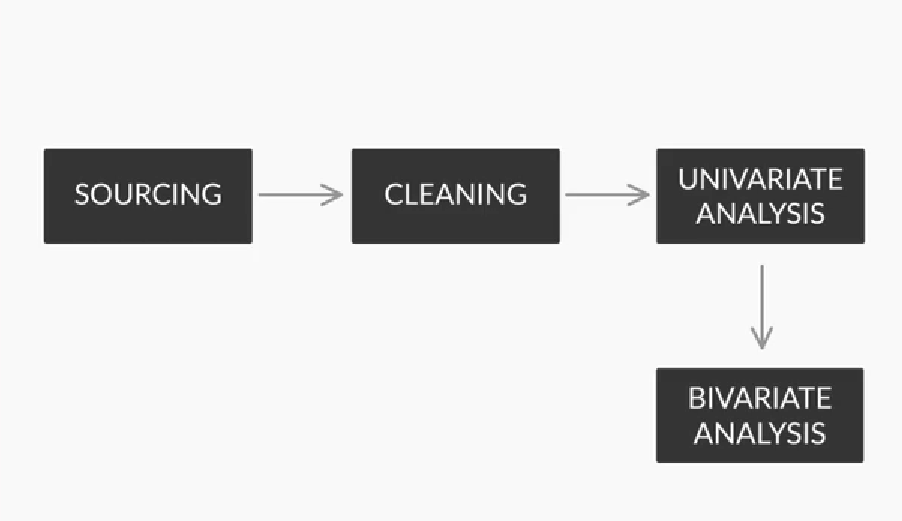
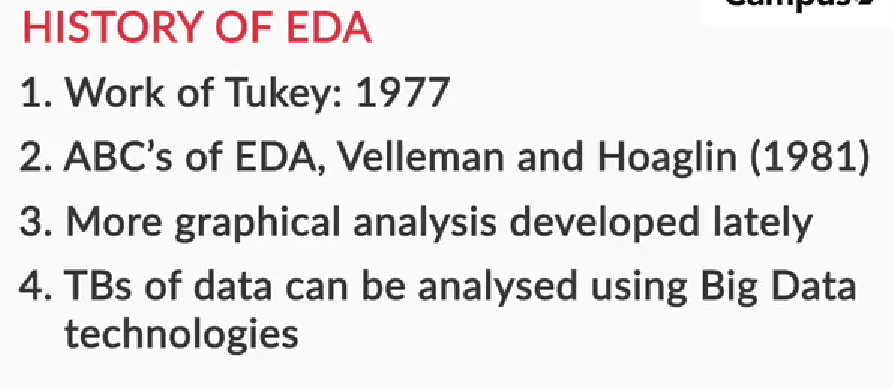
**DATA PREPROCESSING PATHS**



**Introduction to EDA**

****

**Public and Private Data**

Broadly, data sources can be of two types:

* **Private data**: As the name suggests, it is private and belongs to an organisation, and there are certain security and privacy concerns attached to it. It is used for companies’ internal analysis purposes in order to gain business and growth insights. Some examples of private organisational data are**telecom data**, **retail data**, **banking** and **medical data**.
* **Public data**: This is the data that is available for public use and is offered by many sites such as government websites and public agencies for the purpose of research. Accessing this data does not require any special permission or approval, hence the name.

Additionally, there are many programming techniques that are used to fetch public data through code, which you will learn about later in the module.

So, private data has some security and privacy issues and is not publicly available. You learnt about the use of data in the banking, telecom and human resources sectors. Lets look at the different kinds of Private data:

**PRIVATE DATA:-**

* **Banking data**: Banks use data to make credit-related decisions. This data is highly sensitive as it contains customer transaction details, account details, etc. Security of such data is of topmost importance. Banks can use such data to predict which customer is likely to take a loan in the near future or which customers are interested in investing in term deposits, etc. With the help of such data, banks can also identify the customers who are likely to default on their loans.
* **Telecom data**: Telecom companies use data to optimise their plans for customers and predict customer churn. Telecom data can be used to optimise the coverage area based on customers’ call data and their call performances.
* **HR data**: HR data analytics helps identify and predict employee behaviour.
* **Retail data**: Retail data analytics helps drive decisions such as product purchasing, pricing and stocking.
* **Media data:** The media industry uses data extensively to target viewers. Advertisers use data to identify the best avenues for targeting customers, while journalists use data visualisation to gather relevant information.

