



## COMPUTER SCIENCE & ENGINEERING DEPARTMENT

### Final Report (DSC429)

Analysis of work behavior among employees at the  
University of Hafr Al-Batin

### Submitted By:

Student Name	Student ID
Fadiyah alanazi	2201001182
Ghazlan alanazi	2201000995
Elham khatim	2201001444
Ebtisam falih	2201002692
Wejdan alharthi	2201001406

Supervised Dr.

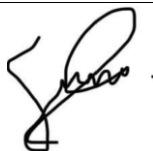

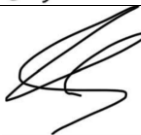

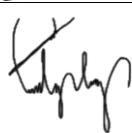
Inas Mohammed Adulraziq

May, 2024

## Declaration

The following statement clarifies that the project has been prepared and implemented by the members of the group whose names are indicated in cooperation among themselves. We hope that what has been presented has satisfied your attention.

Name and signature of every member of the group.

member name	The signature
1. Fadiyah alanazi	
2. Ghazlan alanazi	
3. Elham khatim	
4. Ebtisam falih	
5. Wejdan alharthi	

## **Approval of submission**

This project has been approved by

Supervisor

Dr. Inas Mohammed Adulrazizq \_\_\_\_\_

Head of Department

Dr. Albdulalrahman Alzahrani \_\_\_\_\_

Vice Dean

Dr. Aminat Agipola \_\_\_\_\_

## **Acknowledging**

First, I dedicate this project to Allah, who has empowered me with the strength, knowledge, and perseverance needed to undertake this academic journey. I am deeply grateful for His countless blessings and continuous guidance throughout my studies.

I extend my sincere gratitude to my parents, whose endless support and love have been the cornerstone of my success. Their sacrifices have not gone unnoticed, and this project is a testament to their unwavering faith in my abilities.

Special thanks to my supervisor, whose expert guidance and insightful critiques have significantly shaped this research. His dedication to excellence and academic rigor has not only helped refine this project but has also greatly enhanced my own learning experience.

I am also thankful to my fellow group members, whose collaboration and contributions have been invaluable. Working together has enriched this project with diverse perspectives and innovative ideas, making the journey all the more rewarding.

Appreciation is also due to the Faculty of Computer Science and Engineering for their assistance in accessing the necessary employee data. Their support has been crucial in facilitating a comprehensive analysis and ensuring the success of this study.

Lastly, I am grateful to the university I attend, which has provided an enriching academic environment that has allowed me to grow and thrive. This institution has not only been a place of learning but also a source of constant inspiration.

## **Abstract**

This project, titled “Analysis of the Work Behavior of Hafar Al-Batin University Employees,” aims to explore and analyze the intricate relationship between employee work behavior and various demographic factors such as gender, educational qualifications, rank, and job title. Employing advanced data analysis techniques and machine learning algorithms, the study seeks to understand how these factors influence employee discipline during working hours.

The research utilizes a comprehensive dataset from Hafar Al-Batin University, incorporating various employee demographics and work behavior records. Machine learning models, particularly classification algorithms, are applied to identify patterns and correlations within this data. The study also integrates a dashboard for visualizing and interpreting the results, facilitating a clearer understanding of the impacts these demographic factors have on employee discipline.

The outcomes of this research are expected to offer valuable insights for the university’s administrative and HR departments, aiding in the development of more effective policies and strategies for employee management. Furthermore, the findings could contribute to the broader academic understanding of work behavior in educational institutions, providing a template for similar studies in other contexts.

This project represents a significant step towards integrating data-driven approaches in the analysis of organizational behavior, highlighting the potential of machine learning and data analytics in human resource management and organizational studies.

## Table of Content

<b>Declaration.....</b>	<b>I</b>
<b>Approval of submission .....</b>	<b>II</b>
<b>Acknowledging.....</b>	<b>III</b>
<b>Abstract.....</b>	<b>IV</b>
<b>Chapter1: introduction.....</b>	<b>1</b>
1.1 General introduction.....	1
1.2 Importance of Study.....	1
1.3 Problem Statement.....	1
1.4 Aims and Objectives.....	2
1.5 Scope and Limitations of study.....	2
1.6 Outline of Study.....	3
<b>Chapter 2: Literature Review.....</b>	<b>5</b>
2.1 Literature Review.....	5
2.2 Previous Research.....	7
2.3 Empirical Methods.....	7
2.4 Relevance to Current Study.....	8
2.5 Hypotheses Development .....	8
<b>Chapter3: Methodology and work plan .....</b>	<b>10</b>
3.1 Overview of Project Work Plan .....	10
3.2 Environment Setup .....	11
3.3 ML/Deep Learning Model .....	12
3.5 Dataset Evaluation .....	13
3.6 Data preprocessing .....	14
3.7 Training for ML/ Deep Learning Model .....	29
3.8 Evaluation Metrics of Prediction Model .....	30
3.9 Project Planning and Resource Allocation .....	31
<b>Chapter 4: Results .....</b>	<b>33</b>
<b>Chapter 5: Conclusion .....</b>	<b>43</b>

5.1 Conclusions .....	43
5.2 Recommendations .....	43
References .....	44
<b>Appendices A .....</b>	<b>45</b>
<b>Appendices B .....</b>	<b>46</b>
<b>Appendices C .....</b>	<b>51</b>
<b>Appendices D .....</b>	<b>56</b>
<b>Appendices E .....</b>	<b>63</b>
<b>Appendices F .....</b>	<b>66</b>

## List of Figures

Figure 1 : Change The Data Language 1.....	14
Figure 2 : Change The Data Language 2 .....	14
Figure 3: Merge 2 Dataframes.....	15
Figure 4 : Data Information .....	15
Figure 5 : Data Describtion.....	16
Figure 6 : Data Cheeck.....	16
Figure 7 : Fill Missing Values.....	17
Figure 8 : Double Check If Filling The Nulls.....	17
Figure 9 : Data Before Delete The Space.....	17
Figure 10 : Removing The Space.....	18
Figure 11 : Remove Duplications.....	18
Figure 12 : Erxplor Data.....	19
Figure 13: code of Analyzing based on academic qualification.....	20
Figure 14:Output of the analysis code on the academic qualification.....	20
Figure 15:code of Comparison between males and females .....	21
Figure 16: output of the comparison between males and females.....	21
Figure 17:Code of the The relationship between job rank.....	22
Figure 18: result of the relationship between job rank.....	22
Figure 19: Code of The effect of the job on work discipline .....	23
Figure 20:output of The effect of the job on work discipline .....	23
Figure 21:Code of attendance and departure data across months.....	24
Figure 22:output of attendance and departure data across months.....	24
Figure 23: Code of attendance and departure data across days of the week .....	25
Figure 24: output of attendance and departure data across days of the week.....	25
Figure 25:Code of punctuality during the days of the week for all job ranks.....	26



Figure 26 : output of punctuality during the days of the week for all job ranks.....	26
Figure 27:Code of punctuality across months for both genders.....	27
Figure 28:output of punctuality across months for both genders.....	27
Figure 29:Code of the Show the trend on each month based on count of entries...	28
Figure 30:output of the Show the trend on each month based on count of entries...	28
Figure 31 :data splitting.....	29
Figure 32 : Random Forest.....	29
Figure 33 :Result of prediction model evaluation metrics.....	31
Figure 34:result of regularity of attendance and departure based on academic qualification.....	34
Figure 35:result attendance and departure patterns between males and females...	35
Figure 36:result job rank and employees' commitment to working hours.....	35
Figure 37:result extent to which the job affects discipline at work.....	36
Figure 38:result attendance and departure data across months.....	37
Figure 39:result attendance and departure data across days of the week .....	38
Figure 40:result punctuality during the days of the week for all job ranks.....	39
Figure 41:result punctuality across months for both genders .....	39
Figure 42:result trend on each month based on count of entries .....	40
Figure 43 :Dashboard .....	41
Figure 44: Gantt Chart.....	65

## **List of Tables**

Table 1: Project Plan.....	10
Table 2:Timetable .....	63
Table 3:Team and Roles .....	64

## **List Abbreviations**

- ML - Machine Learning
- EDA - Exploratory Data Analysis
- RF - Random Forest (a machine learning algorithm)
- KNN - K-Nearest Neighbors (a machine learning algorithm)
- ID - Identifier
- HR - Human Resources
- AI - Artificial Intelligence
- CSV - Comma-Separated Values (file format)
- API - Application Programming Interface
- SQL - Structured Query Language
- DB - Database
- UI - User Interface
- UX - User Experience
- IT - Information Technology
- GDPR - General Data Protection Regulation
- ANOVA - Analysis of Variance (statistical method)
- ROC - Receiver Operating Characteristic (in the context of model evaluation)
- AUC - Area Under the Curve (used with ROC)

## Chapter 1 : Introduction

## **Chapter1: introduction**

### **1.1 General introduction:**

The study of employee behavior within university settings is crucial for optimizing administrative and academic operations, fostering a productive work environment, and enhancing employee satisfaction. This research focuses on analyzing several factors that influence employee behavior at the University of Hafr Al-Batin, utilizing a data-driven approach to uncover underlying patterns and determinants.

### **1.2 Importance of Study :**

Understanding the dynamics of employee behavior is vital for higher education institutions that strive to maintain an efficient and harmonious workplace. This study is particularly significant as it addresses the need for a comprehensive analysis of employee behavior in the context of a Saudi Arabian university, which is often underrepresented in global research. The findings can provide valuable insights into the management strategies that can be employed to enhance employee performance and job satisfaction.

### **1.3 Problem Statement:**

Despite the critical role of employee behavior in the success of academic institutions, there is a gap in systematic and data-driven analysis concerning how various demographic and job-related factors influence such behaviors in the context of universities in the Middle East, particularly Saudi Arabia. This gap hinders the ability of university administrators to implement effective strategies tailored to their unique organizational culture and workforce.

#### **1.4 Aims and Objectives :**

The primary aim of this project is to analyze and understand the factors influencing employee behavior at the University of Hafr Al-Batin. The specific objectives include:

- To identify key demographic and job-related factors affecting employee behavior.
- To utilize statistical and machine learning techniques to analyze these factors.
- To provide recommendations based on the analysis that can help improve employee management and policymaking.
- Identifying key factors that influence employee behavior.
- Understanding the relationship between these factors and overall employee performance and satisfaction.
- Providing data-driven recommendations to the university's management for policy and strategy development.

#### **1.5 Scope and Limitations of study:**

This study will focus on analyzing available data from the University of Hafr Al-Batin's employee records, considering variables such as gender, age, educational background, job title, and department. While aiming for a comprehensive analysis, the project acknowledges limitations in data availability and the scope of factors considered. The findings are intended to offer insights and recommendations specific to the University of Hafr Al-Batin but may also be relevant to similar institutions seeking to enhance their understanding of employee behavior.

## **1.6 Outline of Study:**

The study begins with an introduction outlining the research's background, problem statement, objectives, and scope. It progresses into a literature review that discusses existing literature on employee behavior in university settings, identifying theoretical frameworks and research gaps. The methodology section explains the research design, data collection methods, and analytical techniques, including statistical analysis and machine learning. Findings are presented in the results chapter with detailed data analysis and visual representations. The discussion interprets these findings in the context of the reviewed literature, discussing their implications. The study concludes with a summary of conclusions and recommendations for university administrators and suggestions for future research. References and appendices provide citations and supplementary material to support the research integrity and detail.

## Chapter 2: Literature Review



## **Chapter 2: Literature Review**

### **2.1 Literature Review :**

This chapter provides a comprehensive review of the existing literature related to employee behavior within academic settings, focusing on studies that explore the influences of various demographic and job-related factors. It draws from a range of disciplines, including organizational psychology, human resources management, and business administration, to build a foundational understanding for this study.

- **Theoretical Foundations**

Research on employee behavior frequently utilizes a number of established theoretical frameworks. Herzberg's Two-Factor Theory, which distinguishes between hygiene factors and motivators, serves as a fundamental model for understanding employee satisfaction and motivation (Herzberg, Mausner, & Snyderman, 1959). Additionally, Maslow's Hierarchy of Needs is often applied to explore how unique needs from basic to advanced drive employee behaviors (Maslow, 1943).

- **Demographic Influences on Employee Behavior**

Several studies have focused on how demographic factors such as age, gender, and educational attainment impact employee behavior. For instance, a study by Smith and Robertson (2015) found that age and educational level significantly influence job satisfaction and commitment within academic environments. Gender differences in workplace behavior have also been examined, with findings suggesting that these can affect communication styles and conflict management (Johnson & Smith, 2013).

- **Organizational Culture and Employee Behavior**

The role of organizational culture in shaping employee behavior has been extensively studied. A supportive and positive organizational culture has been linked to enhanced employee morale and productivity (Kotter & Heskett, 1992). Research specific to universities indicates that an inclusive and empowering culture is crucial for fostering academic freedom and job satisfaction among faculty members (Turner & Müller, 2005).

- **Leadership Styles and Their Impact**

Leadership style is another critical factor affecting employee behavior. Transformational leadership, which involves inspiring and motivating employees, has been shown to have a positive impact on employee engagement and organizational loyalty (Bass & Avolio, 1994). In the context of higher education, effective leadership has been correlated with lower turnover rates and higher job satisfaction among academic staff (Leith wood & Jantzi, 2000).

- **Empirical Studies Specific to Higher Education**

Several empirical studies have directly addressed employee behavior in higher education settings. For example, a longitudinal study by García-Morales, Jiménez-Barrionuevo, and Gutiérrez-Gutiérrez (2012) demonstrated that innovation in university management practices positively influences employee behavior and institutional performance. Further, research by Nguyen, Taylor, and Bradley (2016) explored the relationship between job autonomy and job satisfaction among university staff, finding significant positive correlations.

The existing literature provides valuable insights into the factors influencing employee behavior in academic institutions. However, there remains a gap in the application of these theories and findings within the specific cultural and administrative context of Middle Eastern universities, such as the University of Hafr Al-Batin. This study aims to bridge that gap by applying these broad concepts to a focused analysis of employee behavior within this unique setting.

- **Organizational Behavior:** Review theories that explain employee behavior within organizations, such as Maslow's Hierarchy of Needs and Herzberg's Two-Factor Theory. Discuss how these theories are applicable in understanding the motivations and satisfactions of university employees.
- **Predictive Analytics:** Explore the foundations of using statistical and machine learning methods in predicting human behavior. This includes a review of common algorithms used in predictive modeling and their effectiveness in various organizational settings.

## **2.2 Previous Research :**

- **Employee Behavior Studies:** Summarize previous studies focused on employee behavior in the educational sector, highlighting findings related to factors like job satisfaction, performance, and retention. Include studies from both Western and Middle Eastern contexts to provide a comprehensive view.
- **Data-Driven Approaches in HR:** Discuss the application of data science in human resources, particularly in behavior analysis and prediction. Highlight case studies where machine learning has been effectively implemented to improve HR decisions.

## **2.3 Empirical Methods:**

- **Machine Learning Techniques:** Delve into specific machine learning models commonly used for classification and prediction in behavioral data, such as logistic regression, decision trees, random forests, and neural networks. Discuss their pros and cons based on existing literature.

