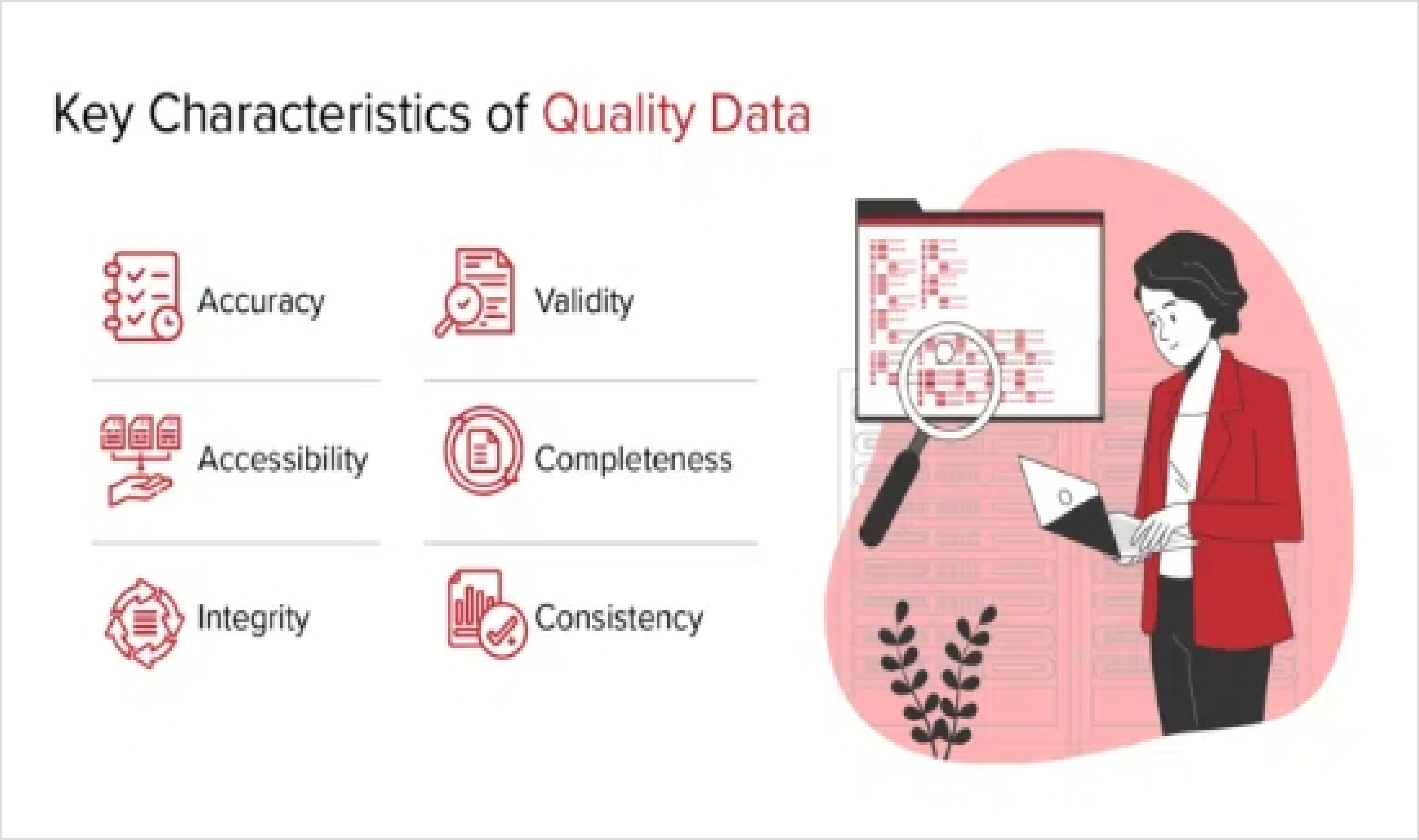
Process and clean the collected data to ensure its quality and accuracy

Data collection:It is the process of gathering data for use in business decision-making, strategic planning, research and other purposes. It's a crucial part of data analytics applications and research projects: Effective data collection provides the information that's needed to answer questions, analyze business performance or other outcomes, and predict future trends, actions and scenarios.

Data cleansing: It also known as data scrubbing or data cleaning, is the process of detecting and removing or correcting errors, inconsistencies, inaccuracies, and duplicates in a dataset. It involves identifying and rectifying any discrepancies or anomalies to ensure that the data is accurate, complete, and reliable

Importance of Data Cleansing Process:A variety of errors and problems in B2B data sets, such as inaccurate, invalid, incompatible, and corrupt data, are addressed by data cleansing. Some of these issues are brought on by human error when data is entered, while others occur due to inconsistent data structures, formats, and terminologies in various systems across an organization.