**Public Data:-**

* **GitHub**: [Awesome Public datasets](http://github.com/awesomedata/awesome-public-datasets), [GitHub data sets](http://github.com/datameet)
* **Open government data set**: [Open government data](http://data.gov.in/)

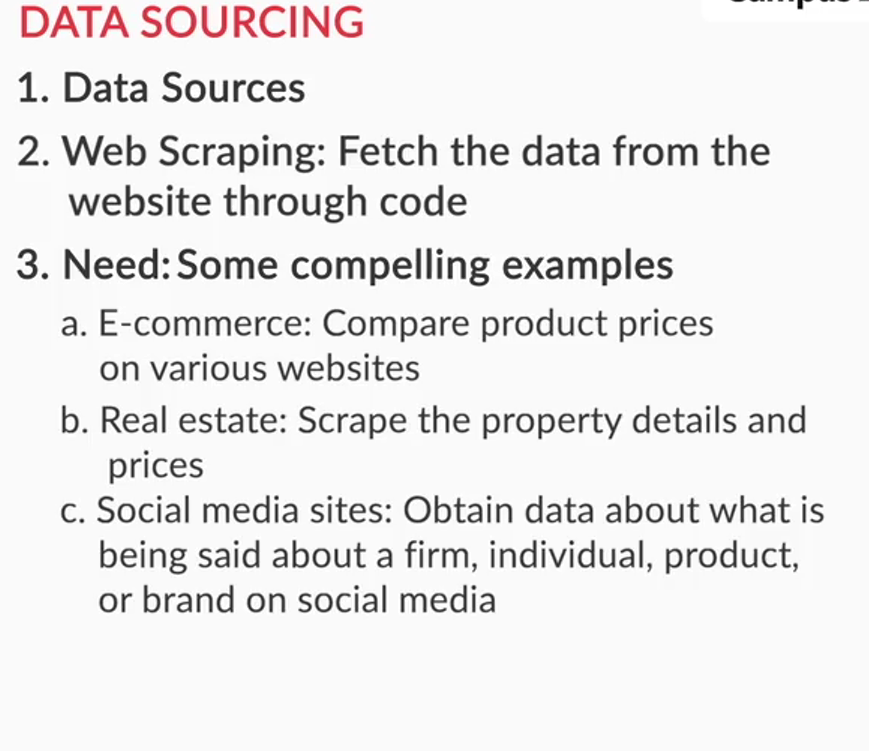
In the next segment, you will be solving some assessments based on 2011 Census data.

**Public Data Exercise**

Census in India is a massive data collection exercise conducted every 10 years by the Registrar General and Census Commissioner of India under the Ministry of Home Affairs, Government of India.

The Census in 2011 was conducted in two phases: house listing and population enumeration. The Indian Population Census, 2011 covers population characteristics such as gender, religion, education and age, among other important parameters. When this data is analysed at a macro level, such as a district or a state, it presents a bigger picture, which helps the government make essential decisions and frame the requisite policies.

**Web Scraping**

****

An HTML page broadly consists of two basic elements:

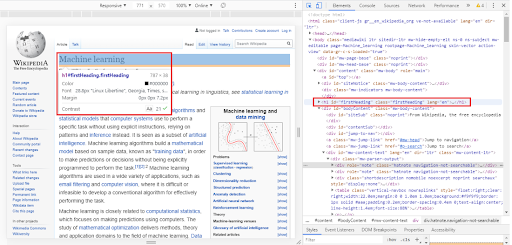
* **Attributes**: These are used to describe the characteristics of an element. They majorly contain the **class**, **id** and **href**. These are like objects that are created to define the different segments of a web page.
* **Tags**: A tag is a way to represent an HTML element. Tags majorly contain**h (heading), p (paragraph), a (hyperlink)**and**div**.

Let’s briefly go through the attributes one by one using the Wikipedia page examples:

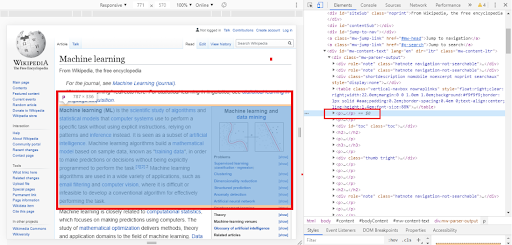
* **Class**: The HTML class attribute is used to specify a single or multiple class names for an HTML element.
* **ID**: This attribute is used to provide a specific ID to an element.
* **href**: This attribute is used to provide any web page link that is embedded in the text on the HTML page.

A group of elements may have the same attributes but will have different tags. Let’s go through the tags using the Wikipedia page examples to understand the concept better:

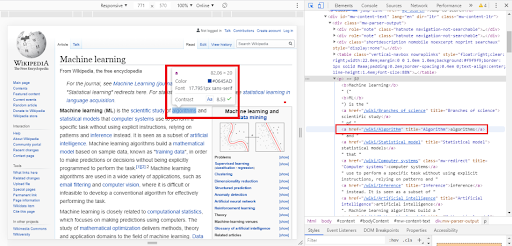
* **Heading:**It is represented by ‘h’ in HTML code and is used to place the headings of sections on a web page, as shown in the following image.  
    