- **Data Handling and Ethics:** Address the challenges and ethical considerations of handling employee data, including privacy concerns, data security, and the implications of predictive modeling on employee privacy.

## **2.4 Relevance to Current Study:**

- **Applying Machine Learning in University Settings:** Discuss how the methodologies reviewed can be applied to the context of Hafr Al-Batin University. Consider the cultural, organizational, and technological factors that might influence the implementation and outcomes of such analyses.
- **Gap Analysis:** Identify gaps in the current research landscape, particularly in studies involving Middle Eastern educational institutions, and articulate how this project aims to fill these gaps.

## **2.5 Hypotheses Development :**

- Based on the literature review, develop specific hypotheses that the project will test. For example, hypotheses could relate to the impact of job rank or educational qualification on employee punctuality and overall behavior.

## **Chapter 3 : Methodology and work plan**

## Chapter3: Methodology and work plan

This chapter details the methodology and work plan used in this data science project aimed at analyzing employee behavior at the University of Hafr Al-Batin. The methods include data collection, preprocessing, machine learning techniques, and the specific tools and software employed to manage and analyze the data.

### 3.1 Overview of Project Work Plan:

The project work plan involves several stages:

WEEK	TASKS
1	Summary report
2	Summary report
3	Project Plan
4	Design document with Machine Learning Algorithm(s)
5	Follow document design using machine learning
6	Relevant Data Science/ML Modelling (Lab work)
7	Discussion
8	Lab-based Review
9	Lab-based Review
10	First Draft of Project Report
11	Final report document
12	Final report document
13	Final report document
14	Final report document
15	Presentation / Meetings and Laboratory Progress

Table 1: Project Plan

### 3.2 Environment Setup:

The analysis was conducted using several key tools:

- **Programming Environment:** Python 3.8, an excellent choice for data science due to its robust libraries and community support.
- **Development Tools:** google colab, Jupyter Notebook, which provides an interactive coding environment ideal for data exploration and visualization.
- **Python library:** Pandas for data manipulation and analysis , and matplotlib for creating static, animated and interactive visualization and seaborn is a statical data visualization based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.
- **Google Colab :** is an improved cloud version of Jupyter Notebook dedicated to writing and running code and Notebook documents. It helps you write and execute code written in Python through your browser without the need to install any editor or program.

### **3.3 ML/Deep Learning Model :**

Through the data obtained, we created Sample Machine Learning, which is a simple sample model that explains how to use Sample Machine Learning using scikit-learn, through which we used what is known as Random Forest.

Since our data does not strongly support machine learning, this model was the best fit after conducting several experiments using other models.

scikit-learn it is a library in the Python language that provides many supervised and unsupervised learning algorithms. It is built on some technologies that you may already be familiar with, such as NumPy, pandas, and Matplotlib. It is characterized by being a consistent interface for machine learning models.

Random Forest is a powerful and versatile supervised machine learning algorithm that grows and combines multiple decision trees to create a “forest” that works on the principle of working in parallel rather than sequentially, which in turn helps improve results and connect to higher accuracy.



### 3.5 Dataset Evaluation :

In the initial phase of our project, we submitted a request to the university to access the employment data for use in our project. This request was approved, and we were granted access to the data, which consisted of two files. The first file contained employee information, and the second comprised their fingerprint data. We merged these two files using a unique identifier, resulting in a consolidated dataset. This comprehensive data file now contains information organized into 10 columns and encompasses a total of 73,035 entries.

The dataset includes the following column names, which facilitate detailed analyses and categorization of the employee data:

- **ID:** A unique identifier assigned to each employee.
- **Date Time:** The timestamp associated with each recorded entry.
- **Type:** Classification of entries, detailing the nature of the data recorded.
- **M:** A code representing specific metrics or categories relevant to the employee data.
- **Sex:** The gender of the employee.
- **Formation:** Details pertaining to the employee's formative background or grouping within the organization.
- **Rank:** The official rank of the employee within the organization.
- **Class:** Classification based on employment or administrative criteria.
- **Job Title:** The official title of the employee's position.
- **Qualification:** The academic or professional qualifications held by the employee.

We will proceed to analyze this dataset and present the findings. The results will be visualized through an interactive dashboard, which will allow for dynamic interaction with the data. This visualization will enable stakeholders to easily comprehend the trends and patterns within the employee data, facilitating informed decision-making and strategic planning.

### 3.6 Data preprocessing:

In this chapter, we will review the most prominent processing operations that were carried out on the data, which in turn helped in making the data clearer and the ability to perform the visualization process and draw conclusions clearly without any problems that might hinder the process.

#### ➤ Change the data language :

After receiving and reviewing the data, it was found that some columns are in Arabic and others are in English. Therefore, all columns have been converted to English to ensure that the data is more clearly understood during the processing and visualization process.

المؤهل	المسمى الوظيفي	الدرجة	المرتبة	التشكيل	الجنس	ID	م
ثانوي	مراقب امن وسلامة	الدرجة 09	المرتبة السابعة	سلم الموظفين العام	ذكر	٤٠١٠٠٦٣٥	1
ثانوي	مراقب امن وسلامة	الدرجة 08	المرتبة الخامسة	سلم الموظفين العام	انثى	٤٠١٠٠٤٢٩	2
دبلوم بعد البكالوريوس	مساعد اداري	الدرجة 11	المرتبة الثامنة	سلم الموظفين العام	انثى	٤٠١٠٠٥١٣	3
بكالوريوس	مراقب امن وسلامة	الدرجة 08	المرتبة السابعة	سلم الموظفين العام	انثى	٤٠١٠٠٣٨٣	4
بكالوريوس	أخصائي علاقات عامة متقدم	الدرجة 15	المرتبة العاشرة	سلم الموظفين العام	ذكر	٤٠١٠٠٨٤٠	5
بكالوريوس	أمين صندوق	الدرجة 11	المرتبة الثامنة	سلم الموظفين العام	انثى	٤٠١٠٠٣٢٣	6
بكالوريوس	مساعد اداري	الدرجة 11	المرتبة الثامنة	سلم الموظفين العام	انثى	٤٠١٠٠٣٢٥	7
	مساعد اداري	الدرجة 11	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٣٧٨	8
بكالوريوس	مساعد اداري	الدرجة 05	المرتبة الثامنة	سلم الموظفين العام	انثى	٤٠١٠٠٣٦١	9
بكالوريوس	فني مختبر	الدرجة 08	المرتبة السابعة	سلم الموظفين العام	انثى	٤٠١٠٠٣٥٦	10
بكالوريوس	مشغل أجهزة مكتبية	الدرجة 11	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٣٨٩	11
دبلوم بعد الثانوي	مطور برامج	الدرجة 12	المرتبة العاشرة	سلم الموظفين العام	ذكر	٤٠١٠٠٦١٣	12
	مساعد اداري	الدرجة 10	المرتبة السادسة	سلم الموظفين العام	ذكر	٤٠١٠٠٦٣٦	13
متوسط	مساعد إداري ممارس ثاني	الدرجة 07	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٥٣٤	14
ثانوي	مساعد اداري	الدرجة 05	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٥٢٣	15
	مراقب طلبة	الدرجة 11	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٣٥٨	16

Figure 1 : Change The Data Language 1

ID	DateTime	Type
٤٠١٠٠٦٣٥	٣١/١٢/٢٠٢٢ ١٩:٠٢:٢٠	OUT
٤٠١٠٠٦٣٨	٣١/١٢/٢٠٢٢ ١٨:٥٩:٠١	OUT
٤٠٨٠٠٣٤٨	٣١/١٢/٢٠٢٢ ١٧:٣٨:١٧	OUT
٤٠١٠٠٥٥٥	٣١/١٢/٢٠٢٢ ١٦:٢٩:٢٥	OUT
٤٠١٠٠٦٥٩	٣١/١٢/٢٠٢٢ ١٤:٠١:٢٥	OUT
٤٠١٠٠٦٣٥	٣١/١٢/٢٠٢٢ ١٣:٠٣:٠٢	OUT
٤٠١٠٠٤٦٦	٣١/١٢/٢٠٢٢ ١٣:٠١:١٥	OUT
٤٠١٠٠٥٥٥	٣١/١٢/٢٠٢٢ ١١:٣٧:٠٩	OUT
٤٠١٠٠٦٣٥	٣٠/١٢/٢٠٢٢ ١٨:٥٠:٢٣	OUT
٤٠١٠٠٦٤٤	٣٠/١٢/٢٠٢٢ ١٨:٣٩:٤١	OUT
٤٠١٠٠٦٣٥	٣٠/١٢/٢٠٢٢ ١٣:٠٩:٠١	OUT

Figure 2 : Change The Data Language 2

### ➤ Merge 2 Dataframes :

Two files of data were obtained, one related to employees' fingerprints and the other to tracking departures. Both files have a common column, which is the ID. Accordingly, both files are merged into one file to facilitate analysis and processing.

```
file2.columns = file2.columns.str.strip()
file1.columns = file1.columns.str.strip()

data = pd.merge(file1, file2 ,on = 'ID')
data.head()
```

	ID	DateTime	Type	M	Sex	Formation	Rank	Class	Job title	qualification
0	40100635	2022-12-31 19:02:20	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary
1	40100635	2022-12-31 13:03:02	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary
2	40100635	2022-12-30 18:50:23	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary
3	40100635	2022-12-30 13:09:01	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary
4	40100635	2022-12-29 19:05:26	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary

Figure 3: Merge 2 Dataframes

### ➤ Exploratory data analysis (EDA) :

Exploratory data analysis was conducted to determine the number of columns and rows, in addition to knowing the column names and the type of each column, which in turn will provide more detailed information about the data.

```
data.shape
(73035, 10)

data.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 73035 entries, 0 to 73034
Data columns (total 10 columns):
#   Column          Non-Null Count  Dtype
---  -
0   ID               73035 non-null  int64
1   DateTime         73035 non-null  datetime64[ns]
2   Type             73035 non-null  object
3   M                73035 non-null  object
4   Sex              73035 non-null  object
5   Formation        73035 non-null  object
6   Rank             73035 non-null  object
7   Class            73035 non-null  object
8   Job title        73035 non-null  object
9   qualification     62133 non-null  object
dtypes: datetime64[ns](1), int64(1), object(8)
memory usage: 5.6+ MB
```

Figure 4 : Data Information

### ➤ Statistics Summary :

A statistical summary was made to know the average, max, count, etc. about the data.

```
data.describe()
```

	ID	DateTime
count	7.303500e+04	73035
mean	4.010057e+07	2022-07-19 20:26:29.293927680
min	4.010032e+07	2022-01-02 11:31:58
25%	4.010043e+07	2022-04-17 15:50:29
50%	4.010057e+07	2022-08-01 14:15:12
75%	4.010070e+07	2022-10-24 14:15:33.500000
max	4.010085e+07	2022-12-31 19:02:20
std	1.501584e+02	NaN

Figure 5 : Data Description

### ➤ Check nulls count :

In the column known as the qualifier, we notice that 10,902 pieces of data have an unknown qualifier, so these columns are named “unknown” because it is not correct and logical to put another qualifier to them when they are unknown, because it will affect the data and its credibility.

- The step of checking the number of null values:

```
data.isna().sum()
```

ID	0
DateTime	0
Type	0
M	0
Sex	0
Formation	0
Rank	0
Class	0
Job title	0
qualification	10902
dtype:	int64

Figure 6 : Data Cheeck

- Fill missing values in qualification column with unknown:

```
data['qualification'] = data['qualification'].fillna('unknown')
data['qualification'].unique()

array(['secondary', 'unknown', "Bachelor's", 'middle', 'Primary',
      'Post-secondary diploma', "Master's",
      "Diploma after bachelor's degree", 'Ph.D', 'Literacy'],
      dtype=object)
```

Figure 7 : Fill Missing Values

- Double check if filling the nulls working correctly:

```
data.isna().sum()
```

```
ID          0
DateTime    0
Type        0
M           0
Sex         0
Formation   0
Rank        0
Class       0
Job title   0
qualification 0
dtype: int64
```

Figure 8 : Double Check If Filling The Nulls

### ➤ Remove unwanted spaces in values:

What is known as strip () was used to remove and delete unwanted spaces

- Before deleting the space :

```
data['qualification'].unique()

array([' secondary', nan, " Bachelor's", ' middle', ' Primary',
      ' Post-secondary diploma', " Master's",
      " Diploma after bachelor's degree", ' Ph.D', ' Literacy'],
      dtype=object)
```

Figure 9 : Data Before Delete The Space

- After using strip () and removing the space :

```
data = data.applymap(lambda x: x.strip() if isinstance(x, str) else x)

data['qualification'].unique()

array(['secondary', nan, "Bachelor's", 'middle', 'Primary',
      'Post-secondary diploma', "Master's",
      "Diploma after bachelor's degree", 'Ph.D', 'Literacy'],
      dtype=object)
```

Figure 10 : Removing The Space

#### ➤ Check and remove duplications :

The total number of duplicates after exploring them reached 2094, as the total data with the presence of duplicates amounted to about 73035 and after removing them, the total became 70941.

```
data.duplicated().sum()
```

2094

```
print('Data size with duplicaion: ', data.shape[0])
data = data.drop_duplicates()
print('Data size without duplicaion: ', data.shape[0])
```

Data size with duplicaion: 73035

Data size without duplicaion: 70941

Figure 11 : Remove Duplications

➤ **Explore the different values frequency in some columns :**

This step was performed to find out the most frequent columns in terms of 'Sex', 'Qualification', 'Class' and 'Rank'.

```
cols = ['Sex' , 'qualification' , 'Class' , 'Rank']

for col in cols:
    print(data[col].value_counts())
    print('*'*100)
```

Sex  
feminine 37451  
male 33490  
Name: count, dtype: int64  
\*\*\*\*\*

qualification  
Bachelor's 34842  
unknown 10585  
secondary 8835  
Post-secondary diploma 7827  
Diploma after bachelor's degree 3395  
Master's 2830  
middle 1376  
Primary 948  
Literacy 183  
Ph.D 120  
Name: count, dtype: int64  
\*\*\*\*\*

Class  
Class 07 12494  
Class 08 10929  
Class 09 9706  
Class 11 7724  
Class 04 6977  
Class 06 6909  
Class 05 5670  
Class 10 2450  
Class 12 1879  
Class 03 1787  
Class 15 1524  
Class 13 1181  
Class 02 1167  
Class 14 544  
Name: count, dtype: int64  
\*\*\*\*\*

Rank  
Seventh place 21038  
Sixth place 20219  
Eighth place 14552  
Ninth rank 5186  
Fifth place 3778  
Tenth place 3746  
Fourth place 1803  
Eleventh place 410  
Third place 209  
Name: count, dtype: int64  
\*\*\*\*\*

Figure 12 : Erxplor Data

## Data visualization

- Analyzing the regularity of attendance and departure based on academic qualification

```
# Calculate the count of each category in the 'qualification' column
qualification_counts = data['qualification'].value_counts()

# Get the categories sorted by count
qualification_order = qualification_counts.index.tolist()

plt.figure(figsize=(10, 6))
sns.countplot(x='qualification', hue='Type', data=data, palette='Set2', order=qualification_order)
plt.title('Punctuality by Educational Qualification', fontsize=16)
plt.xlabel('Educational Qualification', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(rotation=45, ha='right')
plt.legend(title='Type', title_fontsize='14')
plt.tight_layout()
plt.show()
```

