DATA ANALYSIS

SMART WATCH ANALYSIS

AGENDA

- Introduction
- Data collection
- Data cleaning
- Analysis and visualization
- summary

INTRODUCTION

- In the fast-paced world we live in, staying fit and leading a healthy lifestyle has become an essential goal for many individuals. As a result, the fitness technology industry has experienced remarkable growth, introducing innovative devices that empower users to monitor and optimize their physical activities. Among these cutting-edge devices, smartwatches have emerged as a popular choice, offering multifunctional features that extend beyond timekeeping. One of the most significant and widely used functions of these smart wearables is the accurate tracking and analysis of caloric expenditure.

- This analysis aims to explore the pivotal role of smartwatches in providing users with insightful data on their daily caloric burn. Understanding caloric expenditure is crucial for individuals seeking to maintain a balanced diet, achieve weight loss or gain goals, and optimize their fitness routines. By leveraging advanced sensors and algorithms, modern smartwatches can accurately estimate the number of calories burned throughout the day, enabling users to make informed decisions about their physical activities and nutrition.

DATA COLLECTION

- First we import the libraries and proceed to print the information header

```
[ ]  import pandas as pd
     import numpy as np
     import matplotlib.pyplot as plt
     import plotly.express as px
     import plotly.graph_objects as go
```

- Next importing data set

```
[ ]  data = pd.read_csv("/content/dailyActivity_merged.csv")
     print(data.head())
```

OUTPUT

```
           Id ActivityDate  TotalSteps  TotalDistance  TrackerDistance  \
0  1503960366    4/12/2016       13162           8.50             8.50
1  1503960366    4/13/2016       10735           6.97             6.97
2  1503960366    4/14/2016       10460           6.74             6.74
3  1503960366    4/15/2016        9762           6.28             6.28
4  1503960366    4/16/2016       12669           8.16             8.16

   LoggedActivitiesDistance  VeryActiveDistance  ModeratelyActiveDistance  \
0                       0.0                1.88                      0.55
1                       0.0                1.57                      0.69
2                       0.0                2.44                      0.40
3                       0.0                2.14                      1.26
4                       0.0                2.71                      0.41

   LightActiveDistance  SedentaryActiveDistance  VeryActiveMinutes  \
0                 6.06                      0.0                 25
1                 4.71                      0.0                 21
2                 3.91                      0.0                 30
3                 2.83                      0.0                 29
4                 5.04                      0.0                 36

   FairlyActiveMinutes  LightlyActiveMinutes  SedentaryMinutes  Calories
0                   13                   328               728      1985
1                   19                   217               776      1797
2                   11                   181              1218      1776
3                   34                   209               726      1745
```

DATA CLEANING

- Before starting the analysis, we must verify if the information presents empty fields

```
[ ]  print(data.isnull().sum())

Id                          0
ActivityDate                0
TotalSteps                  0
TotalDistance               0
TrackerDistance             0
LoggedActivitiesDistance    0
VeryActiveDistance          0
ModeratelyActiveDistance    0
LightActiveDistance         0
SedentaryActiveDistance     0
VeryActiveMinutes           0
FairlyActiveMinutes         0
LightlyActiveMinutes        0
SedentaryMinutes            0
Calories                    0
dtype: int64
```

ONCE VERIFIED THAT THERE IS NO NULL DATA, WE CAN SEE INFORMATION ABOUT COLUMNS AND THEIR DATA TYPES

```
[ ]  print(data.info())

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 940 entries, 0 to 939
Data columns (total 15 columns):
 #   Column                    Non-Null Count  Dtype  
---  ------                    --------------  -----  
 0   Id                        940 non-null    int64  
 1   ActivityDate              940 non-null    object 
 2   TotalSteps                940 non-null    int64  
 3   TotalDistance             940 non-null    float64
 4   TrackerDistance           940 non-null    float64
 5   LoggedActivitiesDistance  940 non-null    float64
 6   VeryActiveDistance        940 non-null    float64
 7   ModeratelyActiveDistance  940 non-null    float64
 8   LightActiveDistance       940 non-null    float64
 9   SedentaryActiveDistance   940 non-null    float64
 10  VeryActiveMinutes         940 non-null    int64  
 11  FairlyActiveMinutes       940 non-null    int64  
 12  LightlyActiveMinutes      940 non-null    int64  
 13  SedentaryMinutes          940 non-null    int64  
 14  Calories                  940 non-null    int64  
dtypes: float64(7), int64(7), object(1)
memory usage: 110.3+ KB
None
```