  **Heading Tag**



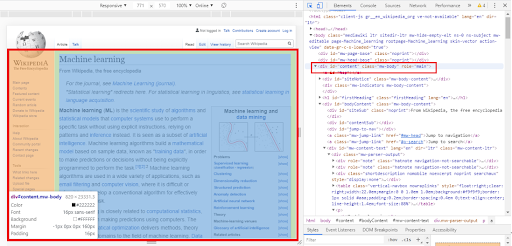
* **Paragraph**: It is represented by ‘p’ in HTML code and is used to place a paragraph on the web page, as illustrated in the following image.  
    
  **Paragraph Tag**



* **Hyperlink:** It is represented by ‘a’ in HTML code and is used to provide a link to any other web page on the present web page. This is shown in the following image.  
    
  **Hyperlink Tag**



* **Div:** It is used to structure the HTML page and is a nested structure that contains other HTML elements. The main purpose of the div tag is to promote encapsulation. This is illustrated in the image below.  
    
  **Div Tag**

****

* **Span:** This tag is used for grouping and applying styles to inline elements.

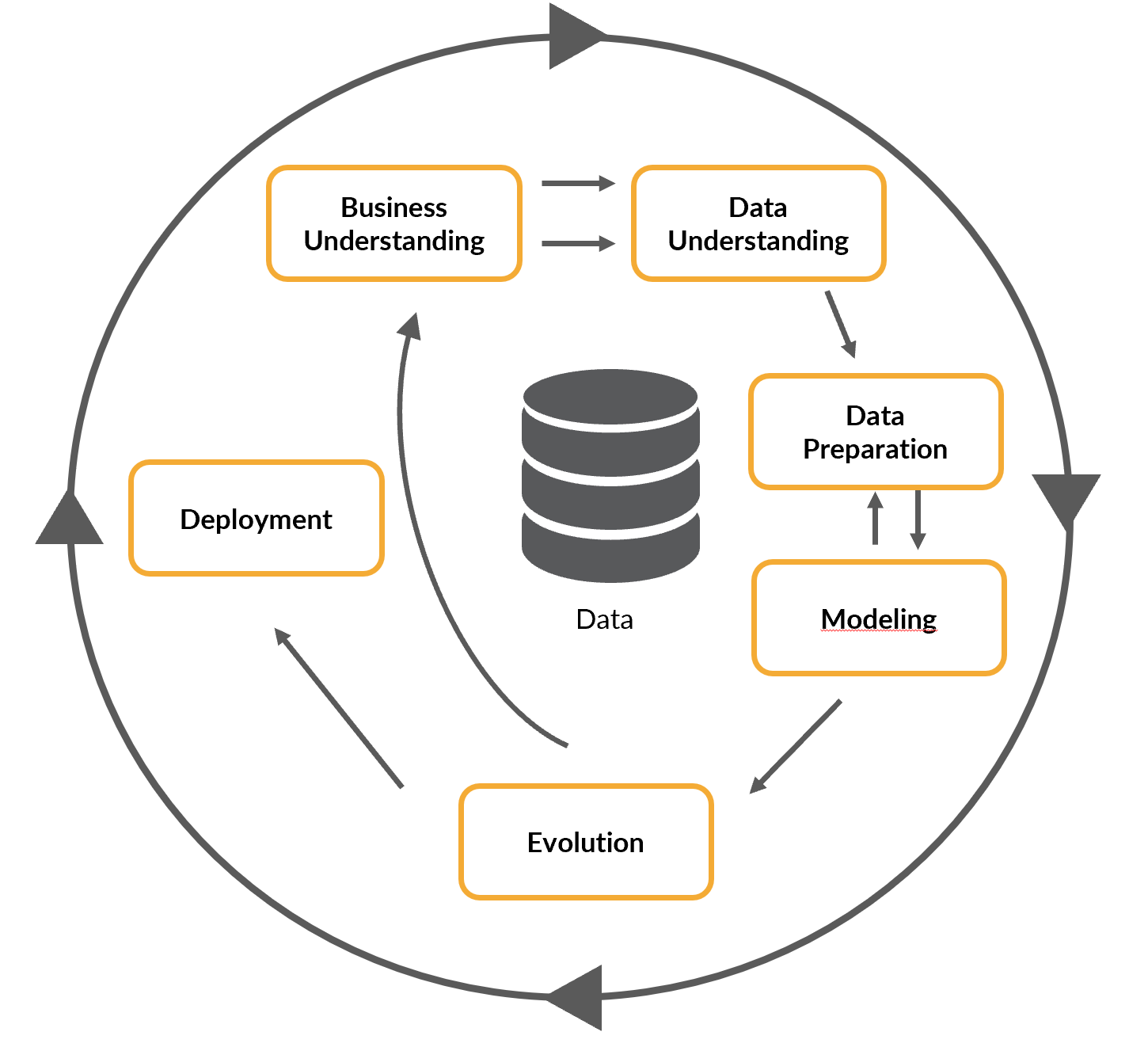
**Data Sources**

Given below are the links to some public datasets. You can explore these sources to get the data:

* **GitHub:** [Awesome public data sets](https://github.com/awesomedata/awesome-public-datasets), [GitHub data sets](https://github.com/datameet)
* **Open government data set:** [Open government data](https://data.gov.in/)
* **Kaggle website link:**[Kaggle Website](https://www.kaggle.com/datasets)
* **UCI repository of machine learning:**[UCI machine learning data set repository](https://archive.ics.uci.edu/ml/index.php)

**Intro to Data Sourcing**

Before we delve into sourcing data, let us first understand the **Cross Industry Standard Process for Data Mining (CRISP-DM)** framework which is used to explain the data-science life cycle. It is shown below.  
  
**CRISP-DM Framework**



1. **Business understanding:** Understand the business need and the goal of the project.
2. **Data understanding:** Understand the data needed for the project.
3. **Data preparation:**Plan how to organise the data.
4. **Modelling:** Decide on the modelling techniques to be used.
5. **Evaluation:** Evaluate if the model gives the desired results.
6. **Deployment:** Decide on the deployment strategy.

The following table summarises the different types of data.

|  |  |
| --- | --- |
| **Private Data** | **Public Data** |
| This is private data, which belongs to an organisation. | This data is available for public use and is offered by many websites such as government websites and public agencies. |
| Certain security and privacy concerns are attached to this data type. | Accessing this data does not require any special permission or approval. |
| It is used for a company’s internal analysis purposes to gain business and growth insights. | It is mainly used for the purpose of research. |
| Examples: Telecom data, retail data, banking and medical data | Examples: You can get public data from websites like kaggle.com and github.com. |

**Web Scraping**

You learnt web scraping  in the live session and understood how you can fetch data from websites using code. Web scraping majorly involves four steps:

1. **HTML loading and reading:** It includes the loading of the HTML page into Python. The library which is used here to request for the HTML page is the **‘request’** library.
2. **HTML parsing:** This step involves the process of presenting HTML code into a readable format. One of the important classes of Python called **‘BeautifulSoup’** is used here to parse the data.
3. **Data extraction:** This step involves the extraction of data from the web page using HTML elements like **tags and attributes**.
4. **Transformation into the required format:** Once you have the data, you can save it into the required format, like **CSV**.

The basic requirement of web scraping is the web page that we are going to scrape. All web pages are written in HTML. So, you can perform web scraping using Python only after you understand the basic structure of an HTML page.

An HTML page broadly consists of two basic elements:

* **Attributes:** These are used to describe the characteristics of an element. They majorly contain the class, id and href. These are like objects that are created to define the different segments of a web page.
* **Tags:** A tag is a way to represent an HTML element. Tags majorly contain h (heading), p (paragraph), a (hyperlink) and div.

The following table summarises the different attributes used in an HTML page.

|  |  |
| --- | --- |
| **Attribute** | **Application** |
| class | It is used to specify a single or multiple class names for an HTML element. |
| id | It is used to provide a specific ID to an element. |
| href | It is used to provide any web page link that is embedded in the text on the HTML page. |

The following table summarises some of the commonly used tags in an HTML page.

|  |  |
| --- | --- |
| **Tags** | **Application** |
| Paragraph <p> | It is used to place a paragraph on the web page. |
| Hyperlink <a> | It is used to provide a link to any other web page on the current web page. |
| <div> | It is used to structure an HTML page. It is a nested structure that contains other HTML elements. The main purpose of the div tag is to promote encapsulation. |
| <span> | It is used for grouping and applying styles to inline elements. |
| <title> | It is used to define the title of a web page. |
| <h1> to <h6> | It is used to define headings with <h1> being the most important heading and <h6> being the least important heading. |
| <body> | It is used to define the body of a web page. It contains all other HTML elements like <p>, <h1> and <h6>. |
| <img> | It is used to embed an image into an HTML web page. |
| <header> and <footer> | These are used to add header and footer in an HTML web page. |
| <ol>,<ul> and <li> | <ol> is used to add an ordered list, <ul> for unordered list and <li> for a list item in lists. |

**Intro to Data Cleaning**

**Data Formatting**

**Data formatting** is the process of arranging or converting data into a specific structure, layout, or style so that it becomes easy to read, use, or analyze. It ensures that data is consistent, clean, and properly organized for tasks like reporting, analysis, or data entry.

**Example:**

If you have a date written as 20250416, you can format it as:

* 16-04-2025
* April 16, 2025
* 16/04/25

All these are different **formats** of the same data.