Figure 13: code of Analyzing based on academic qualification

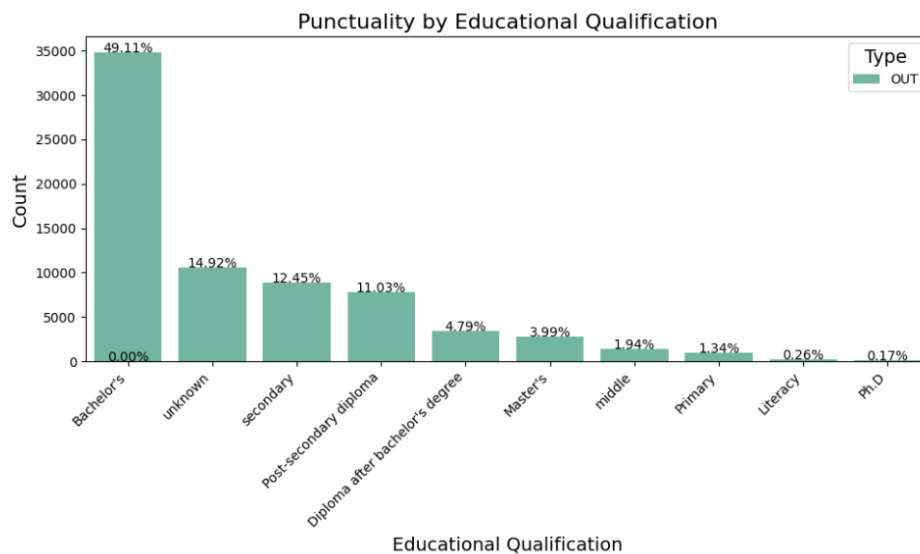


Figure 14: Output of the analysis code on the academic qualification



- **Comparison of attendance and departure patterns between males and females:**

```
: plt.figure(figsize=(8, 6))
  sns.countplot(x='Sex', hue='Type', data=data, palette='Set2')
  plt.title('Punctuality by Gender', fontsize=16)
  plt.xlabel('Gender', fontsize=14)
  plt.ylabel('Count', fontsize=14)
  plt.legend(title='Type', title_fontsize='14')
  plt.tight_layout()
  plt.show()
```

Figure 15:code of Comparison between males and females

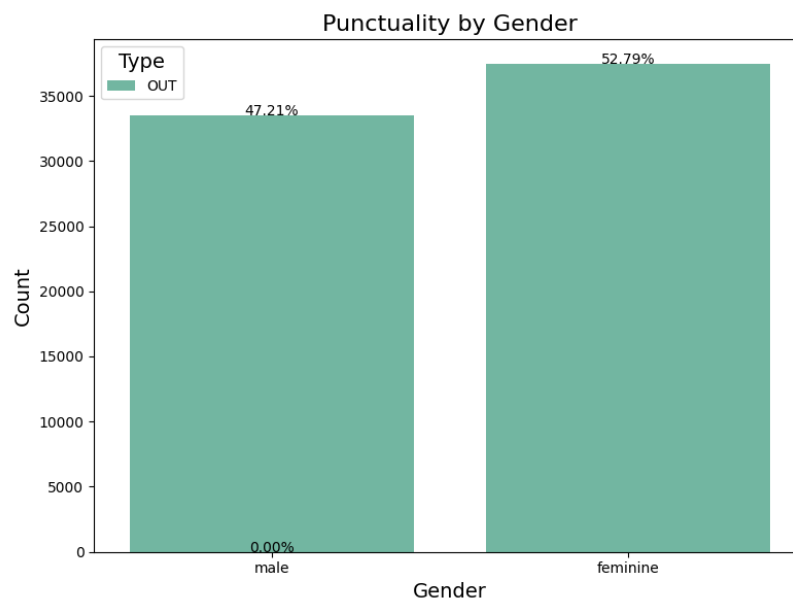


Figure 16: output of the comparison between males and females

- Analyzing data to see the relationship between job rank and employees' commitment to working hours :

```
# Calculate the count of each category in the 'Rank' column
rank_counts = data['Rank'].value_counts()

# Get the categories sorted by count
rank_order = rank_counts.index.tolist()

plt.figure(figsize=(12, 6))
sns.countplot(x='Rank', hue='Type', data=data, palette='Set2', order=rank_order)
plt.title('Punctuality by Job Rank', fontsize=16)
plt.xlabel('Job Rank', fontsize=14)
```

Figure 17:Code of the The relationship between job rank

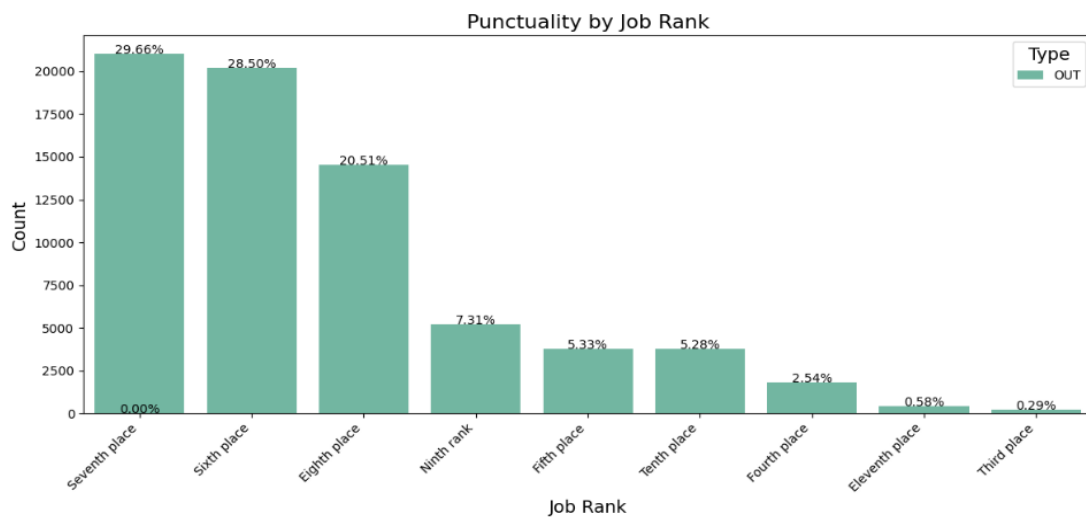


Figure 18: result of the relationship between job rank

- Examining the extent to which the job affects discipline at work Filtering only top 10 job titles

```

: # Calculate the count of each category in the 'Job title' column
Job_title_counts = data['Job title'].value_counts()

# Get the categories sorted by count
Job_title_order = Job_title_counts.index.tolist()

# Filter only the top 10 job titles
top_10_job_titles = Job_title_order[:10]

plt.figure(figsize=(12, 8))
ax = sns.countplot(x='Job title', hue='Type', data=data, palette='Set2', order=top_10_job_titles)

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality by Job Title (Top 10)', fontsize=16)
plt.xlabel('Job Title', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(rotation=90, ha='right')
plt.legend(title='Type', title_fontsize='14')
plt.tight_layout()
plt.show()

```

Figure 19: Code of The effect of the job on work discipline

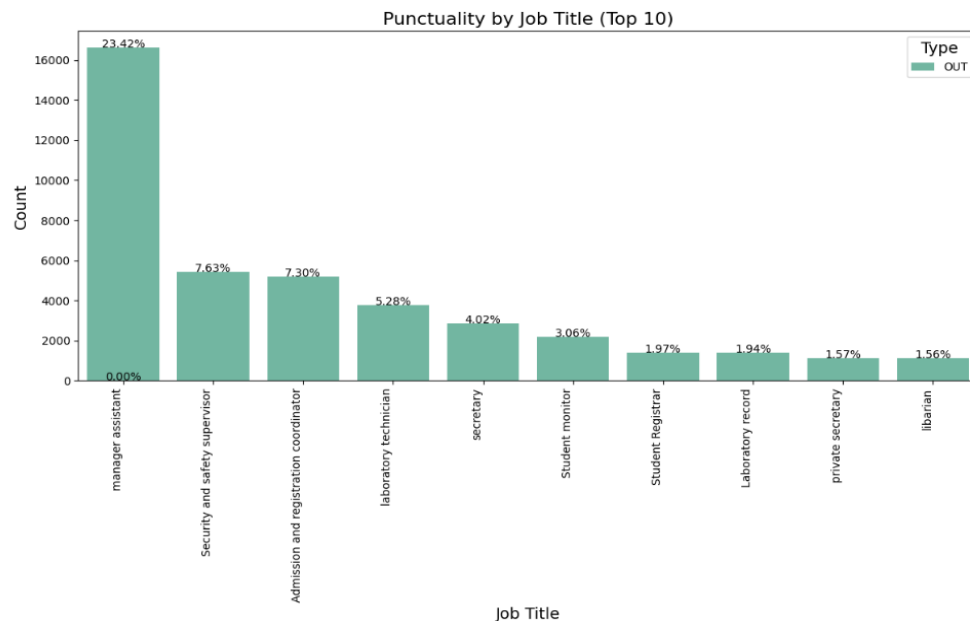


Figure 20:output of The effect of the job on work discipline

- Analyzing attendance and departure data across months to monitor patterns :

```
data['Month'] = data['DateTime'].dt.month

plt.figure(figsize=(12, 6))
ax = sns.countplot(x='Month', data=data, palette='Set2', hue='Month', legend=False)

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality Across Months', fontsize=16)
plt.xlabel('Month', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.tight_layout()
plt.show()
```

Figure 21:Code of attendance and departure data across months

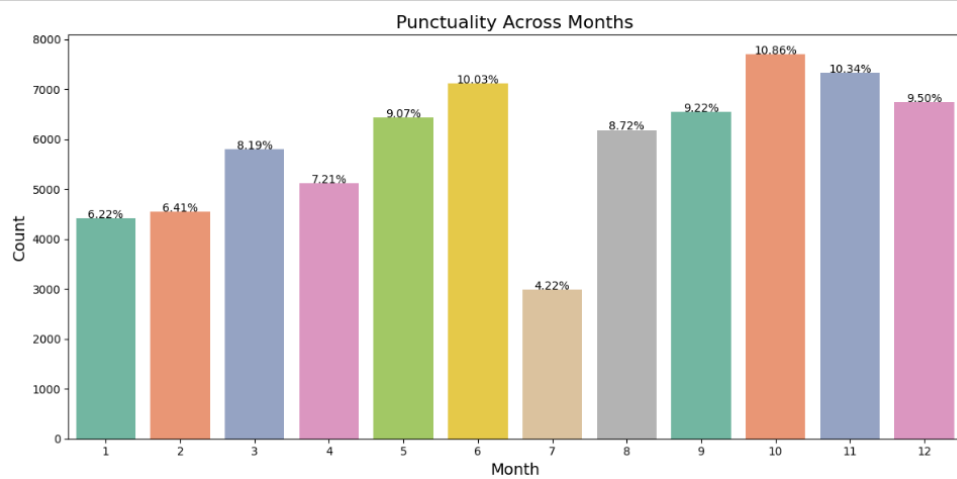


Figure 22:output of attendance and departure data across months

- Analyzing attendance and departure data across days of the week to monitor patterns :

```
data['DayOfWeek'] = data['DateTime'].dt.dayofweek

plt.figure(figsize=(10, 6))
ax = sns.countplot(x='DayOfWeek', data=data, palette='Set2', hue='DayOfWeek', legend=False)

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality Across Days of the Week', fontsize=16)
plt.xlabel('Day of the Week', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(ticks=[0, 1, 2, 3, 4, 5, 6], labels=['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun'])
plt.tight_layout()
plt.show()
```

Figure 23: Code of attendance and departure data across days of the week

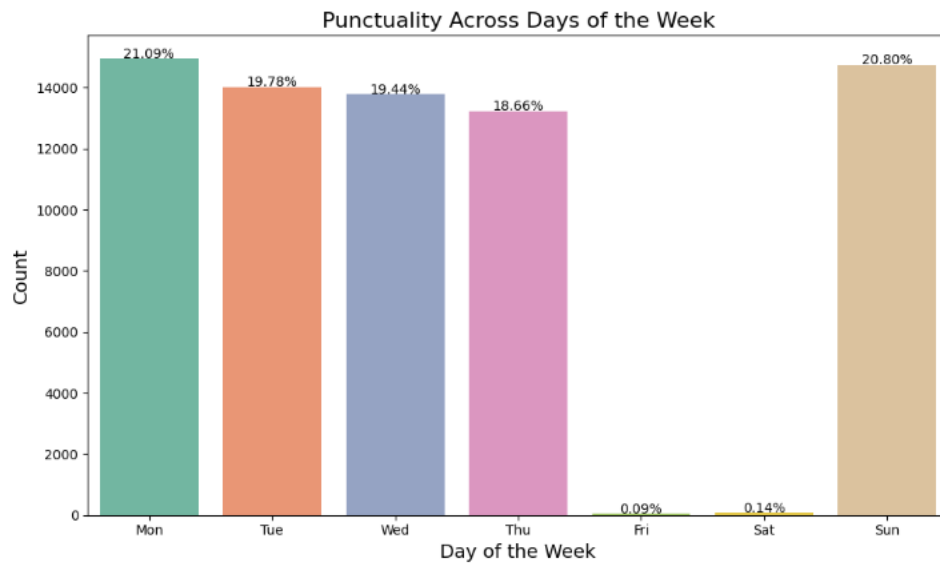


Figure 24: output of attendance and departure data across days of the week

- Discover the rate of punctuality during the days of the week for all job ranks:

```
plt.figure(figsize=(16, 6))
ax = sns.countplot(x='DayOfWeek', hue='Rank', data=data, palette='Set2')

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality Across Days of the Week by Rank', fontsize=16)
plt.xlabel('Day of the Week', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(ticks=[0, 1, 2, 3, 4, 5, 6], labels=['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun'])
plt.tight_layout()
plt.legend(title='Rank', title_fontsize='14')
plt.show()
```

Figure 25:Code of punctuality during the days of the week for all job ranks

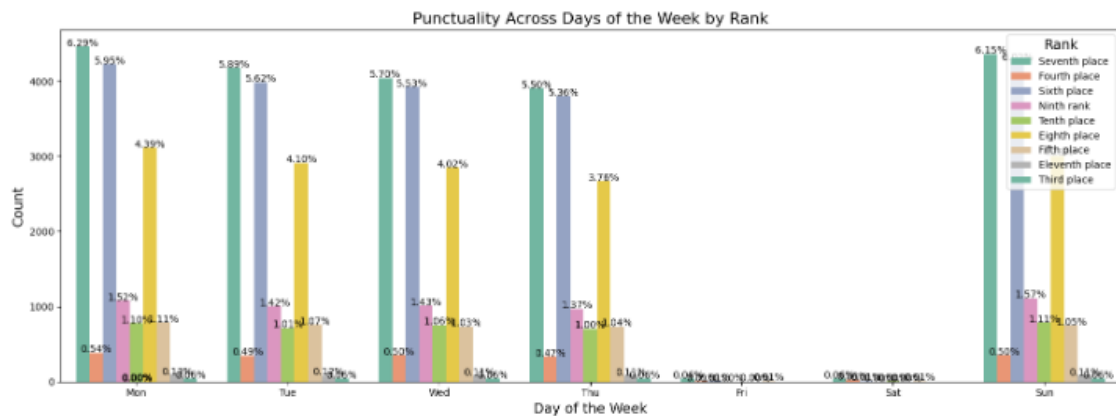


Figure 26 : output of punctuality during the days of the week for all job ranks

- Discover the rate of punctuality across months for both genders :

```
plt.figure(figsize=(12, 6))
ax = sns.countplot(x='Month', hue='Sex', data=data, palette='Set2')

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality Across Months by Sex', fontsize=16)
plt.xlabel('Month', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.tight_layout()
plt.legend(title='Sex', title_fontsize='14')
plt.show()
```