Key Characteristics of Quality Data

- Accuracy
- Validity
- Accessibility
- Completeness
- Integrity
- Consistency

Typos and incomplete or incorrect data: Different structural errors in data sets are fixed by data cleansing. This includes typographical errors, incorrect numerical inputs, syntax errors, and missing values, such as empty or null fields that should have been filled with data.

Inconsistent data: Different systems frequently use different formats for names, addresses, and other attributes. For instance, a customer’s middle initial might be present in one data set but not in another. Data cleansing makes sure that the data is consistent, enabling accurate analysis.

Duplicate data: Data cleansing uses deduplication techniques to identify duplicate records in data sets and remove or merge them. For instance, when data from two different systems are combined, duplicate data entries can be resolved to produce a single record. Duplicate data can account for 20 of a company's data asset

Irrelevant data: Some data, such as outliers or old entries, may be irrelevant to analytics applications and can skew their results. Data cleansing eliminates redundant data from data sets, which speeds up data preparation and reduces the amount of data processing and storage needed.

Benefits of Data Cleansing: Unlock the Power of Clean Data

Data cleansing plays a vital role in ensuring high-quality data and deriving meaningful insights from it. Here are a few of the many benefits you can derive from it:

1. **Staying Organized**

Businesses today gather a lot of B2B data from clients, product users, and other sources including B2B data providers. These details range from addresses and phone numbers to bank information and more. Regular data cleansing helps maintain order. It can then be more proficiently and securely stored.

2. Error Prevention

Bad data doesn’t just affect campaign analytics, daily operations are also impacted. For instance, marketing departments typically have a large database of customers. Data cleansing ensures they have access to useful, accurate information. On the other hand, cost of bad data is chaos, like using the wrong name in personalized email blasts.

3. Increased Productivity

By regularly updating and cleaning data, erroneous data is quickly and effectively eliminated. Teams will no longer need to search through outdated databases or documents to find what they need.

The Future of Data Cleansing

The future of data cleansing is expected to involve advancements in technology and techniques that streamline and automate the process,

1. Artificial Intelligence (AI) and Machine Learning (ML)

AI and ML technologies are likely to play a significant role in the future of data cleansing. These technologies can analyze vast amounts of data, identify patterns, and automatically cleanse and correct errors. ML algorithms can learn from historical data cleansing processes to improve accuracy and automate repetitive tasks, reducing the need for manual intervention.

2. Data Quality Assessment

Future data cleansing processes will likely include advanced data quality assessment techniques. These techniques can evaluate the quality of data based on various parameters such as completeness, accuracy, consistency, and integrity. By identifying and quantifying data quality issues, organizations can prioritize cleansing efforts and allocate resources more effectively.

Program:

```
#imports necessary libraries to do basic things on the dataset
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
print('Successfully imported')
#Reading data
data = pd.read_csv('project2.csv')
data.head(30)
df = pd.read_csv("project2.csv")
df.shape
df.info()
df.isna().sum()
```

```
self_employed_percent =
(df["self_employed"].isnull().sum()/len(df["self_employ
ed"]))*100
work_interfere_percent =
(df["work_interfere"].isnull().sum()/len(df["work_interfe
re"]))*100
print(f"The percentage of missing values in
self_employed column is
{round(self_employed_percent, 2)}%")
print(f"The percentage of missing values in
work_interfere column is
{round(work_interfere_percent,
2)}%")df["self_employed"] =
df["self_employed"].fillna(df["self_employed"].mode()[0]
)
df["work_interfere"] =
df["work_interfere"].fillna(df["work_interfere"].mode()[0])
df.head()
df.drop(["state", "comments"], axis=1,
inplace=True)
df.isna().sum()
df.columns
plt.figure(figsize=(17,5))
ax = sns.countplot(x='Timestamp.3', data=df)
ax.set_xticklabels(
    ax.get_xticklabels(),
    rotation=45,
    horizontalalignment='right'
)
for p in ax.patches:
    ax.annotate(p.get_height(), (p.get_x()+0.25,
p.get_height()+0.01), ha='center')
```

```
min_age = df["Timestamp.1"].min()
max_age = df["Timestamp.1"].max()
mean_age = df["Timestamp.1"].mean()
median_age = df["Timestamp.1"].median()
print(f"Min: {min_age}, \nMax: {max_age},
\nMean: {mean_age}, \nMedian: {median_age}")

df["Timestamp.1"].unique()

negative_age = (df["Timestamp.1"]<0).sum()
over_age = (df["Timestamp.1"]>80).sum()
print(f"Number of negative age entries:
{negative_age}\nNumber of overage: {over_age}")
```

```
df["Timestamp.1"].unique()
df["Timestamp.1"].hist()

sns.boxplot(x=df["Timestamp.1"])

sns.boxplot(x=df["Timestamp.1"])
sns.boxplot(x=df["Timestamp.1"])
```