**Why is data formatting important?**

* Makes data easier to understand
* Helps in sorting and filtering
* Ensures consistency
* Avoids errors in calculations and analysis

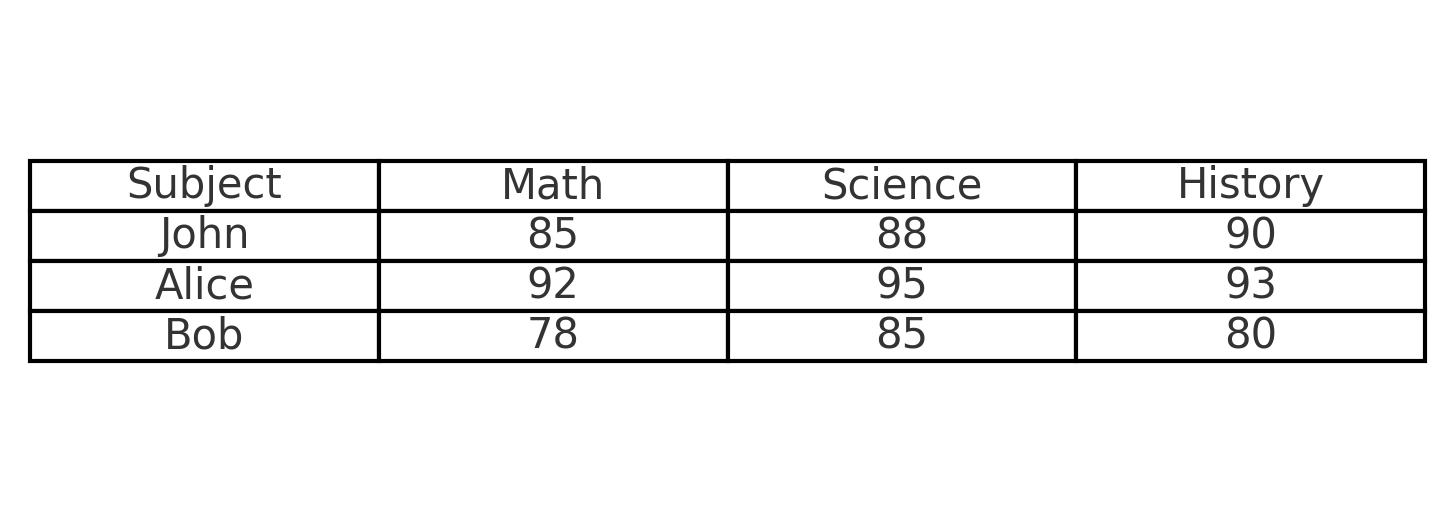
**Common types of data formatting:**

* **Text formatting:** Changing case (UPPER, lower), removing spaces
* **Number formatting:** Adding commas, currency symbols, decimal places
* **Date/time formatting:** Changing how dates and times are displayed
* **Conditional formatting:** Highlighting data based on rules (e.g., red if value < 0)

The difference between long and wide format tables primarily relates to how data is organized,

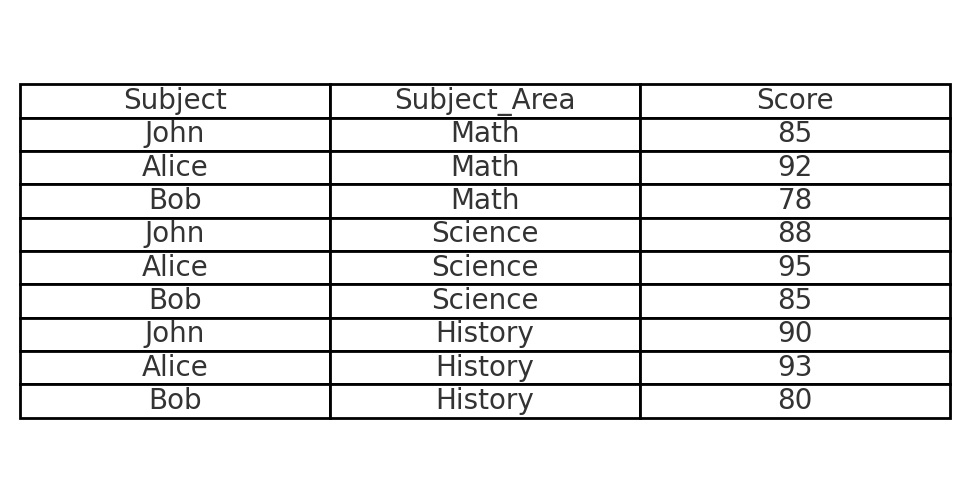
**Wide Format Table**

As shown in the image below, each subject is a different column and each person has a unique row with all their subject marks specified. It is easy for humans to interpret data from a wide format table.



**Long Format Table**

As shown in the image below, in this format each row specifies a data entry in the table. Long format table is prefered for data analysis because it allows for better data manipulation.