ANALYSING

- Our target column will be Activated, since this contains the first data we will analyze, so we need to convert its format to Datetime

```
[ ]  data['ActivityDate'] = pd.to_datetime(data['ActivityDate'],format='%m/%d/%Y')
     print(data.info())

     <class 'pandas.core.frame.DataFrame'>
     RangeIndex: 940 entries, 0 to 939
     Data columns (total 15 columns):
      #   Column                    Non-Null Count  Dtype
     ---  ------                    --------------  -----
      0   Id                        940 non-null    int64
      1   ActivityDate              940 non-null    datetime64[ns]
      2   TotalSteps                940 non-null    int64
      3   TotalDistance             940 non-null    float64
      4   TrackerDistance           940 non-null    float64
      5   LoggedActivitiesDistance  940 non-null    float64
      6   VeryActiveDistance        940 non-null    float64
      7   ModeratelyActiveDistance  940 non-null    float64
      8   LightActiveDistance       940 non-null    float64
      9   SedentaryActiveDistance   940 non-null    float64
      10  VeryActiveMinutes         940 non-null    int64
      11  FairlyActiveMinutes       940 non-null    int64
      12  LightlyActiveMinutes      940 non-null    int64
      13  SedentaryMinutes          940 non-null    int64
      14  Calories                  940 non-null    int64
     dtypes: datetime64[ns](1), float64(7), int64(7)
     memory usage: 110.3 KB
     None
```

WE PROCEED TO COMBINE THE INFORMATION FROM THE VERYACTIVEMINUTES, FAIRLYACTIVEMINUTES, LIGHTLYACTIVEMINUTES, SEDENTARYMINUTES COLUMNS OF THE DATASET IN ORDER TO OBTAIN A DATASET

```
[ ]  data['TotalMinutes'] = data['VeryActiveMinutes'] + data['FairlyActiveMinutes'] + data['LightlyActiveMinutes'] + data['SedentaryMinutes']
     print(data['TotalMinutes'].sample(5))

616     922
66     1440
222    1440
25     1091
508    1209
Name: TotalMinutes, dtype: int64
```

NOW LET'S HAVE A LOOK TO THE DESCRIPTIVE STATISTICS OF THE DATASET

```
[ ]  print(data.describe())
```