Figure 27:Code of punctuality across months for both genders

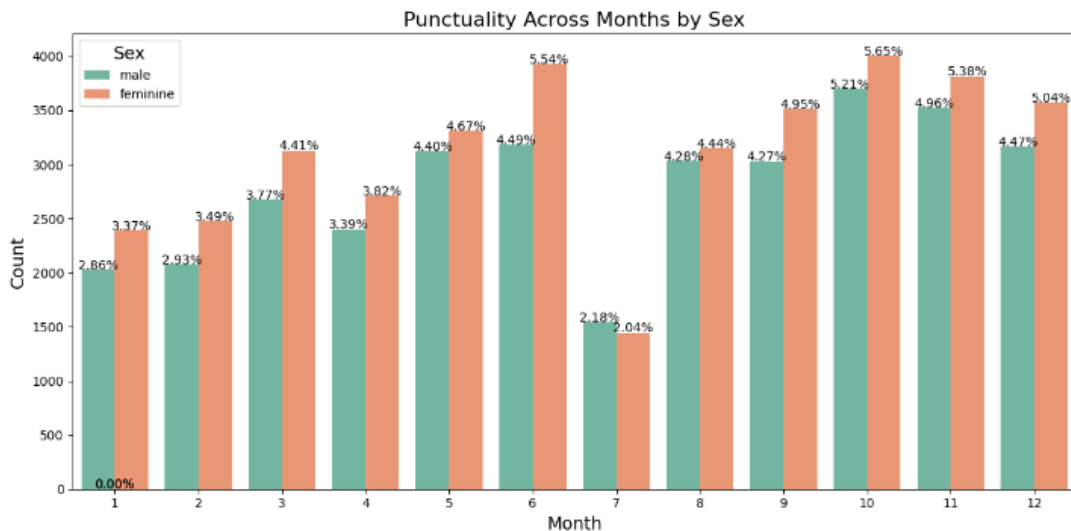


Figure 28:output of punctuality across months for both genders

- Show the trend on each month based on count of entries :

```
plt.figure(figsize=(10, 6))
data['DateTime'] = pd.to_datetime(data['DateTime'])
monthly_counts = data['Month'].value_counts().sort_index()
total_counts = monthly_counts.sum() # Total count of all data points

# Calculate percentages
percentages = (monthly_counts / total_counts) * 100

sns.lineplot(x=monthly_counts.index, y=monthly_counts.values, marker='o')

# Annotate each point with its percentage
for i, count in enumerate(monthly_counts.values):
    plt.text(monthly_counts.index[i], count, f"{percentages.iloc[i]:.2f}%", ha='center', va='bottom')

plt.title('Monthly Trends', fontsize=16)
plt.xlabel('Month', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(ticks=range(1, 13), labels=['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'])
plt.tight_layout()
plt.show()
```

Figure 29:Code of the Show the trend on each month based on count of entries

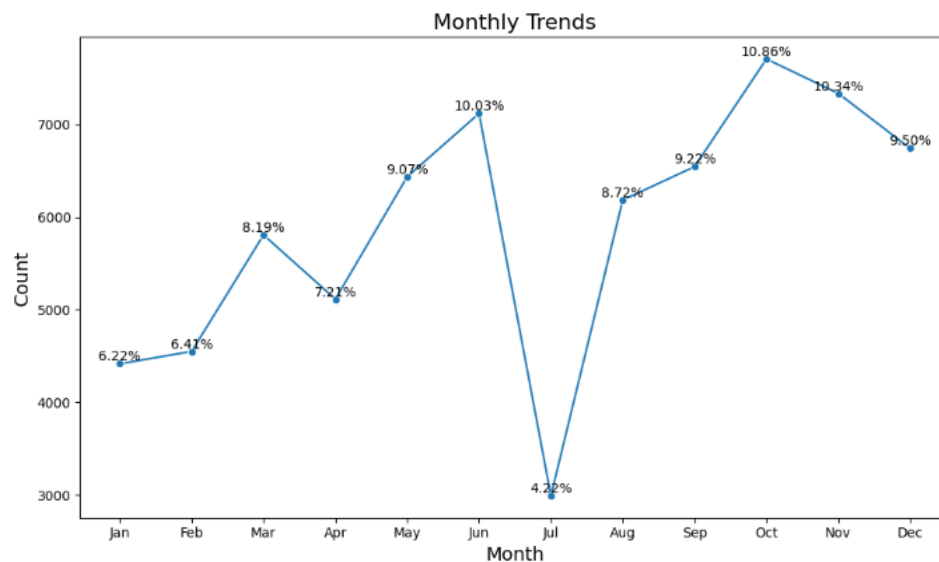


Figure 30:output of the Show the trend on each month based on count of entries



### 3.7 Training for ML/ Deep Learning Model :

The data was divided into features and objectives, training, and testing, then we used both Random Forest and K-Nearest Neighbors (KNN) models. We found that Random Forest was 0.73% better compared to K-Nearest Neighbors (KNN), which was 0.73% less efficient. 0.69.

- **Data splitting to features and target, then to training and testing :**

```
X = oversampled_data[['Sex', 'Formation', 'Job title', 'Month', 'DayOfWeek']]
y = oversampled_data['qualification']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Figure 31 :data spliting

- **Random Forest:**

```
# Training
rf_classifier = RandomForestClassifier()
rf_classifier.fit(X_train, y_train)

# Predictions
rf_predictions = rf_classifier.predict(X_test)

# Evaluation
rf_accuracy = accuracy_score(y_test, rf_predictions)
rf_precision = precision_score(y_test, rf_predictions, average='weighted')
rf_recall = recall_score(y_test, rf_predictions, average='weighted')
rf_f1_score = f1_score(y_test, rf_predictions, average='weighted')

# Results
print("Random Forest:")
print("Accuracy:", rf_accuracy)
print("Precision:", rf_precision)
print("Recall:", rf_recall)
print("F1 Score:", rf_f1_score)
```

Random Forest:  
Accuracy: 0.731535812232923  
Precision: 0.742003339080411  
Recall: 0.731535812232923  
F1 Score: 0.7258920435849853

Figure 32 : Random Forest

### 3.8 Evaluation Metrics of Prediction Model:

Since the project relies on classification models, it needed evaluation standards that were appropriate and compatible with it, the most prominent of which were used as follows:

- Accuracy: Accuracy measures the proportion of true results (both true positives and true negatives) among the total number of cases examined.

$$\text{Accuracy} = \frac{\text{True Positives}(TP) + \text{True Negatives}}{\text{Total number of samples}}$$

- Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positives. It is a measure of a classifier's exactness.

$$\text{Precision} = \frac{\text{True Positives}(TP)}{\text{True Positives}(TP) + \text{False Positives}(FP)}$$

- Recall: Recall is the ratio of correctly predicted positive observations to all observations in actual class - yes. It is a measure of a classifier's completeness.

$$\text{Recall} = \frac{\text{True Positives}(TP)}{\text{True Positives}(TP) + \text{False Negatives}(FN)}$$

- F1-score: The F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. It is a balance between Precision and Recall.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

```
# Training
rf_classifier = RandomForestClassifier()
rf_classifier.fit(X_train, y_train)

# Predictions
rf_predictions = rf_classifier.predict(X_test)

# Evaluation
rf_accuracy = accuracy_score(y_test, rf_predictions)
rf_precision = precision_score(y_test, rf_predictions, average='weighted')
rf_recall = recall_score(y_test, rf_predictions, average='weighted')
rf_f1_score = f1_score(y_test, rf_predictions, average='weighted')

