly facilitated my learning and evaluation process.

In addition to my collaborative efforts, I embarked on a few personal projects aimed at deepening my knowledge of data science principles and techniques. As I neared the end of my industrial training, I was assigned to a team where I assumed responsibility for analyzing large datasets and developing predictive models, combining my growing data analysis skills with machine learning expertise for a holistic learning experience.

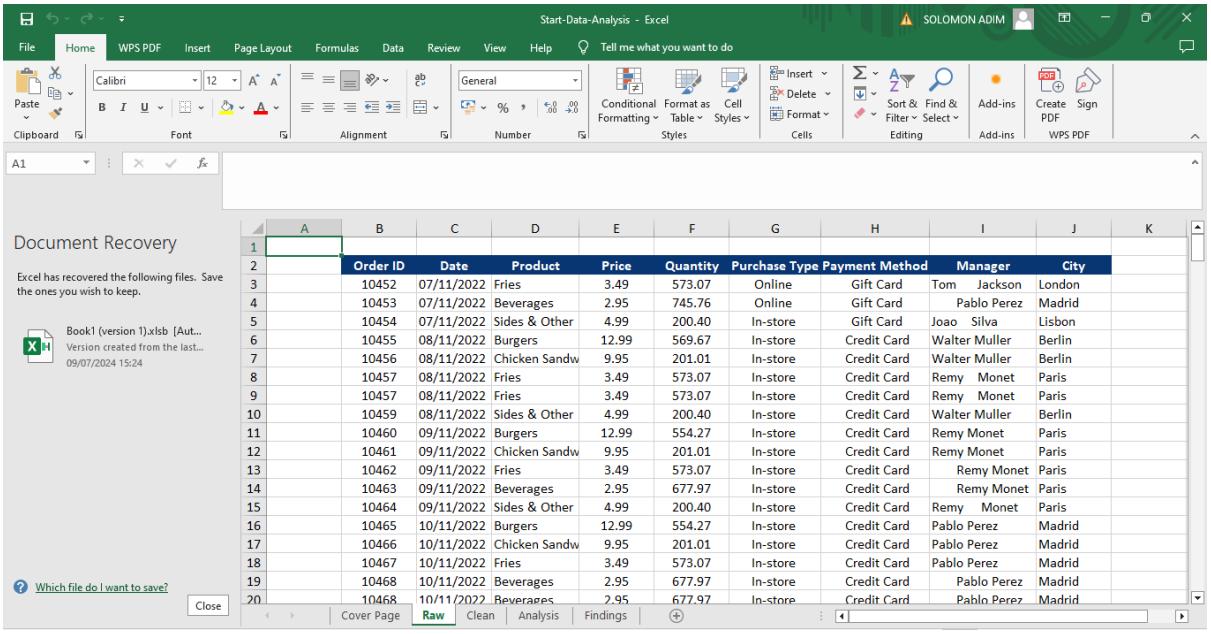
DATA CLEANING

Data cleaning is a crucial process in data analysis that involves identifying and correcting errors, inconsistencies, and inaccuracies in a dataset to ensure its quality and reliability. This process is essential for obtaining accurate and meaningful insights from data.

Key Steps in Data Cleaning:

1. **Removing Duplicates:** Identifying and eliminating duplicate entries to avoid skewed results.
2. **Handling Missing Values:** Addressing missing data by either filling in gaps using appropriate methods (e.g., mean, median, mode) or removing incomplete records if necessary.
3. **Correcting Errors:** Fixing typographical errors, incorrect data entries, and formatting issues to maintain consistency.
4. **Standardizing Data:** Ensuring data is in a consistent format, such as standardizing date formats, units of measurement, and categorical labels.
5. **Outlier Detection:** Identifying and managing outliers that may distort analysis, either by investigating and correcting them or removing them if they are erroneous.
6. **Validation:** Checking data against known validation rules or reference data to ensure accuracy and consistency.

Data cleaning is a fundamental step in the data analysis workflow, essential for achieving meaningful and accurate insights from any dataset.



Order ID	Date	Product	Price	Quantity	Purchase Type	Payment Method	Manager	City
10452	07/11/2022	Fries	3.49	573.07	Online	Gift Card	Tom Jackson	London
10453	07/11/2022	Beverages	2.95	745.76	Online	Gift Card	Pablo Perez	Madrid
10454	07/11/2022	Sides & Other	4.99	200.40	In-store	Gift Card	Joao Silva	Lisbon
10455	08/11/2022	Burgers	12.99	569.67	In-store	Credit Card	Walter Muller	Berlin
10456	08/11/2022	Chicken Sandw	9.95	201.01	In-store	Credit Card	Walter Muller	Berlin
10457	08/11/2022	Fries	3.49	573.07	In-store	Credit Card	Remy Monet	Paris
10457	08/11/2022	Fries	3.49	573.07	In-store	Credit Card	Remy Monet	Paris
10459	08/11/2022	Sides & Other	4.99	200.40	In-store	Credit Card	Walter Muller	Berlin
10460	09/11/2022	Burgers	12.99	554.27	In-store	Credit Card	Remy Monet	Paris
10461	09/11/2022	Chicken Sandw	9.95	201.01	In-store	Credit Card	Remy Monet	Paris
10462	09/11/2022	Fries	3.49	573.07	In-store	Credit Card	Remy Monet	Paris
10463	09/11/2022	Beverages	2.95	677.97	In-store	Credit Card	Remy Monet	Paris
10464	09/11/2022	Sides & Other	4.99	200.40	In-store	Credit Card	Remy Monet	Paris
10465	10/11/2022	Burgers	12.99	554.27	In-store	Credit Card	Pablo Perez	Madrid
10466	10/11/2022	Chicken Sandw	9.95	201.01	In-store	Credit Card	Pablo Perez	Madrid
10467	10/11/2022	Fries	3.49	573.07	In-store	Credit Card	Pablo Perez	Madrid
10468	10/11/2022	Beverages	2.95	677.97	In-store	Credit Card	Pablo Perez	Madrid
10468	10/11/2022	Beverages	2.95	677.97	In-store	Credit Card	Pablo Perez	Madrid