**Program:** Professional Certificate Program in AI and Data Science | **Course:** Python for Data Science

**Intro to Data Cleaning**

Intro to Data Cleaning

**Data Formatting**

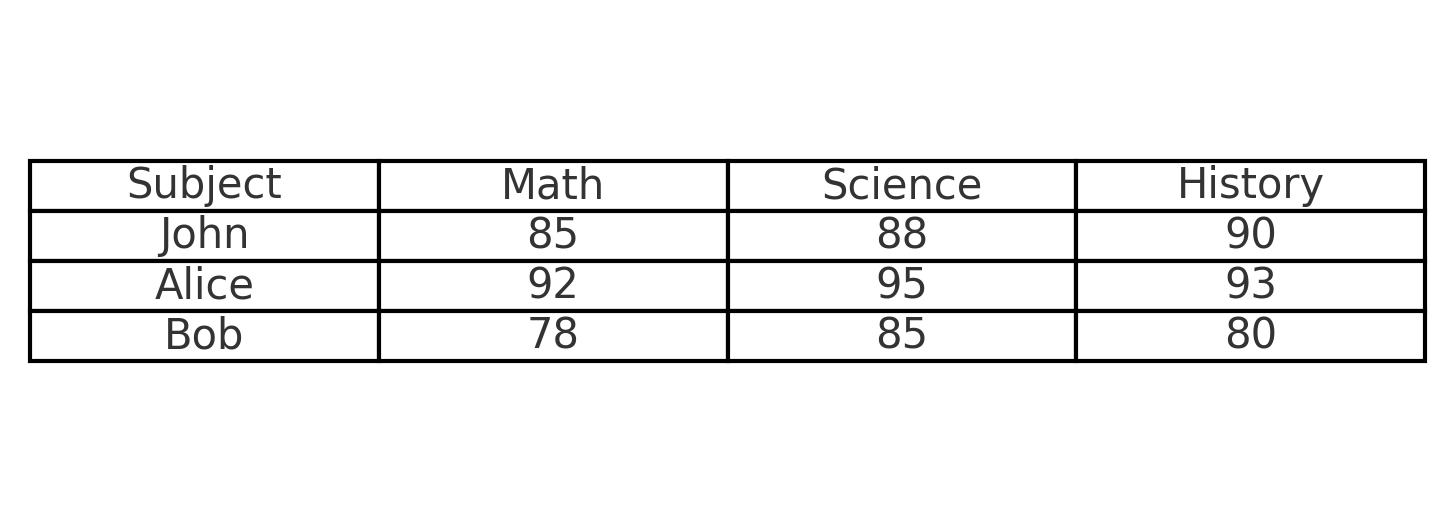
Data formatting involves converting raw data into a format that is readable and can be used for analysis.

Let's start by discussing about wide form and long form of tables.

The difference between long and wide format tables primarily relates to how data is organized,

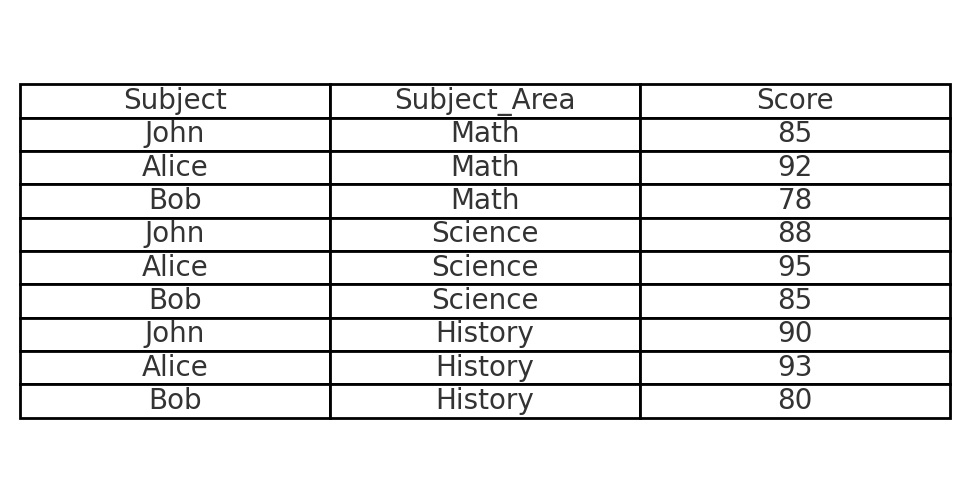
**Wide Format Table**

As shown in the image below, each subject is a different column and each person has a unique row with all their subject marks specified. It is easy for humans to interpret data from a wide format table.



**Long Format Table**

As shown in the image below, in this format each row specifies a data entry in the table. Long format table is prefered for data analysis because it allows for better data manipulation.