# Results
print("Random Forest:")
print("Accuracy:", rf_accuracy)
print("Precision:", rf_precision)
print("Recall:", rf_recall)
print("F1 Score:", rf_f1_score)
```

Random Forest:  
Accuracy: 0.731535812232923  
Precision: 0.742003339080411  
Recall: 0.731535812232923  
F1 Score: 0.7258920435849853

Figure 33 :Result of prediction model evaluation metrics

### 3.9 Project Planning and Resource Allocation:

This project is considered a small project that does not contain the allocation of resources. The work was done directly on the Colab website as Colab is an information cloud. All resources are available online and not on a computer

## Chapter 4 : Results

## **Chapter 4: Results**

The application of the machine learning model was made using a random forest identifier in order to create an evaluation such as Accuracy - Precision - Recall - F1 - score. Therefore, after completing the implementation, you conclude that the model achieved 70% accuracy, and this means that the model can create a classification of data with a percentage 70% correct and 30% incorrect. Therefore, this value is not the best possible, and this is due to the data we have, as it is not very compatible with machine learning. Therefore, we went to create a visualization and display of the percentages, which in turn will be used to make predictions in the future and make decisions based on them.

### Results from the visualization process:

A data visualization was created to reach conclusions and present them more clearly so that they can be easily understood which in turn will have a major role in the decisions that will be made in the future.

- **Analyzing the regularity of attendance and departure based on academic qualification:**

Through the figure related to qualifications, we conclude that the bachelor's qualification was the highest at a rate of 49.11% compared to the doctorate, which was the lowest at a rate of 0.17% in terms of punctuality.

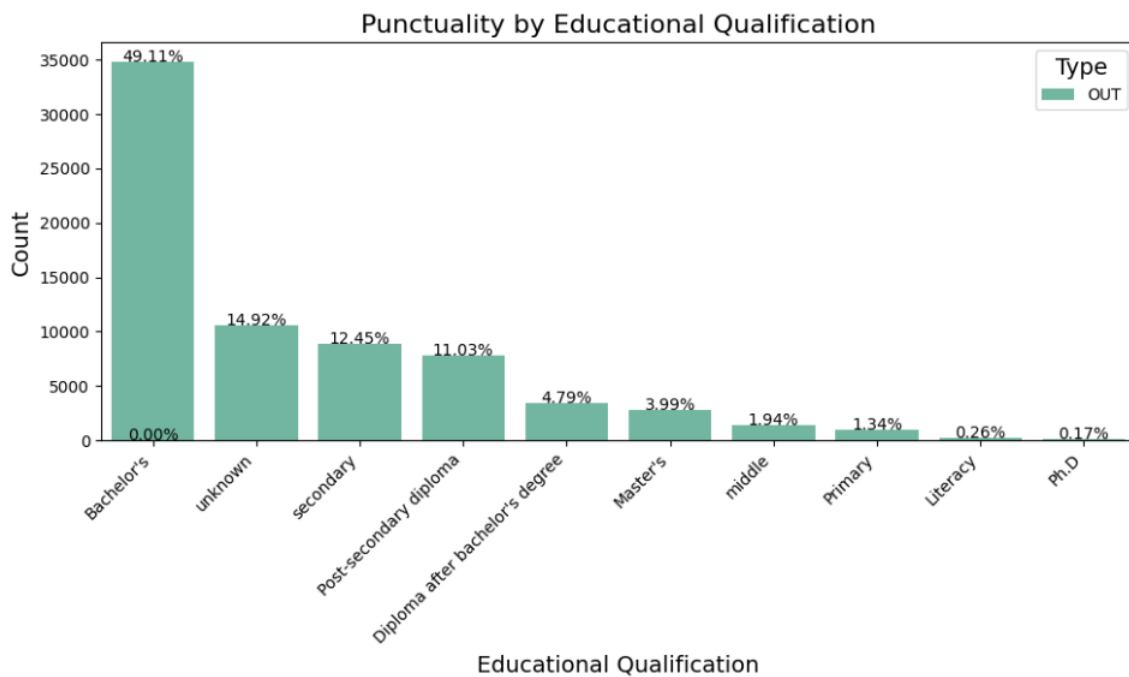


Figure 34: result of regularity of attendance and departure based on academic qualification

- **Comparison of attendance and departure patterns between males and females:**

We note that females were more committed to attendance compared to males, with a total of 52.79% of females and 47.21% of males.

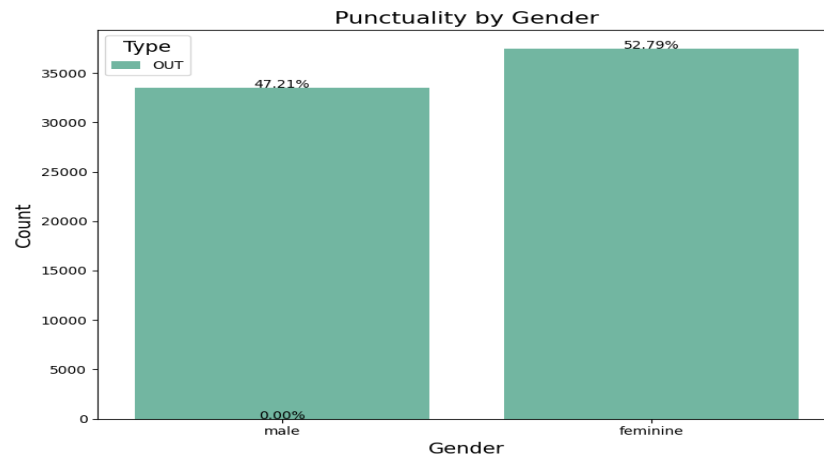


Figure 35: result attendance and departure patterns between males and females

- **Analyzing data to see the relationship between job rank and employees' commitment to working hours :**

Regarding the ranks, we note that the seventh, sixth, and eighth ranks were the highest in terms of commitment to attendance compared to the rest of the ranks, as the seventh rank, which was the highest, reached 29.66%, then the sixth, with a rate of 28.50%, then the eighth, with a rate of 20.51%, compared to the third rank, which is the lowest, with a rate of 0.29%

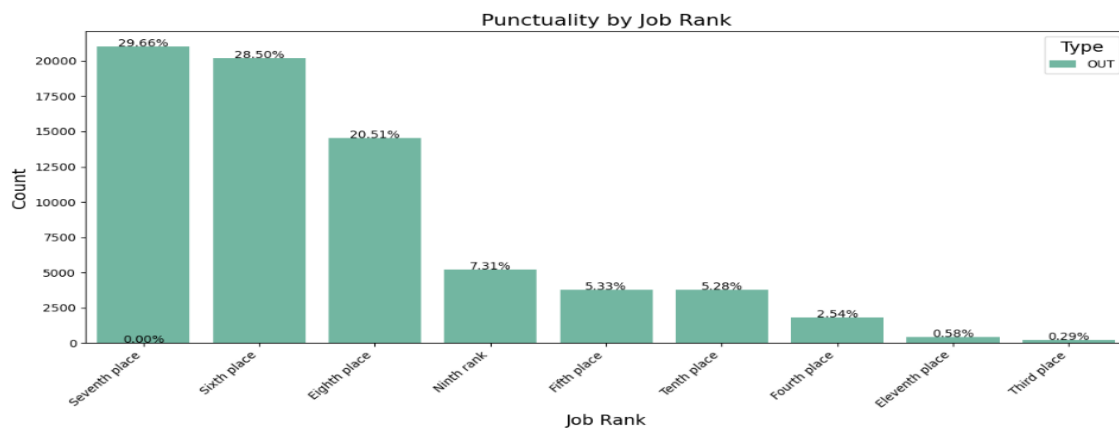


Figure 36: result job rank and employees' commitment to working hours

- Examining the extent to which the job affects discipline at work Filtering only

#### top 10 job titles :

Regarding jobs, we conclude that the job of assistant director was the highest in terms of discipline, at a rate of 23.42%, compared to the job of librarian, which was the lowest, at a rate of 1.56%.

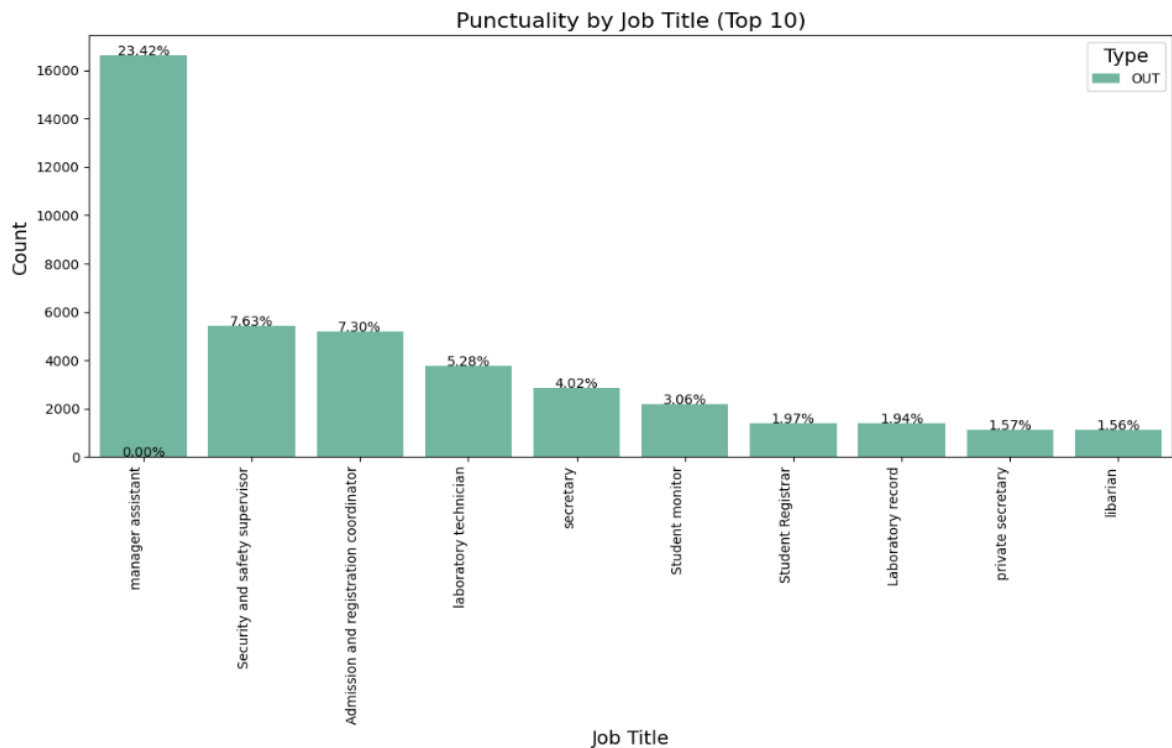


Figure 37:result extent to which the job affects discipline at work



- **Analyzing attendance and departure data across months to monitor patterns :**

In the middle of the year, attendance and departure rates were the lowest, reaching 4.22%, perhaps due to the mid-year vacation, compared to the last months of the year, which were the highest.

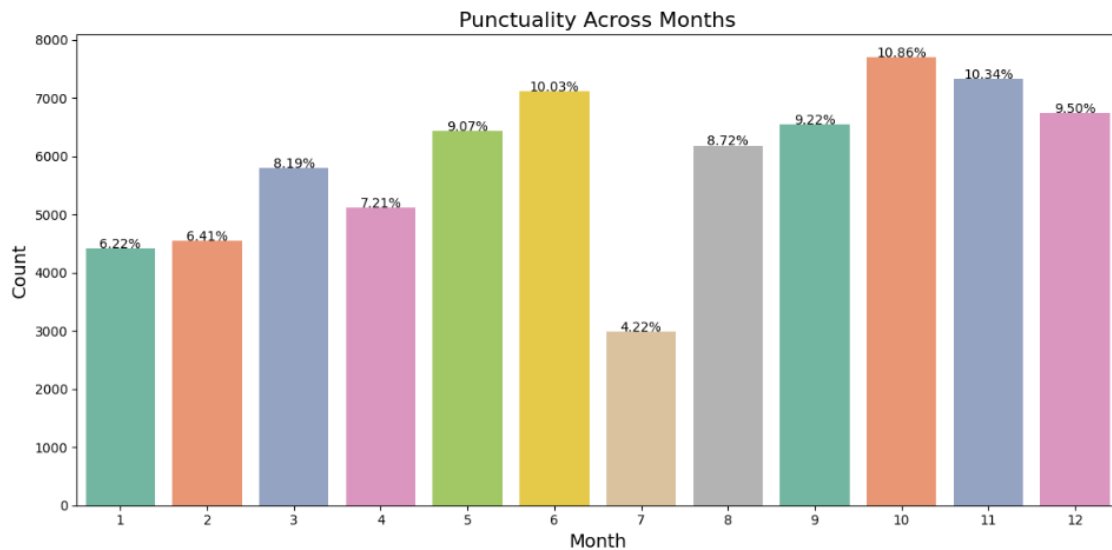


Figure 38:result attendance and departure data across months

- **Analyzing attendance and departure data across days of the week to monitor patterns :**

Attendance was higher during the beginning of the week, specifically on Monday and Tuesday, compared to the rest of the days of the week, such as Friday and Saturday, which were the lowest because they represent the weekend.

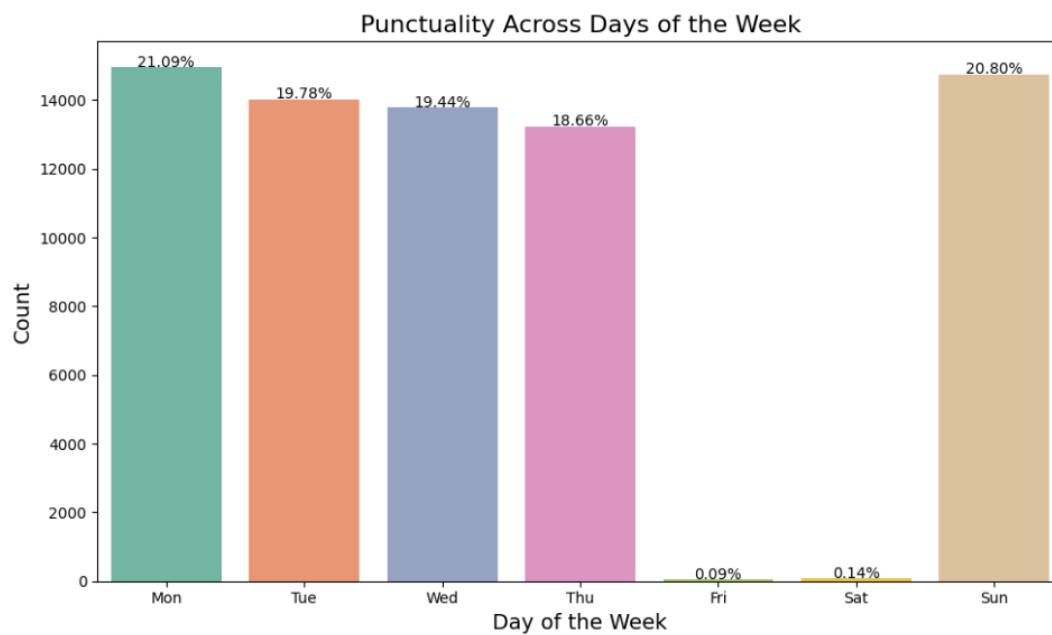


Figure 39:result attendance and departure data across days of the week

- Discover the rate of punctuality during the days of the week for all job ranks:

The seventh, sixth, and eighth ranks were the most committed throughout the week

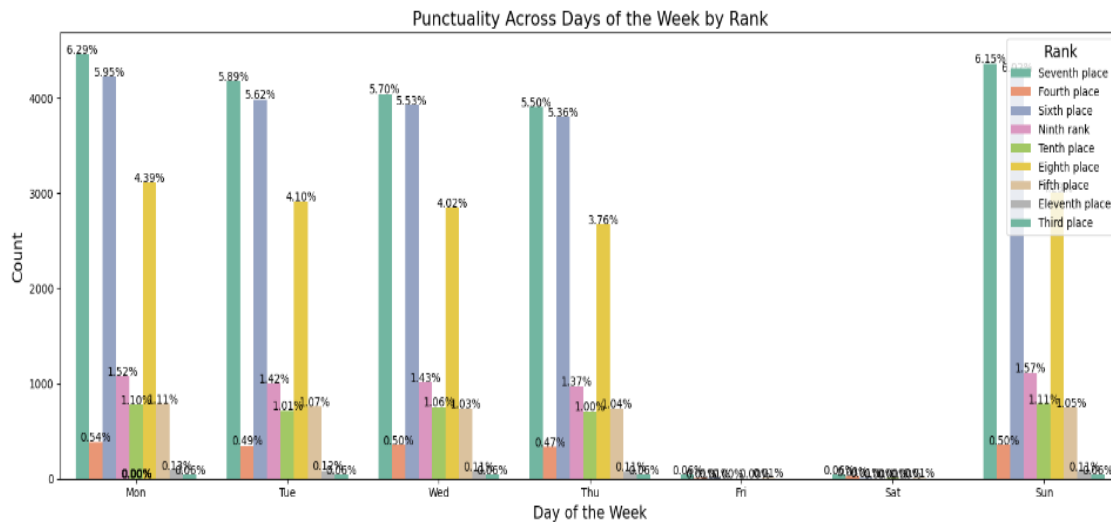


Figure 40:result punctuality during the days of the week for all job ranks

- Discover the rate of punctuality across months for both genders :

Females were the most punctual over the months compared to males who were the least

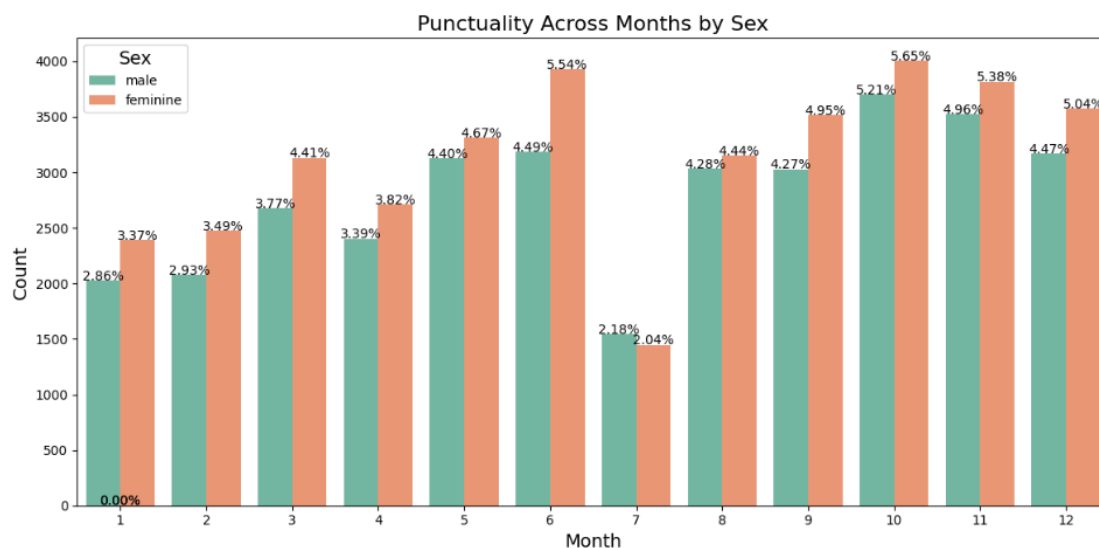


Figure 41:result punctuality across months for both genders

- **Show the trend on each month based on count of entries:**

In terms of discipline and attendance throughout the year, we note that the month of JUL was the lowest in terms of rates, with a rate of 4.22% compared to the month of October, which was the highest, with a rate of 10.86%.

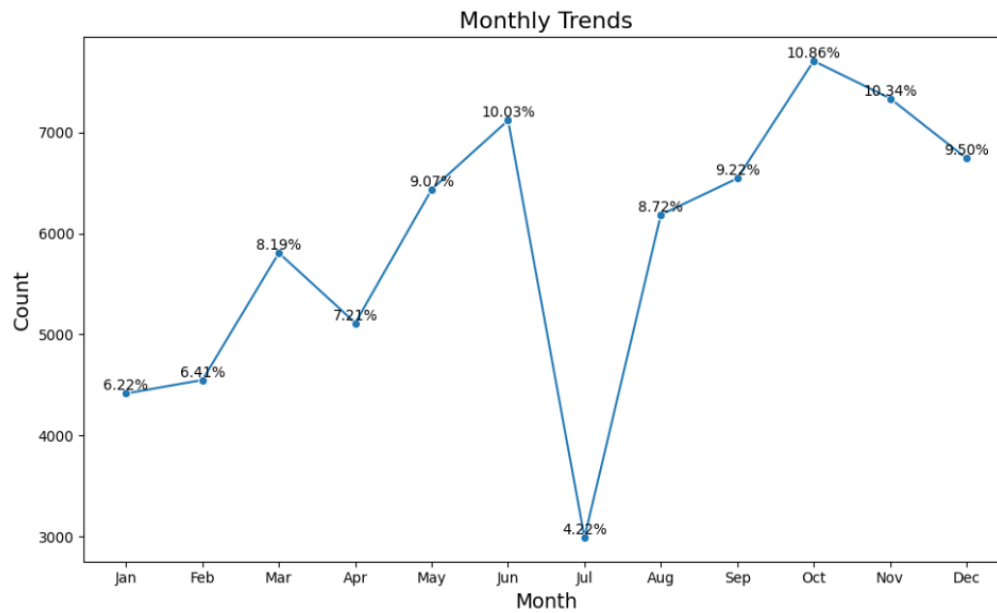


Figure 42:result trend on each month based on count of entries

## Using the dashboard:

An interactive control panel was created using the Tableau program to display the most prominent conclusions that were reached through the analysis and to clarify the relationships between our previous variables, such as gender and practical qualification rank, more clearly so that they can be understood easily and conveniently.



Figure 43 :Dashboard

## Chapter 5 : Conclusion

## **Chapter 5: Conclusion**

### **5.1 Conclusions :**

In conclusion, this project was trying to explore the regularity of employees' attendance and departure based on a set of data: gender, rank, qualification, and job title, which in turn is of great importance in helping institutions and bodies discover the most committed employees and the most non-committal ones, which will help later. To appoint the most committed employees. From a technical standpoint, this was done by analyzing and visualizing the data and developing a different set of visualizations while clarifying the ratios on each form to find the most committed people, in addition to applying machine learning, which is Random Forest, which succeeded in reaching the results of Accuracy: 0.73, Precision: 0.74, Recall: 0.73, F1 Score: 0.72. Therefore, this technology provided a lot of information on the subject that may meet the needs of organizations and bodies when making decisions in the future.

### **5.2 Recommendations :**

Based on the findings, the following recommendations are made for the university's management:

- Enhance Diversity Programs: Implement and expand diversity and inclusion training.
- Review Work Policies: Update work policies for flexibility and fairness.
- Expand Training Opportunities: Offer more professional development to support career growth.
- Regular Evaluations: Conduct periodic evaluations to assess policy effectiveness and adapt as needed.

## References

- Bass, B. M., & Avolio, B. J. (1994). Improving organizational effectiveness through transformational leadership. Thousand Oaks, CA: Sage.
- García-Morales, V. J., Jiménez-Barrionuevo, M. M., & Gutiérrez-Gutiérrez, L. (2012). Transformational leadership influence on organizational performance through organizational learning and innovation. *Journal of Business Research*, 65(7), 1040-1050.
- Herzberg, F., Mausner, B., & Snyderman, B. B. (1959). *The motivation to work* (2nd ed.). New York, NY: John Wiley & Sons.
- Johnson, S., & Smith, P. (2013). Gender differences in workplace behavior and conflict resolution. *Journal of Business Studies Quarterly*, 5(1), 12-22.
- Kotter, J. P., & Heskett, J. L. (1992). *Corporate culture and performance*. New York: Free Press.
- Leith wood, K., & Jantzi, D. (2000). The effects of transformational leadership on organizational conditions and student engagement with school. *Journal of Educational Administration*, 38(2), 112-129.
- Maslow, A. H. (1943). A theory of human motivation. *Psychological Review*, 50(4), 370-396.
- Nguyen, T., Taylor, J., & Bradley, S. (2016). Job autonomy and job satisfaction: new evidence. *Education Economics*, 24(3), 282-307.
- Smith, M., & Robertson, J. (2015). Demographic influences on employee satisfaction in higher education. *Teaching in Higher Education*, 20(5), 513-527.
- Turner, B. A., & Müller, R. (2005). The influence of project managers on



## Appendices A

المؤهل	المسمى الوظيفي	الدرجة	المرتبة	التشكيل	الجنس	ID	م
ثانوي	مراقب امن وسلامة	الدرجة 09	المرتبة السابعة	سلم الموظفين العام	ذكر	٤٠١٠٠٦٣٥	1
ثانوي	مراقب امن وسلامة	الدرجة 08	المرتبة الخامسة	سلم الموظفين العام	انثى	٤٠١٠٠٤٢٩	2
دبلوم بعد البكالوريوس	مساعد اداري	الدرجة 11	المرتبة الثامنة	سلم الموظفين العام	انثى	٤٠١٠٠٥١٣	3
بكالوريوس	مراقب امن وسلامة	الدرجة 08	المرتبة السابعة	سلم الموظفين العام	انثى	٤٠١٠٠٦٨٣	4
بكالوريوس	أخصائي علاقات عامة متقدم	الدرجة 15	المرتبة العاشرة	سلم الموظفين العام	ذكر	٤٠١٠٠٨٤٠	5
بكالوريوس	أمين صندوق	الدرجة 11	المرتبة الثامنة	سلم الموظفين العام	انثى	٤٠١٠٠٢٢٣	6
بكالوريوس	مساعد اداري	الدرجة 11	المرتبة الثامنة	سلم الموظفين العام	انثى	٤٠١٠٠٢٢٥	7
بكالوريوس	مساعد اداري	الدرجة 11	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٢٧٨	8
بكالوريوس	مساعد اداري	الدرجة 05	المرتبة الثامنة	سلم الموظفين العام	انثى	٤٠١٠٠٣٦١	9
بكالوريوس	فني مختبر	الدرجة 08	المرتبة السابعة	سلم الموظفين العام	انثى	٤٠١٠٠٣٥٦	10
بكالوريوس	مشغل اجهزة مكتبية	الدرجة 11	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٣٨٩	11
دبلوم بعد الثانوي	مطور برامج	الدرجة 12	المرتبة العاشرة	سلم الموظفين العام	ذكر	٤٠١٠٠٦١٣	12
	مساعد اداري	الدرجة 10	المرتبة السادسة	سلم الموظفين العام	ذكر	٤٠١٠٠٦٣٦	13
متوسط	مساعد اداري ممارس ثاني	الدرجة 07	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٥٣٤	14
ثانوي	مساعد اداري	الدرجة 05	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٥٢٣	15
	مراقب طلبة	الدرجة 11	المرتبة السادسة	سلم الموظفين العام	انثى	٤٠١٠٠٣٥٨	16

ID	DateTime	Type
٤٠١٠٠٦٣٥	٣١/١٢/٢٠٢٢ ١٩:٠٢:٢٠	OUT
٤٠١٠٠٦٣٨	٣١/١٢/٢٠٢٢ ١٨:٥٩:٠١	OUT
٤٠٨٠٠٣٤٨	٣١/١٢/٢٠٢٢ ١٧:٣٨:١٧	OUT
٤٠١٠٠٥٥٥	٣١/١٢/٢٠٢٢ ١٦:٢٩:٢٥	OUT
٤٠١٠٠٦٥٩	٣١/١٢/٢٠٢٢ ١٤:٠١:٣٥	OUT
٤٠١٠٠٦٣٥	٣١/١٢/٢٠٢٢ ١٣:٠٣:٠٢	OUT
٤٠١٠٠٤٦٦	٣١/١٢/٢٠٢٢ ١٣:٠١:١٥	OUT
٤٠١٠٠٥٥٥	٣١/١٢/٢٠٢٢ ١١:٣٧:٠٩	OUT
٤٠١٠٠٦٣٥	٣٠/١٢/٢٠٢٢ ١٨:٥٠:٢٣	OUT
٤٠١٠٠٦٤٤	٣٠/١٢/٢٠٢٢ ١٨:٣٩:٤٩	OUT
٤٠١٠٠٦٣٥	٣٠/١٢/٢٠٢٢ ١٣:٠٩:٠١	OUT

Class	Rank	Formation	Sex	ID	M
Class 09	Seventh place	General staff ladder	male	40100635	1
Class 08	Fifth place	General staff ladder	feminine	40100429	2
Class 11	Eighth place	General staff ladder	feminine	40100513	3
Class 08	Seventh place	General staff ladder	feminine	40100383	4
Class 15	Tenth place	General staff ladder	male	40100840	5
Class 11	Eighth place	General staff ladder	feminine	40100323	6
Class 11	Eighth place	General staff ladder	feminine	40100325	7
Class 11	Sixth place	General staff ladder	feminine	40100378	8

1	ID	DateTime	Type
2	40100635	12/31/2022 19:02:20	OUT
3	40100638	12/31/2022 18:59:01	OUT
4	40800348	12/31/2022 17:38:17	OUT
5	40100555	12/31/2022 16:29:25	OUT
6	40100659	12/31/2022 14:01:25	OUT

## Appendices B

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
file1 = pd.read_excel('مصنات الخروج فقط للموظفين.xlsx')
file1.head()
```

```
file2 = pd.read_excel('متابعة الانصراف (1).xlsx')
file2.head()
```

```
file2.columns = file2.columns.str.strip()
file1.columns = file1.columns.str.strip()
```

```
data = pd.merge(file1, file2, on = 'ID')
data.head()
```

```
data.shape
```

```
data.info()
```

```
data.describe()
```

```
data.describe(include = 'object')
```

```
data.isna().sum()
```

```
data['qualification'].unique()
```

```
data = data.applymap(lambda x: x.strip() if isinstance(x, str) else x)
```

```
data['qualification'].unique()
```

```
data['qualification'] = data['qualification'].fillna('unknown')
data['qualification'].unique()
```

```
data.isna().sum()
```

```
data.duplicated().sum()
```

```
print('Data size with duplicaion: ', data.shape[0])
data = data.drop_duplicates()
print('Data size without duplicaion: ', data.shape[0])
```

```
cols = ['Sex', 'qualification', 'Class', 'Rank']
```

```
for col in cols:
    print(data[col].value_counts())
    print('*'*100)
```

```

# Calculate the count of each category in the 'qualification' column
qualification_counts = data['qualification'].value_counts()

# Get the categories sorted by count
qualification_order = qualification_counts.index.tolist()

plt.figure(figsize=(10, 6))
ax = sns.countplot(x='qualification', hue='Type', data=data, palette='Set2', order=qualification_order)

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality by Educational Qualification', fontsize=16)
plt.xlabel('Educational Qualification', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(rotation=45, ha='right')
plt.legend(title='Type', title_fontsize='14')
plt.tight_layout()
plt.show()