An image of an Excel worksheet of me trying to clean the data in each column.

3.1 TOOLS USED IN DATA CLEANING

- Microsoft Excel

DATA VISUALIZATION

OVERVIEW

Data visualization is the graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.

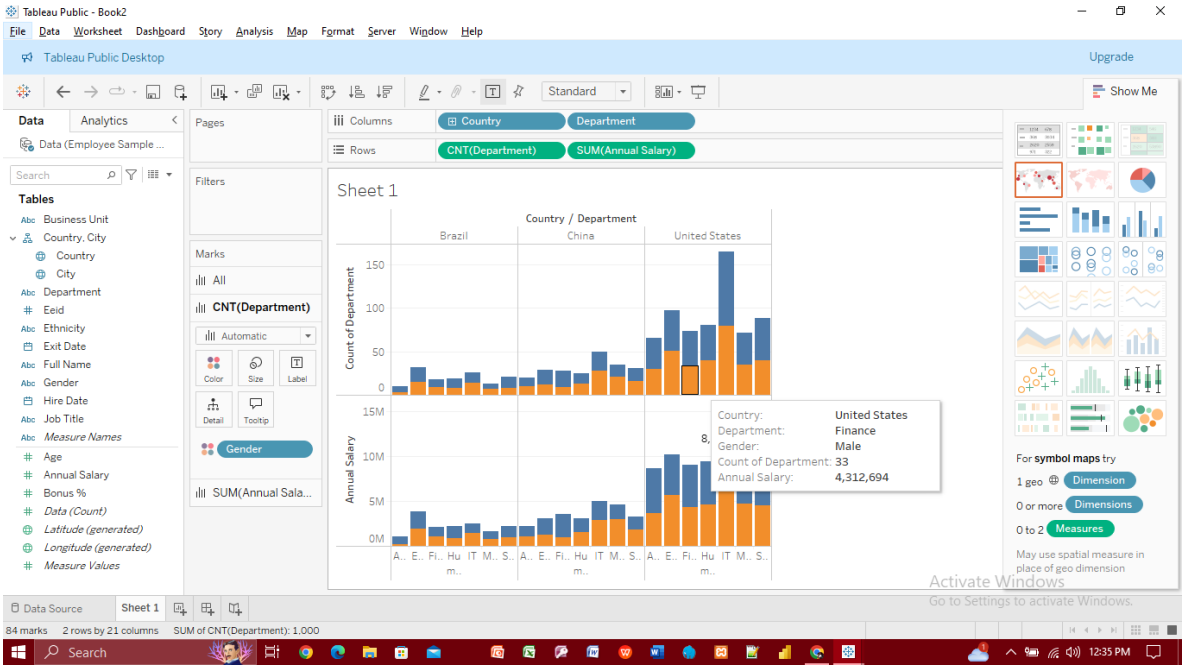


Image showing how data can be visualized

In one of my key projects, I used Tableau to visualize data on employees, focusing on the following aspects:

- **Country:** The geographical distribution of employees.
- **Department:** The various departments where employees work.
- **Gender:** The gender breakdown within the organization.
- **Annual Salary:** The salary distribution across different categories.

I employed stacked bar charts to present this information. Stacked bar charts are effective for showing the composition of different groups within a total, making it easy to compare sub-categories.

This project involved:

- **Data Cleaning and Preparation:** Ensuring the data was accurate and formatted correctly for analysis.
- **Chart Creation:** Tableau's drag-and-drop interface creates stacked bar charts that display the relationships between country, department, gender, and annual salary.
- **Insight Generation:** Analyzing the visualizations to uncover trends and insights, such as salary disparities across different departments and genders, or the concentration of employees in specific countries.

This experience not only honed my skills in data visualization but also provided valuable insights into the power of visual data analysis for effective decision-making and communication.