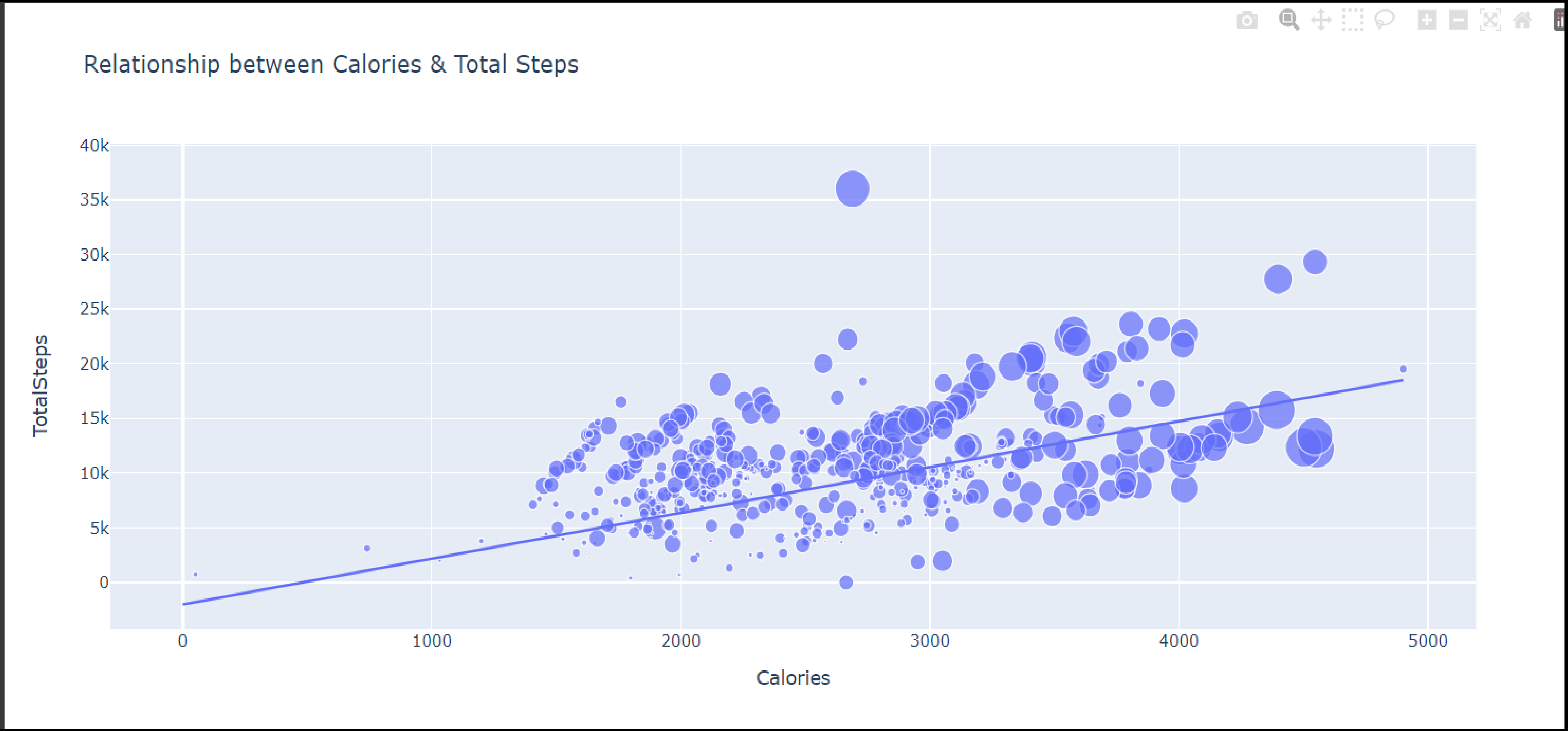
```
                 Id    TotalSteps  TotalDistance  TrackerDistance  \
count  9.400000e+02    940.000000     940.000000       940.000000   
mean   4.855407e+09   7637.910638       5.489702         5.475351   
std    2.424805e+09   5087.150742       3.924606         3.907276   
min    1.503960e+09      0.000000       0.000000         0.000000   
25%    2.320127e+09   3789.750000       2.620000         2.620000   
50%    4.445115e+09   7405.500000       5.245000         5.245000   
75%    6.962181e+09  10727.000000       7.712500         7.710000   
max    8.877689e+09  36019.000000      28.030001        28.030001   

       LoggedActivitiesDistance  VeryActiveDistance  ModeratelyActiveDistance  \
count                940.000000          940.000000                940.000000   
mean                   0.108171            1.502681                  0.567543   
std                    0.619897            2.658941                  0.883580   
min                    0.000000            0.000000                  0.000000   
25%                    0.000000            0.000000                  0.000000   
50%                    0.000000            0.210000                  0.240000   
75%                    0.000000            2.052500                  0.800000   
max                    4.942142           21.920000                  6.480000   
```

```
       LightActiveDistance  SedentaryActiveDistance  VeryActiveMinutes  \
count           940.000000               940.000000         940.000000
mean              3.340819                 0.001606          21.164894
std               2.040655                 0.007346          32.844803
min               0.000000                 0.000000           0.000000
25%               1.945000                 0.000000           0.000000
50%               3.365000                 0.000000           4.000000
75%               4.782500                 0.000000          32.000000
max              10.710000                 0.110000         210.000000

       FairlyActiveMinutes  LightlyActiveMinutes  SedentaryMinutes  \
count           940.000000            940.000000        940.000000
mean             13.564894            192.812766        991.210638
std              19.987404            109.174700        301.267437
min               0.000000              0.000000          0.000000
25%               0.000000            127.000000        729.750000
50%               6.000000            199.000000       1057.500000
75%              19.000000            264.000000       1229.500000
max             143.000000            518.000000       1440.000000

          Calories  TotalMinutes
count   940.000000    940.000000
mean   2303.609574   1218.753191
std     718.166862    265.931767
min       0.000000      2.000000
25%    1828.500000    989.750000
50%    2134.000000   1440.000000
75%    2793.250000   1440.000000
max    4900.000000   1440.000000
```

DATA VISULIZATION

THE DATA SET HAS A COLUMN OF CALORIES, WHICH CONTAINS THE NUMBER OF CALORIES BURNED PER DAY, LET'S SEE THE RELATIONSHIP BETWEEN THE NUMBER OF CALORIES BURNED AND THE TOTAL NUMBER OF STEPS PER DAY

```
fiugure = px.scatter(data_frame = data, x='Calories',
                     y='TotalSteps', size='VeryActiveMinutes',
                     trendline='ols',
                     title='Relationship between Calories & Total Steps')
fiugure.show()
```