```

```

plt.figure(figsize=(8, 6))
ax = sns.countplot(x='Sex', hue='Type', data=data, palette='Set2')

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality by Gender', fontsize=16)
plt.xlabel('Gender', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.legend(title='Type', title_fontsize='14')
plt.tight_layout()
plt.show()

```

```

# Calculate the count of each category in the 'Rank' column
rank_counts = data['Rank'].value_counts()

# Get the categories sorted by count
rank_order = rank_counts.index.tolist()

plt.figure(figsize=(12, 6))
ax = sns.countplot(x='Rank', hue='Type', data=data, palette='Set2', order=rank_order)

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality by Job Rank', fontsize=16)
plt.xlabel('Job Rank', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(rotation=45, ha='right')
plt.legend(title='Type', title_fontsize='14')
plt.tight_layout()
plt.show()

```

```

# Calculate the count of each category in the 'Job title' column
job_title_counts = data['Job title'].value_counts()

# Get the categories sorted by count
job_title_order = job_title_counts.index.tolist()

# Filter only the top 10 job titles
top_10_job_titles = job_title_order[:10]

plt.figure(figsize=(12, 8))
ax = sns.countplot(x='Job title', hue='Type', data=data, palette='Set2', order=top_10_job_titles)

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality by Job Title (Top 10)', fontsize=16)
plt.xlabel('Job Title', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(rotation=90, ha='right')
plt.legend(title='Type', title_fontsize='14')
plt.tight_layout()
plt.show()

```

```

data['Month'] = data['DateTime'].dt.month

plt.figure(figsize=(12, 6))
ax = sns.countplot(x='Month', data=data, palette='Set2', hue='Month', legend=False)

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality Across Months', fontsize=16)
plt.xlabel('Month', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.tight_layout()
plt.show()

```

```

data['DayOfWeek'] = data['DateTime'].dt.dayofweek

plt.figure(figsize=(10, 6))
ax = sns.countplot(x='DayOfWeek', data=data, palette='Set2', hue='DayOfWeek', legend=False)

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality Across Days of the Week', fontsize=16)
plt.xlabel('Day of the Week', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(ticks=[0, 1, 2, 3, 4, 5, 6], labels=['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun'])
plt.tight_layout()
plt.show()

```

```

plt.figure(figsize=(16, 6))
ax = sns.countplot(x='DayOfWeek', hue='Rank', data=data, palette='Set2')

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality Across Days of the Week by Rank', fontsize=16)
plt.xlabel('Day of the Week', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(ticks=[0, 1, 2, 3, 4, 5, 6], labels=['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun'])
plt.tight_layout()
plt.legend(title='Rank', title_fontsize='14')
plt.show()

```

```

plt.figure(figsize=(12, 6))
ax = sns.countplot(x='Month', hue='Sex', data=data, palette='Set2')

total = float(len(data)) # Total number of observations

for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width() / 2.,
            height + 3,
            '{:.2f}%'.format((height / total) * 100),
            ha="center")

plt.title('Punctuality Across Months by Sex', fontsize=16)
plt.xlabel('Month', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.tight_layout()
plt.legend(title='Sex', title_fontsize='14')
plt.show()

```

```

plt.figure(figsize=(10, 6))
data['DateTime'] = pd.to_datetime(data['DateTime'])
monthly_counts = data['Month'].value_counts().sort_index()
total_counts = monthly_counts.sum() # Total count of all data points

# Calculate percentages
percentages = (monthly_counts / total_counts) * 100

sns.lineplot(x=monthly_counts.index, y=monthly_counts.values, marker='o')

# Annotate each point with its percentage
for i, count in enumerate(monthly_counts.values):
    plt.text(monthly_counts.index[i], count, f"{percentages.iloc[i]:.2f}%", ha='center', va='bottom')

plt.title('Monthly Trends', fontsize=16)
plt.xlabel('Month', fontsize=14)
plt.ylabel('Count', fontsize=14)
plt.xticks(ticks=range(1, 13), labels=['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'])
plt.tight_layout()
plt.show()

```

```

from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, accuracy_score
from sklearn.preprocessing import LabelEncoder
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.neighbors import KNeighborsClassifier

```

```
data['qualification'].value_counts()
```

```

from sklearn.utils import resample

max_samples_per_class = data['qualification'].value_counts().max()

oversampled_data = pd.DataFrame(columns=data.columns)
for qualification, group in data.groupby('qualification'):
    if len(group) < max_samples_per_class:
        oversampled_group = resample(group, replace=True, n_samples=max_samples_per_class, random_state=42)
        oversampled_data = pd.concat([oversampled_data, oversampled_group])

print(oversampled_data.shape)

```

```
oversampled_data['qualification'].value_counts()
```

```

label_encoder = LabelEncoder()
oversampled_data['Sex'] = label_encoder.fit_transform(oversampled_data['Sex'])
oversampled_data['Formation'] = label_encoder.fit_transform(oversampled_data['Formation'])
oversampled_data['Job title'] = label_encoder.fit_transform(oversampled_data['Job title'])
# oversampled_data['qualification'] = label_encoder.fit_transform(oversampled_data['qualification'])

```

```

X = oversampled_data[['Sex', 'Formation', 'Job title', 'Month', 'DayOfWeek']]
y = oversampled_data['qualification']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

```

```

# Training
rf_classifier = RandomForestClassifier()
rf_classifier.fit(X_train, y_train)

# Predictions
rf_predictions = rf_classifier.predict(X_test)

# Evaluation
rf_accuracy = accuracy_score(y_test, rf_predictions)
rf_precision = precision_score(y_test, rf_predictions, average='weighted')
rf_recall = recall_score(y_test, rf_predictions, average='weighted')
rf_f1_score = f1_score(y_test, rf_predictions, average='weighted')

# Results
print("Random Forest:")
print("Accuracy:", rf_accuracy)
print("Precision:", rf_precision)
print("Recall:", rf_recall)
print("F1 Score:", rf_f1_score)

```

## Appendices C

-----  
Dr. Maryam M. Al Dabel  
Acting Vice Dean of Computer Science and Engineering College  
College of Computer Science & Engineering  
University of Hafr Al Batin  
Hafr Al-Batin, Saudi Arabia

**From:** Inas Mohammed Abdulraziq <inasm@uhb.edu.sa>  
**Sent:** Thursday, February 1, 2024 12:33 AM  
**To:** Dr. Maryam M. Al Dabel <maldabel@uhb.edu.sa>  
**Subject:** طلب بيانات موظفين

السلام عليكم ورحمة الله وبركاته

تحية طيبة وبعد...

