SCREEN SENSE – KID’S SCREENTIME VISUALIZATION

A PROJECT REPORT

BY

PRABAKARAN N

1.OBJECTIVE

The primary goal for Week 1 was to initiate the project and establish a structured foundation for analysis. This involved defining the overall project goals and workflow, loading the dataset into the working environment, and conducting a preliminary exploration to understand its structure, data types, size, and missing values. Initial observations on data quality, potential issues, and assumptions were also documented to guide future steps. Establishing this foundation ensures that subsequent analysis is accurate, well-structured, and efficient**.**

Dataset Source: Kaggle — Indian Kids Screentime 2025

2. IMPLEMENTATION

2.1 Project Goals and Workflow

Goals:

- Create an interactive and flexible insight dashboard that is easily understandable by parents and teachers.
- Analyze the dataset to gain insights on children's screen time, device usage, and associated health impacts.
- Prepare the dataset for further analysis and modelling by identifying missing values, data types, and potential quality issues.

Workflow for the project:

- **Data Acquisition:** Collect and load datasets.
- **Data Exploration:** Examine schema, data types, missing values, and dataset size.
- **Data Cleaning and Preprocessing:** Handle null values, duplicates, and inconsistencies.
- **Exploratory Data Analysis (EDA):** Understand trends, patterns, and correlations.
- **Model Development and Evaluation:** Build predictive or analytical models.

2.2 Dataset Loading

- **Dataset Source:** Provided by Mentor
- **Dataset Format:** Excel
- **Number of Rows:** 9712 rows
- **Number of Columns:** 8 columns
- **Tools Used:** Python (pandas), Jupyter Notebook

2.3 Data Exploration

2.3.1 Importing and Viewing the Dataset

Step 01:

```
!pip install pandas
import pandas as pd
df = pd.read_csv("Indian_Kids_Screen_Time.csv")
df.head()
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.12/dist-packages (2.2.2)
Requirement already satisfied: numpy>=1.26.0 in /usr/local/lib/python3.12/dist-packages (from pandas) (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.12/dist-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.12/dist-packages (from pandas) (2025.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.12/dist-packages (from pandas) (2025.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.12/dist-packages (from python-dateutil>=2.8.2->pandas) (1.17.0)
```

	Age	Gender	Avg_Daily_Screen_Time_hr	Primary_Device	Exceeded_Recommended_Limit	Educational_to_Recreational_Ratio	Health_Impacts	Urban_or_Rural
0	14	Male	3.99	Smartphone	True	0.42	Poor Sleep, Eye Strain	Urban
1	11	Female	4.61	Laptop	True	0.30	Poor Sleep	Urban
2	18	Female	3.73	TV	True	0.32	Poor Sleep	Urban
3	15	Female	1.21	Laptop	False	0.39	NaN	Urban
4	12	Female	5.89	Smartphone	True	0.49	Poor Sleep, Anxiety	Urban
5	14	Female	4.88	Smartphone	True	0.44	Poor Sleep	Urban
6	17	Male	2.97	TV	False	0.48	NaN	Rural
7	10	Male	2.74	TV	True	0.54	NaN	Urban
8	14	Male	4.61	Laptop	True	0.36	Poor Sleep, Anxiety	Rural
9	18	Male	3.24	Tablet	True	0.48	Poor Sleep, Obesity Risk	Urban

Output:

- The dataset is loaded into a pandas Data Frame named df.
- Using commands like df.head(), the first few rows of the dataset can be viewed, showing column names and values.

Advantages:

- Confirms that the dataset is correctly imported and ready for analysis.
- Provides an initial understanding of the columns, data types, and example values.

Step 02:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9712 entries, 0 to 9711
Data columns (total 8 columns):
 #   Column                             Non-Null Count  Dtype  
---  ------                             --------------  -----  
 0   Age                                9712 non-null   int64  
 1   Gender                             9712 non-null   object 
 2   Avg_Daily_Screen_Time_hr           9712 non-null   float64
 3   Primary_Device                     9712 non-null   object 
 4   Exceeded_Recommended_Limit         9712 non-null   bool   
 5   Educational_to_Recreational_Ratio  9712 non-null   float64
 6   Health_Impacts                     6494 non-null   object 
 7   Urban_or_Rural                     9712 non-null   object 
dtypes: bool(1), float64(2), int64(1), object(4)
memory usage: 540.7+ KB
```

Output:

- Displays column names, number of non-null values, and data types (int, float, object, bool).

Advantages:

- Helps identify which columns are numeric or categorical.
- Highlights missing values (non-null count < total rows).
- Guides preprocessing steps such as handling missing values or converting data types.

Step 03:

```
df.shape
(9712, 8)
```

Output:

The dataset has 9,712 rows and 8 columns. This gives an overview of its size, helping plan preprocessing and analysis steps**.**

Advantages:

- Shows the total number of rows and columns.
- Helps plan for preprocessing, memory usage, and analysis approach

Step 04:

```
df.isnull().sum()
```

	0
Age	0
Gender	0
Avg_Daily_Screen_Time_hr	0
Primary_Device	0
Exceeded_Recommended_Limit	0
Educational_to_Recreational_Ratio	0
Health_Impacts	3218
Urban_or_Rural	0

dtype: int64

Output:

The Health Impacts column has 3,218 null values, indicating that some entries are missing. This highlights the presence of missing data that will need to be addressed during preprocessing.

Advantages:

- Identifies columns with missing values that need imputation or cleaning.
- Ensures that data quality issues are addressed before analysis or modeling.

Step 05:

```
df.describe()
```

	Age	Avg_Daily_Screen_Time_hr	Educational_to_Recreational_Ratio
count	9712.000000	9712.000000	9712.000000
mean	12.979201	4.352837	0.427226
std	3.162437	1.718232	0.073221
min	8.000000	0.000000	0.300000
25%	10.000000	3.410000	0.370000
50%	13.000000	4.440000	0.430000
75%	16.000000	5.380000	0.480000
max	18.000000	13.890000	0.600000

Output:

The summary statistics show the count, mean, standard deviation, minimum, maximum, and quartiles for all numeric columns

Advantages:

This helps identify outliers, unusual values, and understand the distribution of numeric data, guiding preprocessing decisions like normalization or handling extreme values.

Step 06:

```
df.nunique()
```

	0
Age	11
Gender	2
Avg_Daily_Screen_Time_hr	899
Primary_Device	4
Exceeded_Recommended_Limit	2
Educational_to_Recreational_Ratio	31
Health_Impacts	15
Urban_or_Rural	2

dtype: int64

Output:

The number of unique values in each column is as follows: Age has 11 unique values, Gender has 2, Average Daily Screen Tim Hour has 899, Primary Device has 4, Exceeded Recommended Limit has 2, Educational to Recreational Ratio has 31, Health Impacts has 15, and Urban or Rural has 2.

Advantages:

This information is useful for understanding categorical variables and planning encoding or grouping strategies for analysis and visualization.

Step 07:

```
df.dtypes.value_counts()
```

	count
object	4
float64	2
int64	1
bool	1

dtype: int64

Output:

The dataset contains 4 object-type columns, 2 float64 columns, 1 int64 column, and 1 boolean column

Advantages:

This overview helps determine which columns require type conversion or special preprocessing before modelling or visualization.

Step 8:

```
df.duplicated().sum()
```

```
np.int64(44)
```

Output:

The dataset contains 44 duplicate rows where all column values are identical.

Advantages:

Identifying duplicates ensures data integrity and prevents redundancy from affecting analysis or model performance.

Google Collab Workspace:

The project code, data exploration, and initial analysis for Week 1 have been implemented in a Google Collab notebook. The notebook can be accessed at the following link:

View Google Collab Notebook

3. CONCLUSION

The data exploration provided a comprehensive understanding of the dataset's structure, quality, and variability. The dataset contains 9,712 rows and 8 columns, with mostly complete data, except for some missing values in Average Daily screen time Hour and Health Impacts.

The column data types include 4 object, 2 float, 1 int, and 1 Boolean, guiding decisions for type conversion and encoding. There are 44 duplicate rows that should be addressed to maintain data integrity. Unique value analysis shows that categorical columns like Gender, Primary Device, and Exceeded Recommended Limit have few distinct values, while numeric columns like Average Daily screen time Hour have high variability.

These insights will inform the preprocessing steps, including handling missing data, removing duplicates, encoding categorical variables, and preparing the dataset for analysis, modelling, and dashboard visualization.