3.2 TOOLS USED DATA VISUALIZATION

- An Excel data set
- Tableau

Other Data visualization images I worked on

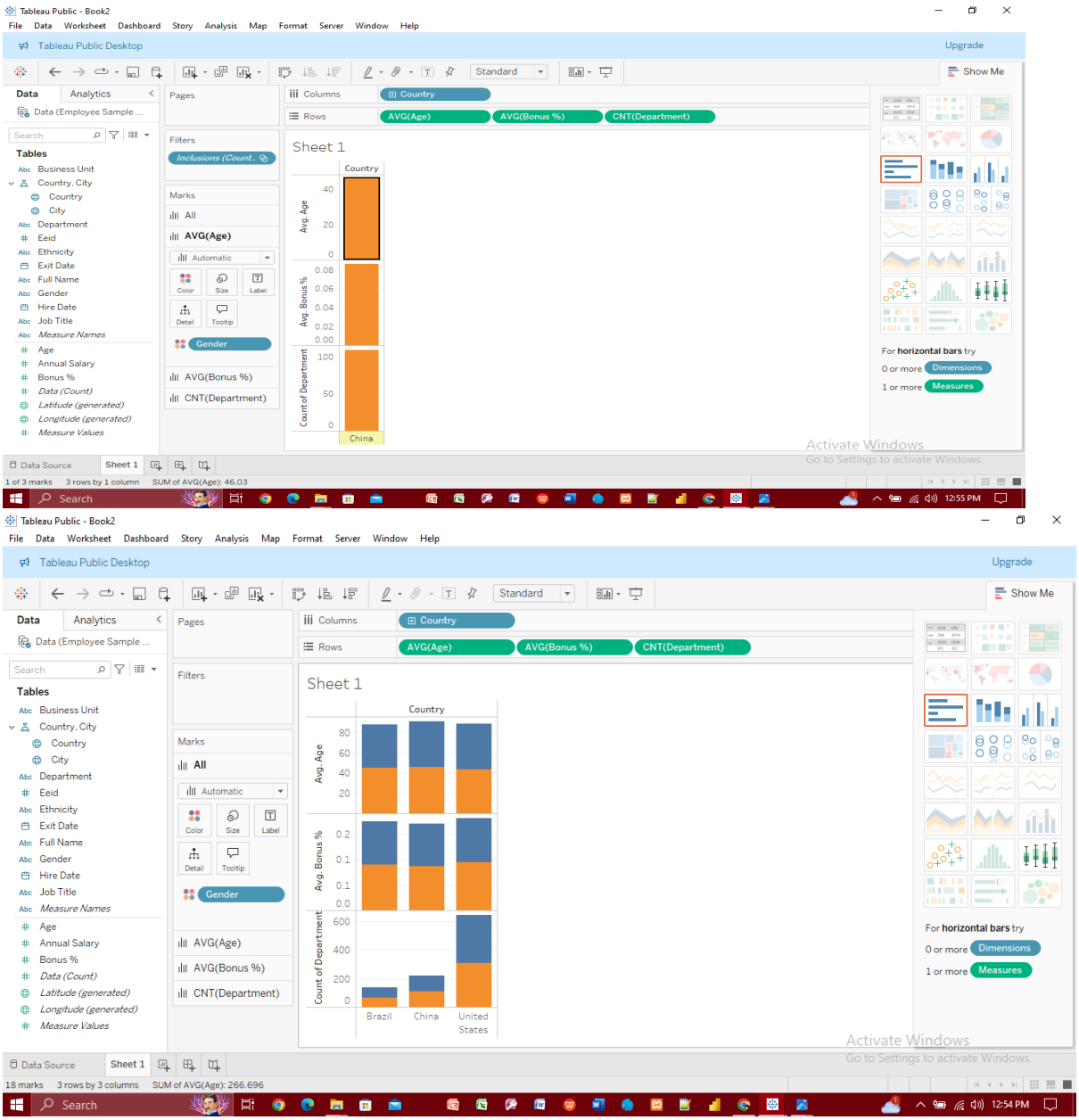
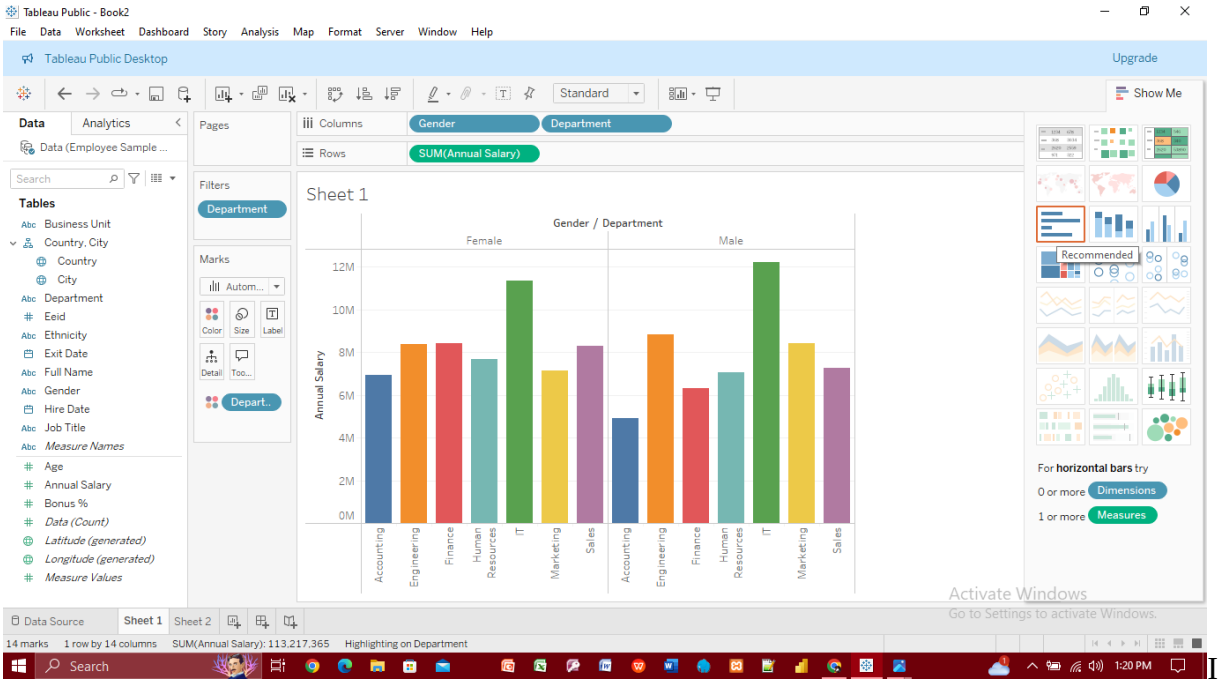