**1. Identifying the data type:**In general, any given data set is expected to have different types of data. Following are some data types with their examples.

|  |  |  |
| --- | --- | --- |
| **Example** | **Variable Type** | **Data Type** |
| Weight, age | Numerical variable | int, float |
| Size of clothes(S,M,L,etc), type of jobs, blood group | Categorical variable | Object |
| Income, grades in exams, education level, integer ratings | Ordinal categorical type | Object, int, float |
| Date, time, timestamp | Date and time variable | Date and time |

**Note:**Ordinal categorical type means that there is a clear ordering of the category. Example: For income, there are low income, average income, and high income.

**2. Fixing the rows and columns:** The following table is the checklist for fixing rows and columns.

|  |  |
| --- | --- |
| **Fixing rows** | **Fixing columns** |
| Delete summary rows, incorrect rows, and extra rows | If needed, merge columns for creating unique identifiers or split columns to get more data, add/rename column names, delete columns, align misaligned columns |

**3. Impute/remove Missing Values:** Sometimes, it is good to drop the missing values because they are missing completely at random, and sometimes, it is good to impute them with another value. Before summarising the imputation, let’s recall the three types of missing values: MCAR (Missing completely at random), MAR (Missing at random) and MNAR (Missing not at random). Following table summarises the imputation on categorical/numeric columns.

|  |  |
| --- | --- |
| **Categorical column** | **Numerical column** |
| Impute the most popular category. | Impute the missing value with mean/ median/ mode. |

Sure! Let’s break down the **three types of missing values** — MCAR, MAR, and MNAR — in a simple and clear way.

**✅ 1. MCAR (Missing Completely At Random)**

**📌 Meaning:**

The missing data has **no relation** to any other data — it’s just missing **purely by chance**.

**💡 Example:**

A survey respondent accidentally skips a question because they were distracted. Their answer is missing, but it’s **not related to any other question** or variable.

**📊 Impact:**

* It’s the **least harmful** kind of missing data.
* You can **safely remove** or impute these values without much bias.

**✅ 2. MAR (Missing At Random)**

**📌 Meaning:**

The missing data is related to **other variables** in the dataset, but **not to the missing variable itself**.

**💡 Example:**

You’re collecting income data, and people with **lower education levels** are more likely to skip the income question. So, the missing income is related to **education**, but not income itself.

**📊 Impact:**

* You can **handle MAR data using statistical models**, like regression or imputation based on other columns.
* Slightly more complex than MCAR, but still manageable.

**✅ 3. MNAR (Missing Not At Random)**

**📌 Meaning:**

The missing data is related to the **value that’s missing itself**.

**💡 Example:**

People with **very high income** may choose **not to disclose** their income. So, the missing income depends on the fact that the income is high.

**📊 Impact:**

* This is the **most difficult** to handle.
* Requires advanced techniques, assumptions, or additional data to deal with properly.
* Simple imputation might introduce **bias**.

**🔁 Summary Table:**

| **Type** | **Reason Missing** | **Can be Predicted from Other Variables?** | **Easy to Handle?** |
| --- | --- | --- | --- |
| MCAR | Pure randomness | ❌ No | ✅ Yes |
| MAR | Related to other variables | ✅ Yes | ⚠️ Moderate |
| MNAR | Related to the missing value itself | ❌ No (or very hard) | ❌ No |

Let me know if you want visuals, examples in Python, or how to handle them using pandas or Power BI!

Great question!

**👉 Impute (in data science/statistics) means:**

**To fill in or replace missing values** in a dataset with **estimated or calculated values**.

**🔧 Why do we impute missing data?**

* So that **algorithms can run smoothly** (most models can't handle missing values).
* To **preserve** as much data as possible.
* To **reduce bias** caused by dropping rows or leaving gaps.

**🔄 Common Imputation Techniques:**

| **Method** | **How it works** | **When to use** |
| --- | --- | --- |
| **Mean imputation** | Replace missing value with column mean | Numeric data with normal distribution |
| **Median imputation** | Replace with column median | Skewed numeric data |
| **Mode imputation** | Replace with most frequent value | Categorical data |
| **Forward fill** | Fill with previous value | Time series data |
| **Backward fill** | Fill with next value | Time series data |
| **Model-based** | Use machine learning to predict missing | Complex or MAR/MNAR data |

**4. Fixing Invalid Values :** Following is a checklist about fixing invalid values in a data set.

* Encode unicode properly
* Convert incorrect data types
* Correct the values that lie beyond the range
* Correct the values not belonging in the list
* Fix incorrect structure
* Validate internal rules, if present.