في خضم متابعة الطالبات في مشاريع التخرج لقسم علوم البيانات وحيث ان احد المشاريع يتمحور حول دراسة وتحليل بيانات دوام الموظفين الاداريين في جامعة حفر الباطن ومحاولة تنبأ نمط معين حول سلوك الموظفين فأرجو من سعادتك تسهيل طلب بيانات (dataset) من الادارة العامة للموارد البشرية والتي تحوي على المتغيرات التالية:

1. جنس الموظف.

2. سنوات الخبرة.

3. المؤهل الوظيفي.

4. السلم الوظيفي.

5. مكان السكن.

6. ايام الاجازات.

7. ساعات المغادرة من الدوام.

علما بان المشروع يفترض ان يستند لبيانات ضمن فترة زمنية طويلة (اكثر من سنتين) وايضا لا يهتم بوجود اسماء الموظفين في البيانات للحفاظ على خصوصيتهم.

ولكم جزيل الشكر

د. رائد بن سفر الحارثي

المشرف العام  
على الإدارة العامة للموارد البشرية  
ralharthi@uhb.edu.sa - 013 720 5228

الملكة العربية السعودية ، تليفون: 0137203462  
فاكس: 0137247212 ص.ب. 1803 - حفر الباطن 31991  
Tel: 0137203462 Fax: 0137247212 P.O. Box: 1803 - Hafr Albatin 31991  
www.uhb.edu.sa



جامعة حفر الباطن  
University of Hafr Al Batin

**From:** Dr. Maryam M. Al Dabel <maldabel@uhb.edu.sa>

**Sent:** Thursday, February 1, 2024 7:57 AM

**To:** Dr. Raed Alharthi <ralharthi@uhb.edu.sa>

**Cc:** عبدالرحمن الزهراني <aalzahrani@uhb.edu.sa>; Dr. Ibrahim Al Zahrani <ialzahrani@uhb.edu.sa>

**Subject:** FW: طلب بيانات موظفين لمشروع تخرج

السلام عليكم ورحمة الله وبركاته

أسعد الله صباحكم بكل خير د. رائد

في الإيميل ادناه طلب توفير بيانات لاحدى مشاريع التخرج في برنامج علوم البيانات يخص الموظفين الإداريين في جامعة حفر الباطن

فهل يمكن توفير البيانات ادناه وماهو الإجراء المناسب في حال إمكانية ذلك

تحياتي وتقديري



د. رائد بن سفر الحارثي

المشرف العام  
على الإدارة العامة للموارد البشرية  
ralharthi@uhb.edu.sa - 013 720 5228



جامعة حفر الباطن  
University of Hafr Al Batin

المملكة العربية السعودية - تلغراف: 0137203462  
فاكس: 0137247212 ص ب 1803 حفر الباطن 31991  
Tel: 0137203462 Fax: 0137247212 P.O.Box: 1803 - Hafr Al Batin 31991  
www.uhb.edu.sa

**From:** Dr. Maryam M. Al Dabel <maldabel@uhb.edu.sa>  
**Sent:** Monday, February 5, 2024 8:29 AM  
**To:** Dr. Raed Alharthi <ralharthi@uhb.edu.sa>  
**Cc:** عبدالرحمن الزهراني <aalzahrani@uhb.edu.sa>; Dr. Ibrahim Al Zahrani <ialzahrani@uhb.edu.sa>  
**Subject:** Re: طلب بيانات موظفين لمشروع تخرج

السلام عليكم ورحمة الله وبركاته

أسعد الله صباحكم بكل خير دكتور رائد

المتغيرات المذكورة كافية بإذن الله حسب إفادة المشرف على المشروع ونرغب بالحصول عليها

شاكرين حسن تعاونكم معنا

تحياتي

-----  
Dr. Maryam M. Al Dabel  
Acting Vice Dean of Computer Science and Engineering College  
College of Computer Science & Engineering  
University of Hafr Al Batin  
Hafr Al-Batin, Saudi Arabia

**From:** Dr. Raed Alharthi <ralharthi@uhb.edu.sa>  
**Sent:** Monday, February 5, 2024 12:01 AM  
**To:** Dr. Maryam M. Al Dabel <maldabel@uhb.edu.sa>  
**Cc:** عبدالرحمن الزهراني <aalzahrani@uhb.edu.sa>; Dr. Ibrahim Al Zahrani <ialzahrani@uhb.edu.sa>  
**Subject:** Re: طلب بيانات موظفين لمشروع تخرج

وعليكم السلام ورحمة الله وبركاته

نفيدكم بعدم وجود قاعدة بيانات تحتوي على جميع المتغيرات المطلوبة، وقد يتوفر الآتي:

1. جنس الموظف.
2. سنوات الخبرة.
3. المؤهل الوظيفي (جزئي)
4. السلم الوظيفي.
5. ساعات المغادرة من الدوام

إذا كانت كافية فقد نحتاج 5 أيام عمل لتوفيرها وتنقيحها.

د. رائد بن سفر الحارثي  
المشرف العام  
على الإدارة العامة للموارد البشرية  
ralharthi@uhb.edu.sa - 013 720 5228

الجامعة العربية السعودية - الرياض، ص.ب. 1803 - ج.ع.ب. 31991  
هاتف: 0137203462 فاكس: 0137247212 P.O. Box: 1803 - Hafr Al-Batin 31991  
www.uhb.edu.sa



جامعة حفر الباطن  
University of Hafr Al Batin

**From:** Dr. Maryam M. Al Dabel <maldabel@uhb.edu.sa>  
**Sent:** Monday, February 19, 2024 10:41 AM  
**To:** Dr. Raed Alharthi <ralharthi@uhb.edu.sa>  
**Cc:** عبدالرحمن الزهراني <aalzahrani@uhb.edu.sa>; Dr. Ibrahim Al Zahrani <ialzahrani@uhb.edu.sa>  
**Subject:** Re: طلب بيانات موظفين لمشروع تخرج

وعليكم السلام ورحمة الله وبركاته

نشكر لكم تعاونكم دكتور رائد  
كما نأمل احاطتنا برد الشركة المشغلة ان امكن  
الطالبات سيستخدمن قاعدة بيانات مختلفة حاليا مع الحفاظ على أهداف المشروع

تحياتي

-----  
Dr. Maryam M. Al Dabel  
Acting Vice Dean of Computer Science and Engineering College  
College of Computer Science & Engineering  
University of Hafr Al Batin  
Hafr Al-Batin, Saudi Arabia

**From:** Dr. Raed Alharthi <ralharthi@uhb.edu.sa>  
**Sent:** Sunday, February 18, 2024 10:13 PM  
**To:** Dr. Maryam M. Al Dabel <maldabel@uhb.edu.sa>  
**Cc:** عبدالرحمن الزهراني <aalzahrani@uhb.edu.sa>; Dr. Ibrahim Al Zahrani <ialzahrani@uhb.edu.sa>  
**Subject:** Re: طلب بيانات موظفين لمشروع تخرج

سعادة الدكتورة مريم

السلام عليكم ورحمة الله وبركاته

نفيدكم بعدم تمكننا للحصول على بيانات أوقات مغادرة الموظفين في قاعدة بيانات مفتوحة تسمح لنا بحذف بيانات الموظفين  
السرية، وسيتم الرفع للشركة المشغلة للتحقق من إمكانية الحصول على ملفات إكسل من خلال قاعدة البيانات لنتمكن من  
تزويدكم بالمطلوب.

مع الشكر والتقدير

**From:** Dr. Maryam M. Al Dabel <maldabel@uhb.edu.sa>  
**Sent:** Tuesday, February 20, 2024 8:11:07 AM  
**To:** Inas Mohammed Abdulraziq <inasm@uhb.edu.sa>  
**Subject:** FW: طلب بيانات موظفين لمشروع تخرج

السلام عليكم ورحمة الله وبركاته

صباح الخير ا. ايناس

مرفق البيانات المطلوبة

مع تمنياتنا بالتوفيق

-----  
Dr. Maryam M. Al Dabel  
Acting Vice Dean of Computer Science and Engineering College  
College of Computer Science & Engineering  
University of Hafr Al Batin  
Hafar Al-Batin, Saudi Arabia

---

**From:** Dr. Raed Alharthi <ralharthi@uhb.edu.sa>  
**Sent:** Tuesday, February 20, 2024 8:03 AM  
**To:** Dr. Maryam M. Al Dabel <maldabel@uhb.edu.sa>  
**Cc:** عبدالرحمن الزهراني <aalzahrani@uhb.edu.sa>; Dr. Ibrahim Al Zahrani <ialzahrani@uhb.edu.sa>  
**Subject:** Re: طلب بيانات موظفين لمشروع تخرج

سعادة الدكتورة مريم

السلام عليكم ورحمة الله وبركاته

تجدون برفقه البيانات المطلوبة، مع الإحاطة بأن رقم (ID) وهمي ليتمكن دمج البيانات المطلوبة.

مع تمنياتنا لكم بالتوفيق

---

## Appendices D

	ID	DateTime	Type
0	40100635	2022-12-31 19:02:20	OUT
1	40100638	2022-12-31 18:59:01	OUT
2	40800348	2022-12-31 17:38:17	OUT
3	40100656	2022-12-31 16:29:25	OUT
4	40100659	2022-12-31 14:01:25	OUT

	M	ID	Sex	Formation	Rank	Class	Job title	qualification
0	1	40100635	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary
1	2	40100429	feminine	General staff ladder	Fifth place	Class 08	Security and safety supervisor	secondary
2	3	40100513	feminine	General staff ladder	Eighth place	Class 11	manager assistant	Diploma after bachelor's degree
3	4	40100383	feminine	General staff ladder	Seventh place	Class 08	Security and safety supervisor	Bachelor's
4	5	40100840	male	General staff ladder	Tenth place	Class 15	Advanced public relations specialist	Bachelor's

	ID	DateTime	Type	M	Sex	Formation	Rank	Class	Job title	qualification
0	40100635	2022-12-31 19:02:20	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary
1	40100635	2022-12-31 13:03:02	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary
2	40100635	2022-12-30 18:50:23	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary
3	40100635	2022-12-30 13:09:01	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary
4	40100635	2022-12-29 19:05:26	OUT	1	male	General staff ladder	Seventh place	Class 09	Security and safety supervisor	secondary

(73035, 10)

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 73035 entries, 0 to 73034
Data columns (total 10 columns):
#   column              Non-Null count  Dtype
---  --
0   ID                  73035 non-null  int64
1   DateTime            73035 non-null  datetime64[ns]
2   Type                73035 non-null  object
3   M                   73035 non-null  object
4   Sex                 73035 non-null  object
5   Formation           73035 non-null  object
6   Rank                73035 non-null  object
7   Class               73035 non-null  object
8   Job title           73035 non-null  object
9   qualification       62133 non-null  object
dtypes: datetime64[ns](1), int64(1), object(8)
memory usage: 5.6+ MB
```

	ID	DateTime
count	7.303500e+04	73035
mean	4.010057e+07	2022-07-19 20:28:29.293927680
min	4.010032e+07	2022-01-02 11:31:58
25%	4.010043e+07	2022-04-17 15:50:29
50%	4.010057e+07	2022-08-01 14:15:12
75%	4.010070e+07	2022-10-24 14:15:33.500000
max	4.010085e+07	2022-12-31 19:02:20
std	1.501584e+02	NaN

	Type	M	Sex	Formation	Rank	Class	Job title	qualification
count	73035	73035	73035	73035	73035	73035	73035	62133
unique	1	389	2	1	9	14	96	9
top	OUT	9	feminine	General staff ladder	Seventh place	Class 07	manager assistant	Bachelor's
freq	73035	476	38562	73035	21581	12933	17022	35741

```
ID          0
DateTime    0
Type        0
M           0
Sex         0
Formation   0
Rank        0
Class       0
Job title   0
qualification 10902
dtype: int64
```

```
array(['secondary', nan, "Bachelor's", 'middle', 'Primary',
      'Post-secondary diploma', "Master's",
      "Diploma after bachelor's degree", 'Ph.D', 'Literacy'],
      dtype=object)
```

```
array(['secondary', nan, "Bachelor's", 'middle', 'Primary',
      'Post-secondary diploma', "Master's",
      "Diploma after bachelor's degree", 'Ph.D', 'Literacy'],
      dtype=object)
```

```
ID          0
DateTime    0
Type        0
M           0
Sex         0
Formation   0
Rank        0
Class       0
Job title   0
qualification 0
dtype: int64
```

```
Data size with duplicaion: 73035
Data size without duplicaion: 70941
```

```

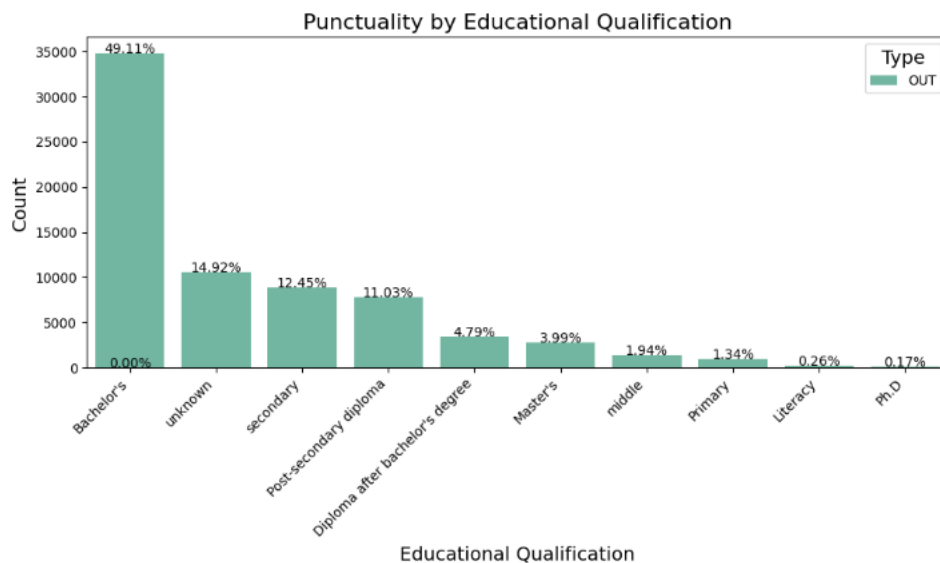
Sex
feminine    37451
male        33490
Name: count, dtype: int64
*****