Image showing the average age, average bonus, and count of department computation

I used Tableau to visualize data on employees, focusing on the following aspects:

- **Average Age:** The average age of employees across different categories.
- **Average Bonus:** The average bonus received by employees.
- **Department Count:** The number of employees in each department.
- **Country:** The geographical distribution of employees.

I presented this information using horizontal bar charts. These charts are effective for comparing different categories side-by-side, making it easy to visualize the distribution of various metrics.



I used Tableau to visualize data on employees, focusing on the following aspects:

- **Gender:** The gender distribution of employees.
- **Department:** The various departments where employees work.
- **Salary:** The salary distribution across different departments and genders.

I employed horizontal bar charts to present this information. Horizontal bar charts are effective for comparing different categories side-by-side, making it easy to visualize the distribution and disparities within various metrics.

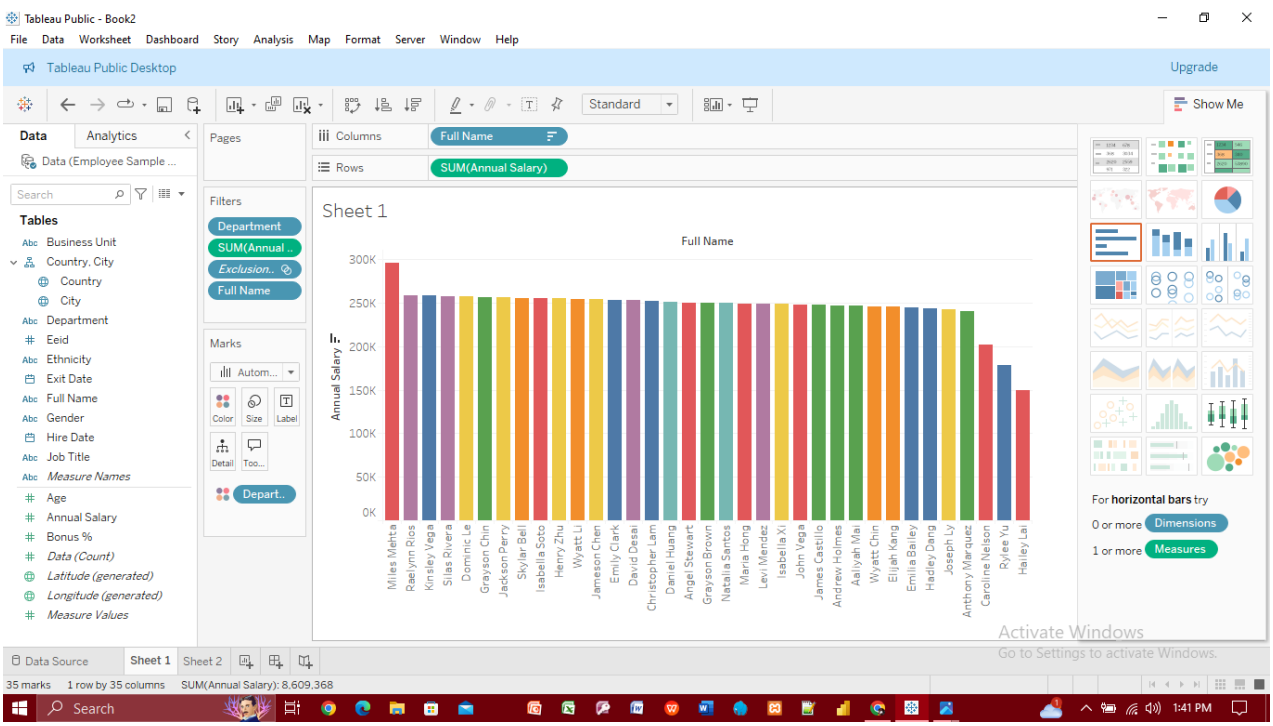
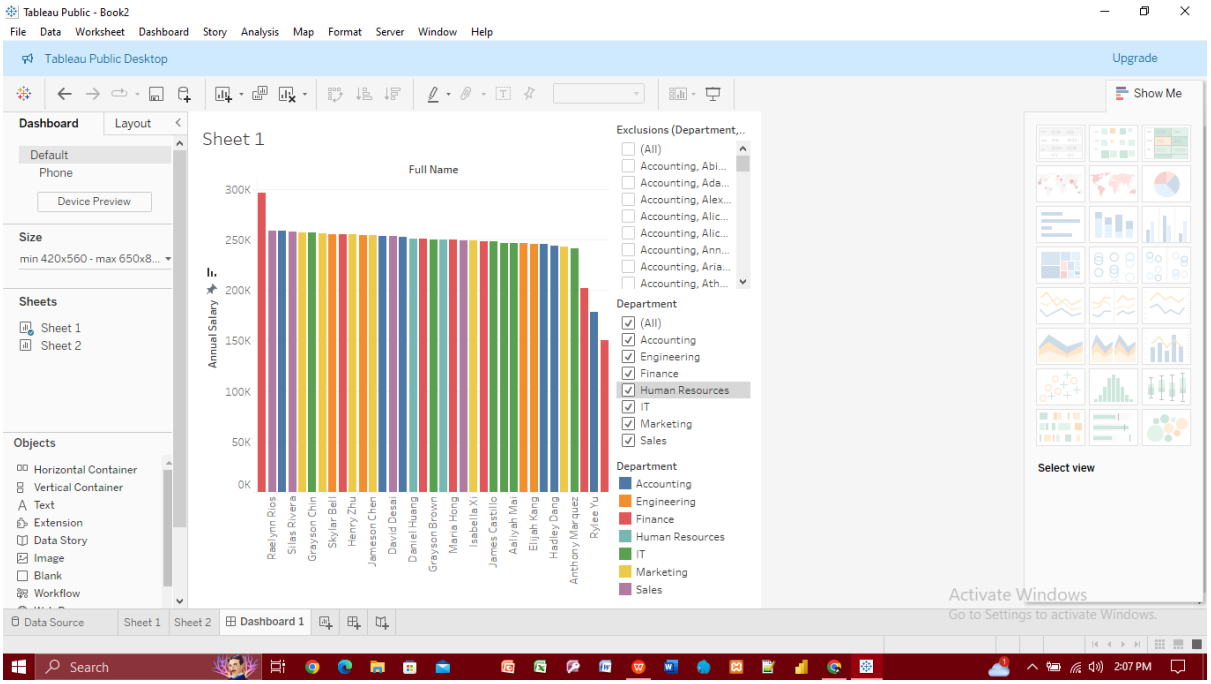