**5. Standardising Values:**Following table is a checklist to standardise the numeric/text values.

|  |  |
| --- | --- |
| **Standardising numeric values** | **Standardising text values** |
| Standardise unit, scale value and precision for a better presentation of data | Remove extra characters, standardise case and format |

However, in the case study, you performed another kind of standardisation using the **StandardScaler** function. This was done because different columns in a dataset may have different ranges of values. The varying scales can affect the performance of statistical models. Hence, such a standardisation was necessary. If not done, the feature with the larger numeric range may dominate, although it may not be necessarily important. Similar is the case with normalisation, which was done using the **MinMaxScaler** function.

**6. Handling Outliers:**Outliers are values that are much beyond or far from the next nearest data points. The following table summarises the two types of outliers.

|  |  |
| --- | --- |
| **Univariate outliers** | **Multivariate outliers** |
| Univariate outliers are those data points in a variable whose values lie beyond the range of expected values. | While plotting data, some values of one variable may not lie beyond the expected range, but when you plot the data with some other variable, these values may lie far from the expected value. These are called multivariate outliers. |
| Example: Here, almost all the points lie between 0 and 5.0, and one point is extremely far away (at 20.0) from the normal norms of this data set. | Example: You can refer to the image below to get a better understanding of multivariate outliers.          If you look individually at the outlier point in the graph, it lies within the range of values but when compared it with the other values. it doesn’t lie in the expected value range. |

The major approaches to the treatment of outliers can include: Imputation, deletion of outliers, binning of values and capping the outliers. In the case study, you used **boxplots** to observe the outliers and the **IQR method** to handle the same.

**7. Filter Data:** You need to filter the data in order to get what is needed for your analysis.  Following is a checklist for filtering the data.

* Deduplicate data, that is remove identical rows and the rows in which some columns are identical.
* Filter rows relevant to the analysis.
* Filter columns relevant to the analysis.
* Aggregate data

It is very important to get rid of such irregularities to be able to analyse a dataset. Otherwise, it may hamper further analysis of the dataset.

**8. Sanity Checks:** Lastly, you need to ensure that the data that is available to you makes sense. For eg: In the case study, the app ratings should be between 1 and 5, the free apps should have a price of 0, and the no. of app reviews should be lesser than the no. of app installs. The tasks to be done as part of the sanity check will change based on the dataset.

**Additional Reading**

[**Outliers explained:**](https://towardsdatascience.com/outliers-analysis-a-quick-guide-to-the-different-types-of-outliers-e41de37e6bf6)You can visit this website to read more about the different types of outliers  
[**Exploratory Data Analysis and Data Cleaning Practical Workout:**](https://towardsdatascience.com/exploratory-data-analysis-and-data-cleaning-practical-workout-2a20442b42fb)Visit this link to understand how to prepare the data for EDA.

**Session Summary**

**Data Format Types:**

* Wide Format: Each variable has its own column, useful for comparison.
* Long Format: Each observation is in a separate row, more flexible for analysis. Reshaping between formats may be necessary.

**Handling Missing Data:**

* Filling missing values with mean, median, or other estimates.
* Dropping rows or columns with excessive missing data.

**Handling Corrupt Data:**

* Identifying issues such as invalid formats or out-of-range values.
* Correcting through data type conversion, replacing values, or manual review.

**Handling Outliers:**

* Detecting outliers using  IQR, or visualization.
* Transforming, eliminating, or keeping extreme values if they are important.

**An outlier** is a data point that differs significantly from other observations in a dataset. It lies far away from the rest of the values and can affect statistical analyses, especially averages and trends.

**📌 Definition:**

An outlier is an observation that lies an abnormal distance from other values in a dataset.

**✅ Types of Outliers:**

**1. Univariate Outliers**

* Involves **only one variable**.
* Example: In a dataset of student heights, a value of 250 cm might be an outlier.

**2. Multivariate Outliers**

* Involves **more than one variable**.
* A data point may not be an outlier in any one variable but may be unusual when considering two or more variables together.
* Example: A person with very high income but very low spending.

**3. Global Outliers (Point Outliers)**

* These are data points that are far from all other data.
* Most common type.
* Example: Temperature reading of 100°C in a room where usual readings are around 25°C.

**4. Contextual Outliers (Conditional Outliers)**

* A value that is only an outlier in a specific **context**.
* Example: A temperature of 35°C is normal in summer but an outlier in winter.

**5. Collective Outliers**

* A group of data points that are outliers **together**, but not necessarily alone.
* Often found in time-series data.
* Example: A sudden drop in stock prices for a week.

**💡 Real-Life Example:**

Imagine you're analyzing exam scores:

Scores: 45, 48, 50, 52, 49, 100

Here, 100 is an outlier because it's far away from the rest of the data.

If you want, I can also explain how to detect outliers using Python, Excel, or by using formulas like IQR and Z-score.

The major approaches to the treatment of outliers can include:

* Imputation,
* Deletion of outliers,
* Binning of values and
* Capping the outliers.