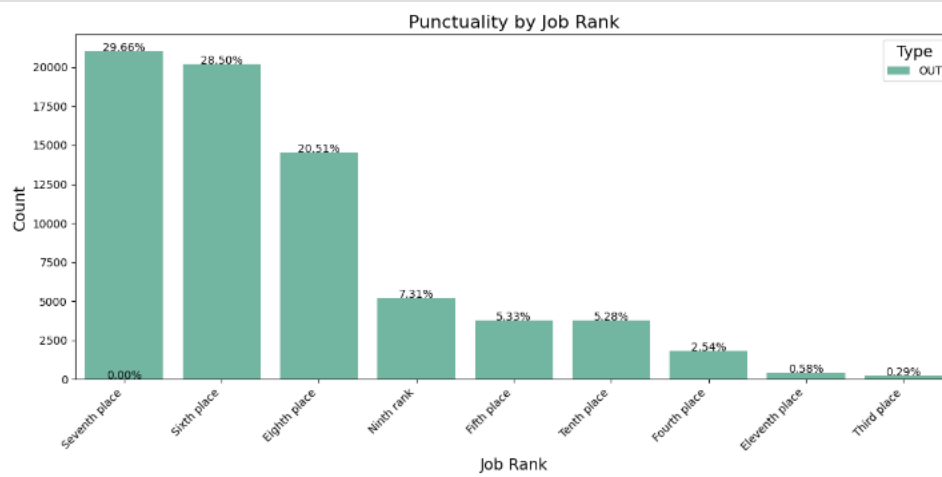
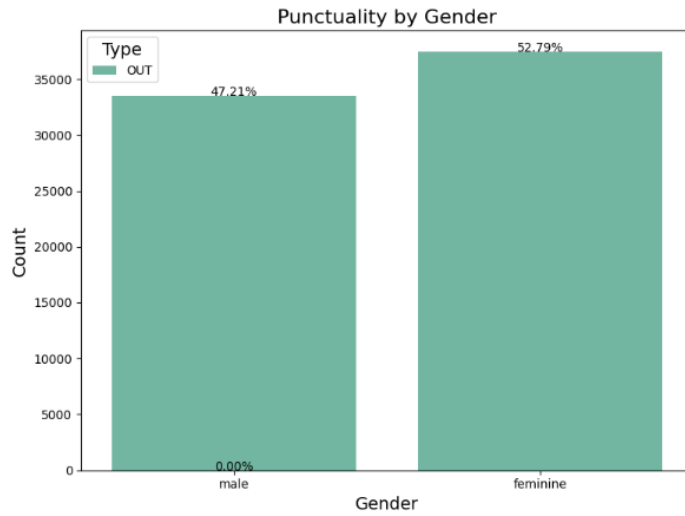
qualification
Bachelor's          34842
unknown            10585
secondary           8835
Post-secondary diploma  7827
Diploma after bachelor's degree  3395
Master's           2830
middle             1376
Primary            948
Literacy           183
Ph.D               120
Name: count, dtype: int64
*****

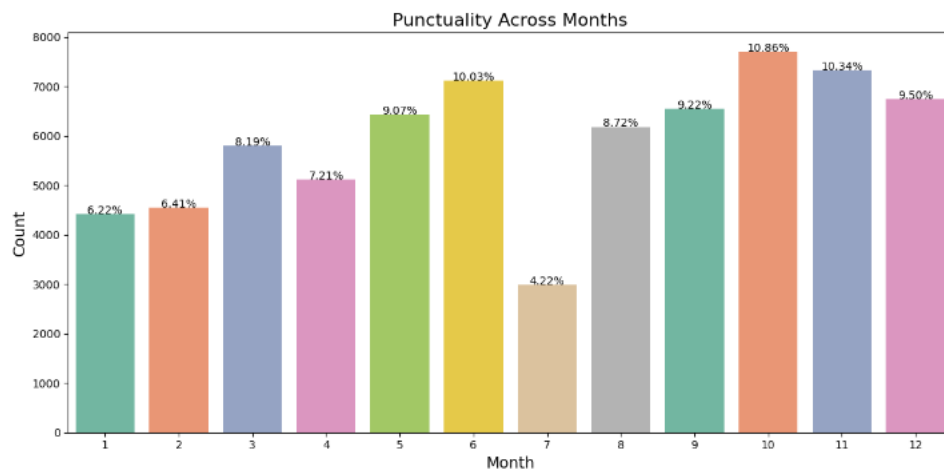
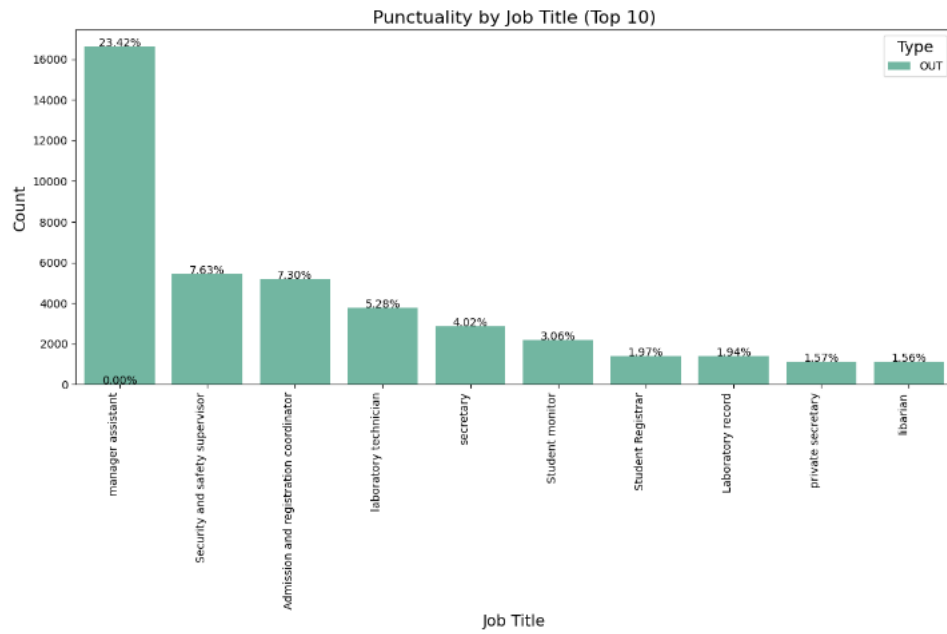
Class
Class 07    12494
Class 08    10929
Class 09     9706
Class 11     7724
Class 04     6977
Class 06     6909
Class 05     5670
Class 10     2450
Class 12     1879
Class 03     1787
Class 15     1524
Class 13     1181
Class 02     1167
Class 14      544
Name: count, dtype: int64
*****

Rank
Seventh place    21038
Sixth place      20219
Eighth place     14552
Ninth rank       5186
Fifth place      3778
Tenth place      3746
Fourth place     1803
Eleventh place   410
Third place      209
Name: count, dtype: int64
*****

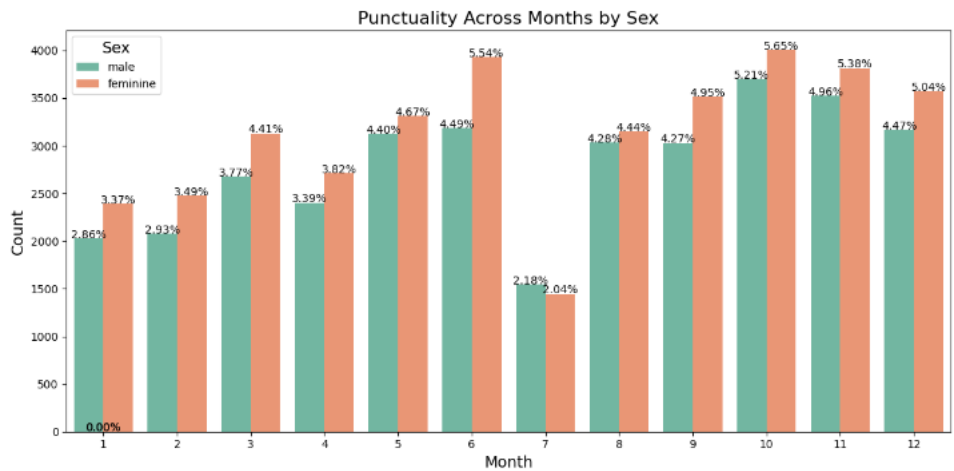
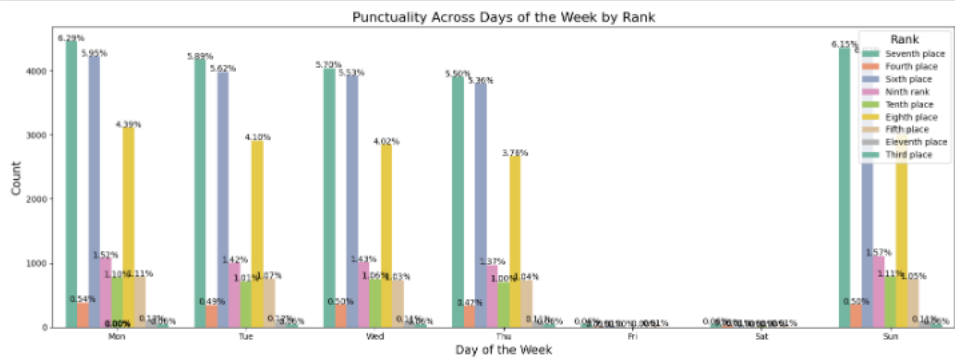
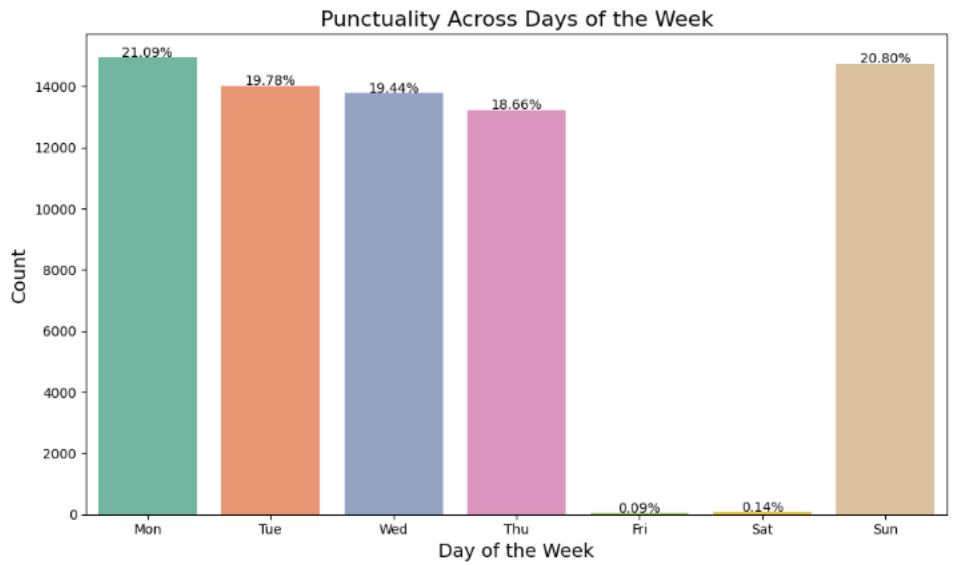
```

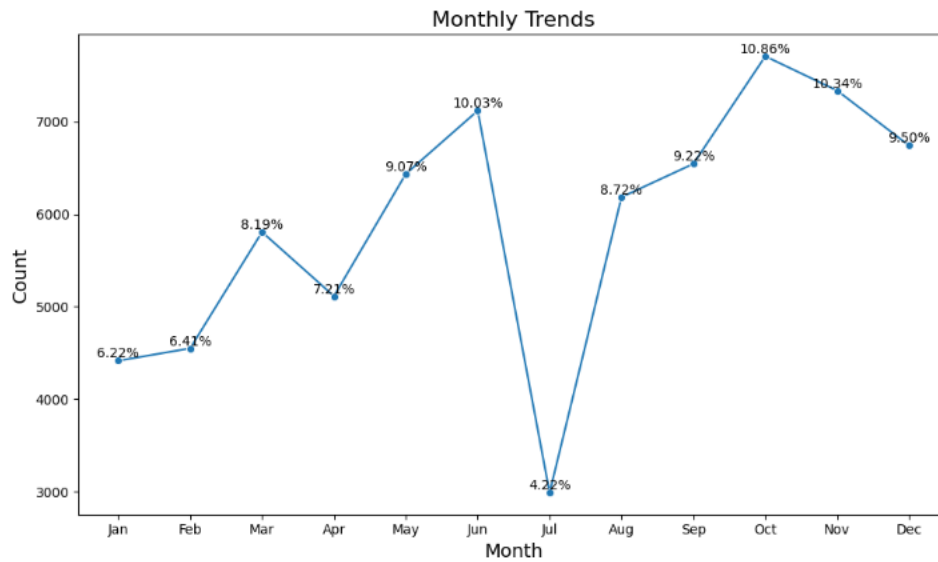












```

qualification
Bachelor's          34842
unknown             10585
secondary            8835
Post-secondary diploma  7827
Diploma after bachelor's degree  3395
Master's            2830
middle              1376
Primary              948
Literacy             183
Ph.D                 120
Name: count, dtype: int64

```

```

qualification
Diploma after bachelor's degree  34842
Literacy                          34842
Master's                          34842
Ph.D                              34842
Post-secondary diploma            34842
Primary                           34842
middle                            34842
secondary                          34842
unknown                           34842
Name: count, dtype: int64

```

---

```

Random Forest:
Accuracy: 0.731535812232923
Precision: 0.742003339080411
Recall: 0.731535812232923
F1 Score: 0.7258920435849853

```

## Appendices E

Phase/Activity	Time Frame and Details
Phase 1: Project Initiation and Planning	Weeks 1- 2: Formulate project objectives, assemble team, develop project plan.
Phase 2: Data Collection and Preprocessing	Weeks 3 - 4: Collect and preprocess data, establish data management systems.
Phase 3: Exploratory Data Analysis	Weeks 5 - 6: Conduct exploratory analysis, refine data preprocessing.
Phase 4: Machine Learning Model Development	Weeks 7- 8: Select and implement algorithms, train/test models, model evaluation.
Phase 5: Dashboard Development and Visualization	Weeks 9 - 11: Design and develop dashboard, test functionality.
Phase 6: Analysis and Reporting	Weeks 12 -13: In-depth data analysis, prepare final reports.
Phase 7: Project Review and Closure	Weeks 14 -15: Review project outcomes, document lessons, formal closure.
Ongoing Activities	Throughout the Project: Regular meetings, quality assurance, stakeholder communication.

Table 2:Timetable

Role	Responsibilities
Project Leader: Fadiyah alanzi	Overseeing project development, Coordination between team members, Liaison with academic and university authorities
Data Analyst: Ghazlan alanazi	Data collection and processing, Implementing machine learning algorithms, Data interpretation and analysis
Software Developer: Wejdan alharthi	Developing and maintaining project software, Integrating machine learning models into software, Dashboard creation for data visualization
Research Specialist: Elham khatim	Literature review and background research, Ensuring research methodologies are followed, Preparing reports and documentation
Quality Assurance: Ebtisam falih	Testing and validating software functionality, Ensuring data accuracy and integrity, Overseeing the final project deliverables for quality

Table 3:Team and Roles



Figure 44: Gantt Chart

## Appendices F

- scikit-learn it is a library in the Python language that provides many supervised and unsupervised learning algorithms.
- Random Forest is a powerful and versatile supervised machine learning algorithm that grows and combines multiple decision trees to create a “forest”
- Accuracy: Accuracy measures the proportion of true results (both true positives and true negatives) among the total number of cases examined.
- Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positives. It is a measure of a classifier's exactness.
- Recall: Recall is the ratio of correctly predicted positive observations to all observations in actual class - yes. It is a measure of a classifier's completeness.
- F1-score: The F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. It is a balance between Precision and Recall.