Image showing the top 35 paid people in the company

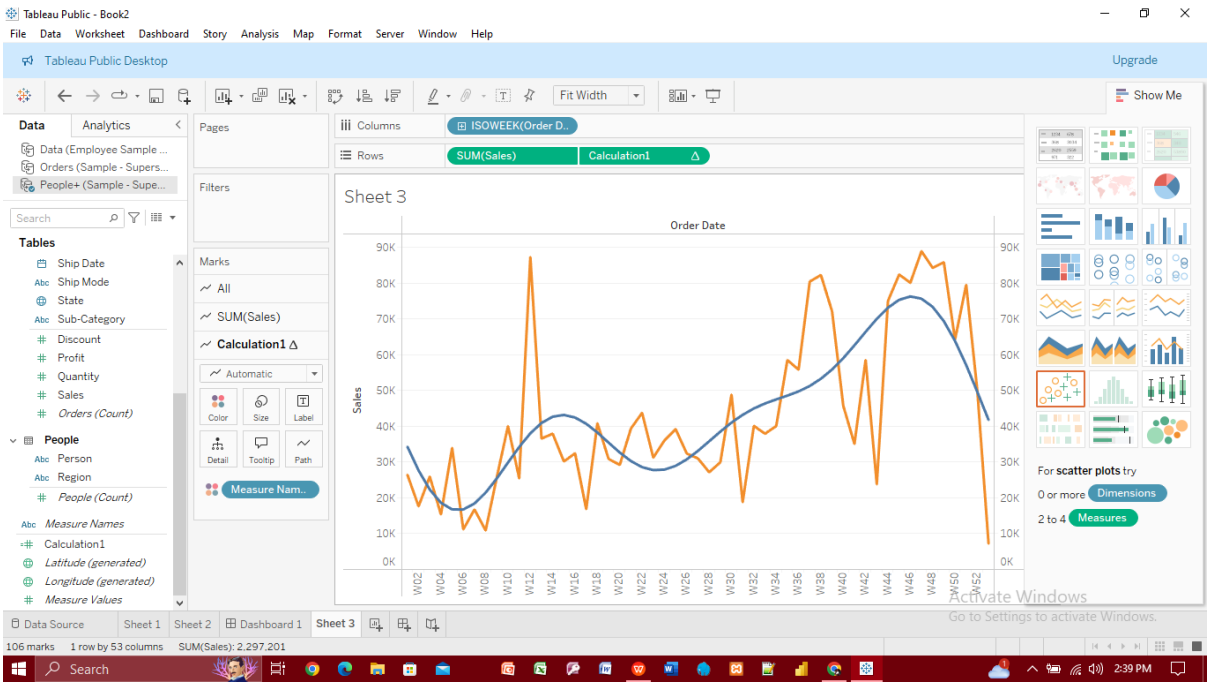
I used Tableau to visualize data on the top 35 highest-paid employees in the company, focusing on the following aspects:

- **Names of Employees:** Displaying the names of the top 35 highest-paid employees.
- **Salaries:** Showing the salary of each employee.

I presented this information using horizontal bar charts. These charts effectively compare different categories side-by-side, making it easy to visualize the salary distribution among the top earners.



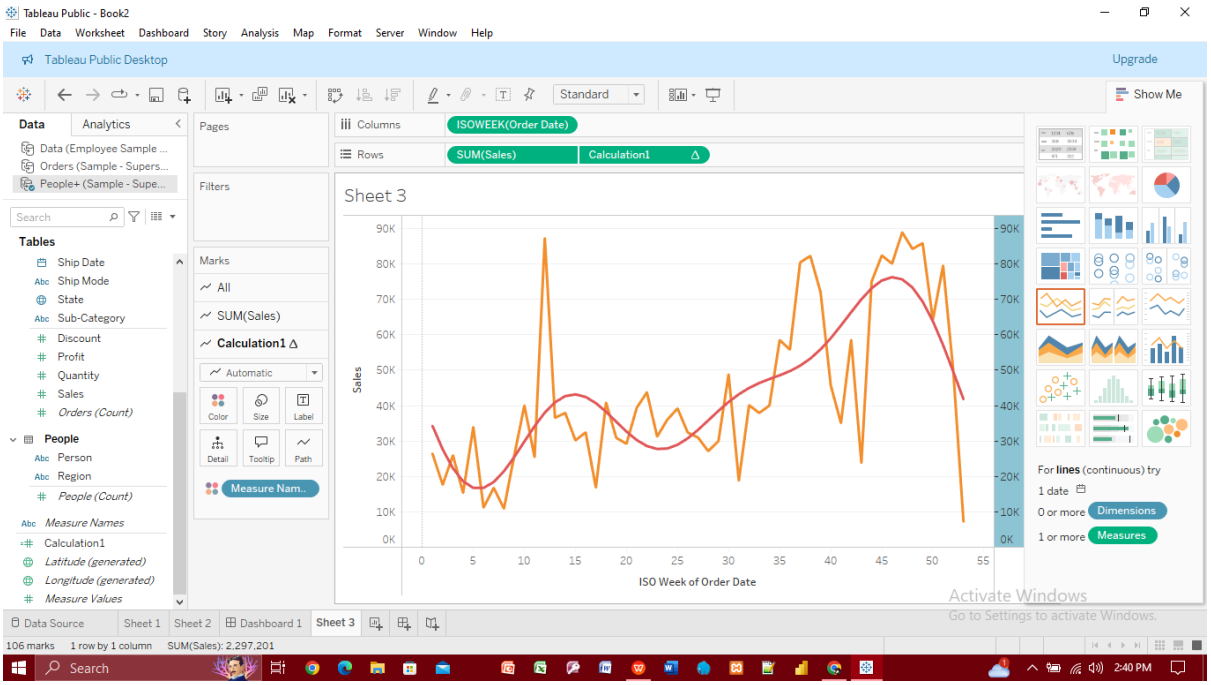
An image also shows the top-paid earners and their departments.



An image showing the predicted future sales of a company, with dual axis without synchronization

I used Tableau to visualize and analyze a sales dataset to predict the company's future sales. This involved:

- **Sales Data Analysis:** Examining historical sales data to identify trends and patterns.
- **Regression Modeling:** Using a regression model to forecast future sales based on historical data.

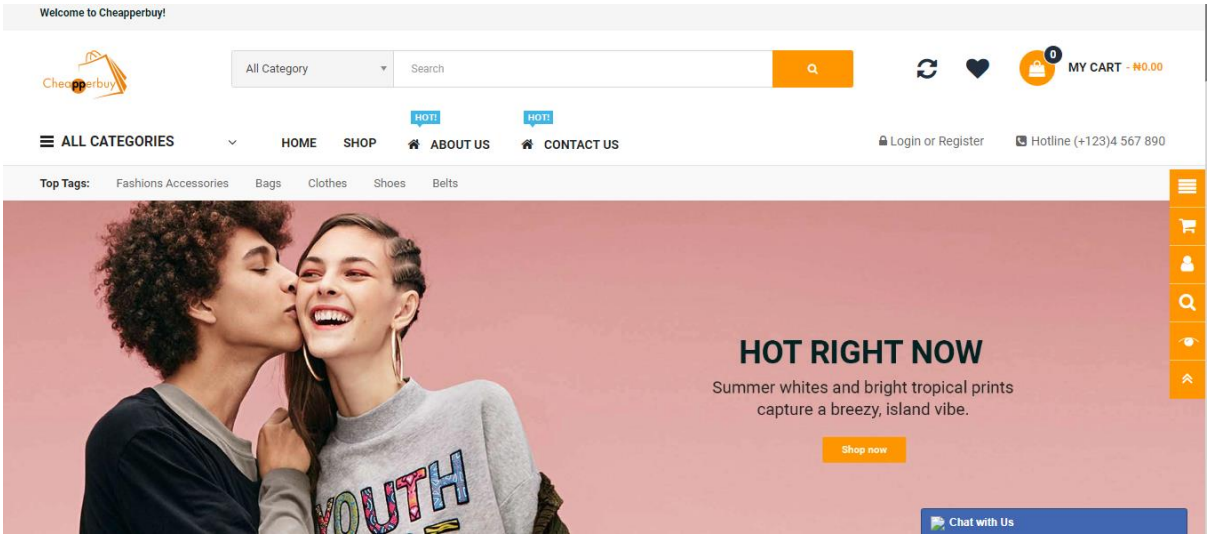


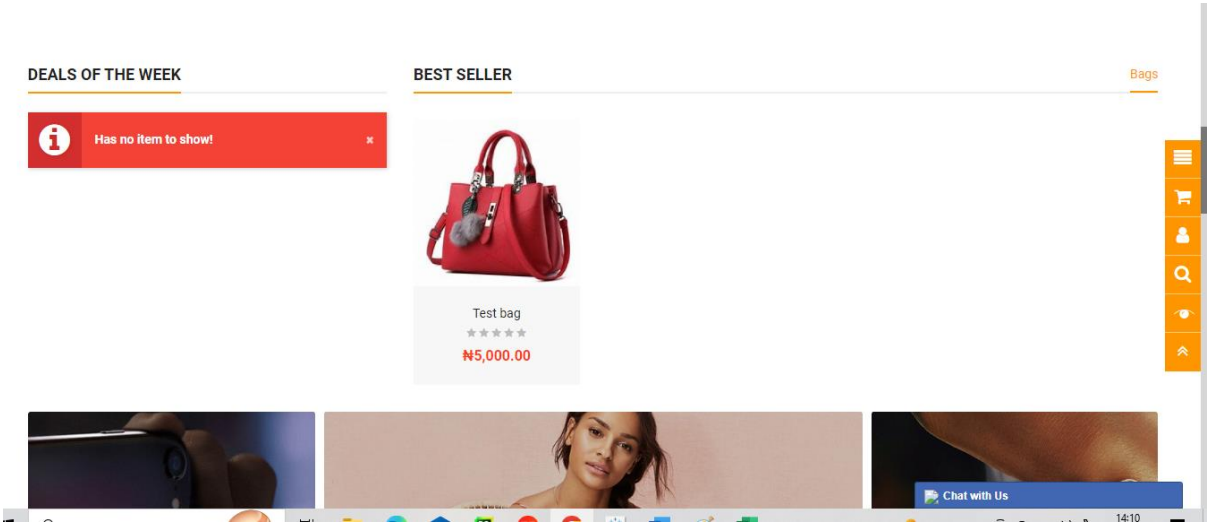
Another image showing a predicted future sale of a company with dual axis and synchronization

Toward the end of my internship, I took up a project to set up an online retail store for a small business. This involved:

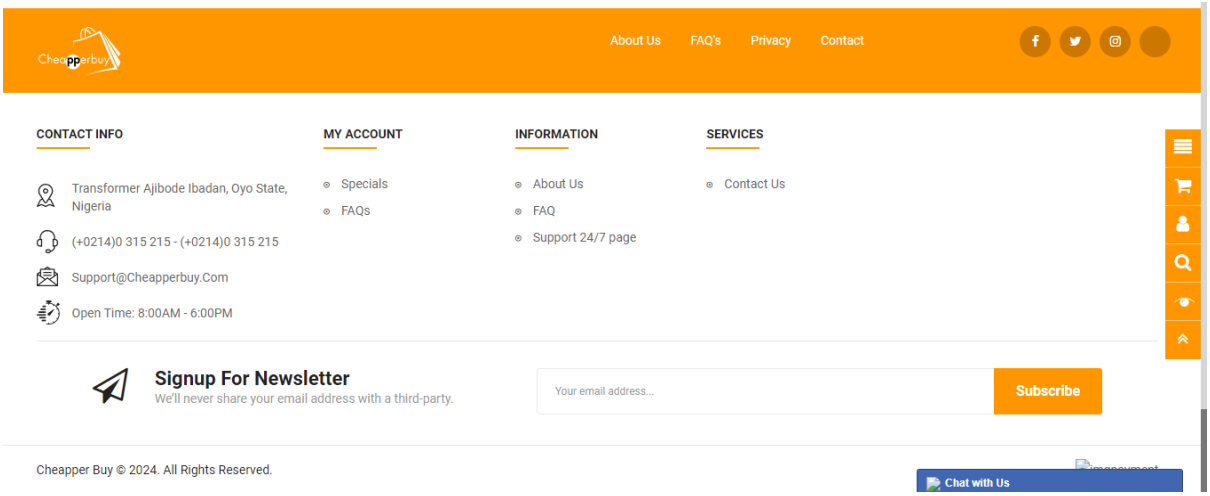
- **Requirements Gathering:** Identified business needs and target audience.
- **Design and Development:** Created the website layout, product pages, and shopping cart.
- **Payment Integration:** Set up secure payment gateways.
- **Testing and Launch:** Ensured functionality across devices and launched the site.
- **Training:** Provided training on order processing and inventory management

Pictures of the website can be found on Cheapperbuy.com





An image showing the homepage of the website



An image showing the footer of the website

3.3 TOOLS USED IN DESIGNING THE WEBSITE

- Shopify/**WooCommerce:** For e-commerce platform and CMS.
- HTML**, CSS, JavaScript:** For front-end development.
- Stripe**, Paystack:** For payment processing.
- Google **Chrome DevTools:** For testing and debugging.
- Photoshop/**Canva:** For creating and editing product images.
- Git**:** For version control and collaborations

3.4 RELEVANCE TO MY COURSE OF STUDY

As a computer science student, my six-month internship focused on data science and web design. This hands-on experience was highly relevant to several key courses I took during my academic journey, demonstrating the practical applications of theoretical knowledge gained in the classroom.

1. **Web Programming:**
    - **Relevance:** The web design aspect of my internship directly applied the concepts learned in my web programming course. I developed and maintained websites using HTML, CSS, and JavaScript, and used platforms like WordPress and Shopify.
    - **Skills Applied:** Front-end development, responsive design, user experience (UX) principles, and integration of web services.
2. **Systems Programming:**
    - **Relevance:** Systems programming concepts were crucial when working with server-side scripts and understanding the backend architecture of web applications.
    - **Skills Applied:** Shell scripting, managing server environments, and optimizing system performance for web applications.
3. **Data Structures:**
    - **Relevance:** Data science heavily relies on efficient data structures for handling large datasets. My internship involved using data structures to organize and process data for analysis.
    - **Skills Applied:** Implementing and optimizing arrays, linked lists, trees, and hash tables in Python to manage data efficiently.
4. **Database Systems:**
    - **Relevance:** Working with databases was a fundamental part of both web design and data science tasks. I designed, queried, and managed databases to store and retrieve information.
    - **Skills Applied:** SQL for querying relational databases, database design, normalization, and working with NoSQL databases like MongoDB for flexible data storage.
5. **Professional Ethics:**
    - **Relevance:** My internship underscored the importance of professional ethics in handling sensitive data and maintaining privacy and security standards.
    - **Skills Applied:** Adhering to ethical guidelines, ensuring data confidentiality, integrity, and implementing security measures to protect user information.

Conclusion

The integration of these courses into my internship experience provided a comprehensive understanding of how theoretical knowledge is applied in real-world scenarios. Each course contributed significantly to my ability to perform tasks efficiently and effectively, preparing me for a future career in data science and web development. This synergy between academic learning and practical application underscores the value of a holistic education in computer science.

CHAPTER 4

4.0 CHALLENGES ENCOUNTERED

During industrial training, they provided me with a glimpse of post-graduation life, marked by a mix of challenges and moments of ease.

The difficulties I encountered during my industrial training were numerous, reflecting the real-world complexities that young professionals in Nigeria often face. Some of the challenges included:

Internet Accessibility and High Data Costs: One major hurdle was restricted weekend internet access and the high cost of data. Accessing the internet during these times was expensive and limited, prompting us to rely on mobile data plans from providers like Etisalat and MTN, which often offered poorer network quality compared to primary providers like Spectranet.

Unreliable Power Supply: Coping with inconsistent electricity supply was a daily struggle. Many areas, including where I worked, depended heavily on fuel-powered generators throughout the day, adding to operational expenses and environmental concerns.

Rising Transportation Expenses: The removal of fuel subsidies significantly increased transportation costs. This made commuting more expensive, particularly for interns managing tight budgets.

Infrastructure Limitations: Apart from electricity, infrastructure challenges such as poor road conditions, inadequate public transport, and limited access to essential services posed additional obstacles. These factors affected punctuality and work-life balance.

Security Concerns: In some areas, security issues added to the challenges. Navigating safety concerns while commuting and during work hours required heightened awareness and caution.

Workplace Dynamics: Adapting to workplace dynamics, including learning organizational culture and navigating hierarchies, presented its own set of challenges.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

5.0 CONCLUSION

During my six-month internship, I gained practical experience in data science, immersing myself in essential tools and techniques. This hands-on experience allowed me to sharpen my skills in data analysis, statistical modelling, and machine learning applications. Working closely with data sets, I learned to extract meaningful insights and apply analytical methods to solve real-world problems.

Throughout this period, I expanded my technical proficiency and cultivated vital workplace competencies such as effective communication and collaborative teamwork. Engaging in projects involving data cleaning, visualization, and predictive modelling deepened my understanding of the data science domain.

In summary, the internship in data science proved invaluable, offering a transformative learning journey that significantly enhanced both my personal and professional growth. I wholeheartedly encourage fellow students to embrace such opportunities to prepare themselves thoroughly for a rewarding career in data science.

5.1 RECOMMENDATION

i. SIWES should schedule regular visits to student workplaces to ensure they receive adequate training and prevent any misuse of their skills. Creating a conducive learning environment and ensuring timely disbursement of student allowances are crucial for motivation and support.

ii. Effective collaboration among students and colleagues is essential for sharing real-world work experiences in their fields. This collaboration enables undergraduate students to gain valuable insights into current industry practices.

CHAPTER SIX

References

Bishop, C. M. (2013). *Pattern Recognition and Machine Learning.* Springer (India) Private Limited,.

Gareth James, D. W. (2013). *An Introduction to Statistical Learning: with Applications in R.* newyork: Springer Science & Business Media.

Hadley Wickham, G. G. (2016). *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data.* massachusetts: O'Reilly Media, Inc.

Tiffany Timbers, T. C. (2022). *Data Science: A First Introduction.* CRC Press.