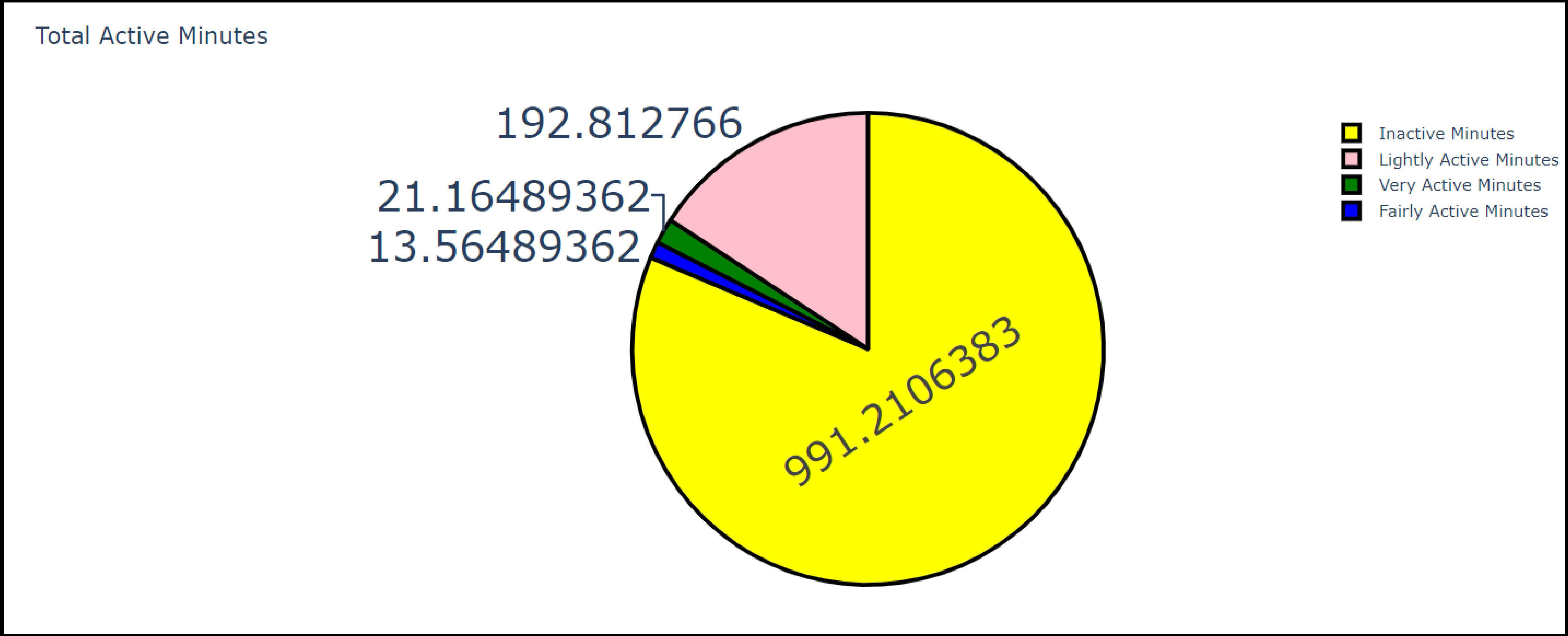


Relationship between Calories & Total Steps
TotalSteps
40k
35k
30k
25k
20k
15k
10k
5k
0
0
1000
2000
3000
4000
5000
Calories

NOW LET'S SEE AN AVERAGE BETWEEN THE TOTAL ACTIVE MINUTES PER DAY

```
[ ]  from turtle import width


     label = ["Very Active Minutes", "Fairly Active Minutes",
              "Lightly Active Minutes", "Inactive Minutes"]
     counts = data[["VeryActiveMinutes", "FairlyActiveMinutes",
                    "LightlyActiveMinutes", "SedentaryMinutes"]].mean()
     colors = ['green','blue', "pink", "yellow"]

     fig = go.Figure(data=[go.Pie(labels=label, values=counts)])
     fig.update_layout(title_text='Total Active Minutes')
     fig.update_traces(hoverinfo='label+percent', textinfo='value', textfont_size=30,
                       marker=dict(colors=colors, line=dict(color='black',width=3 )))
```

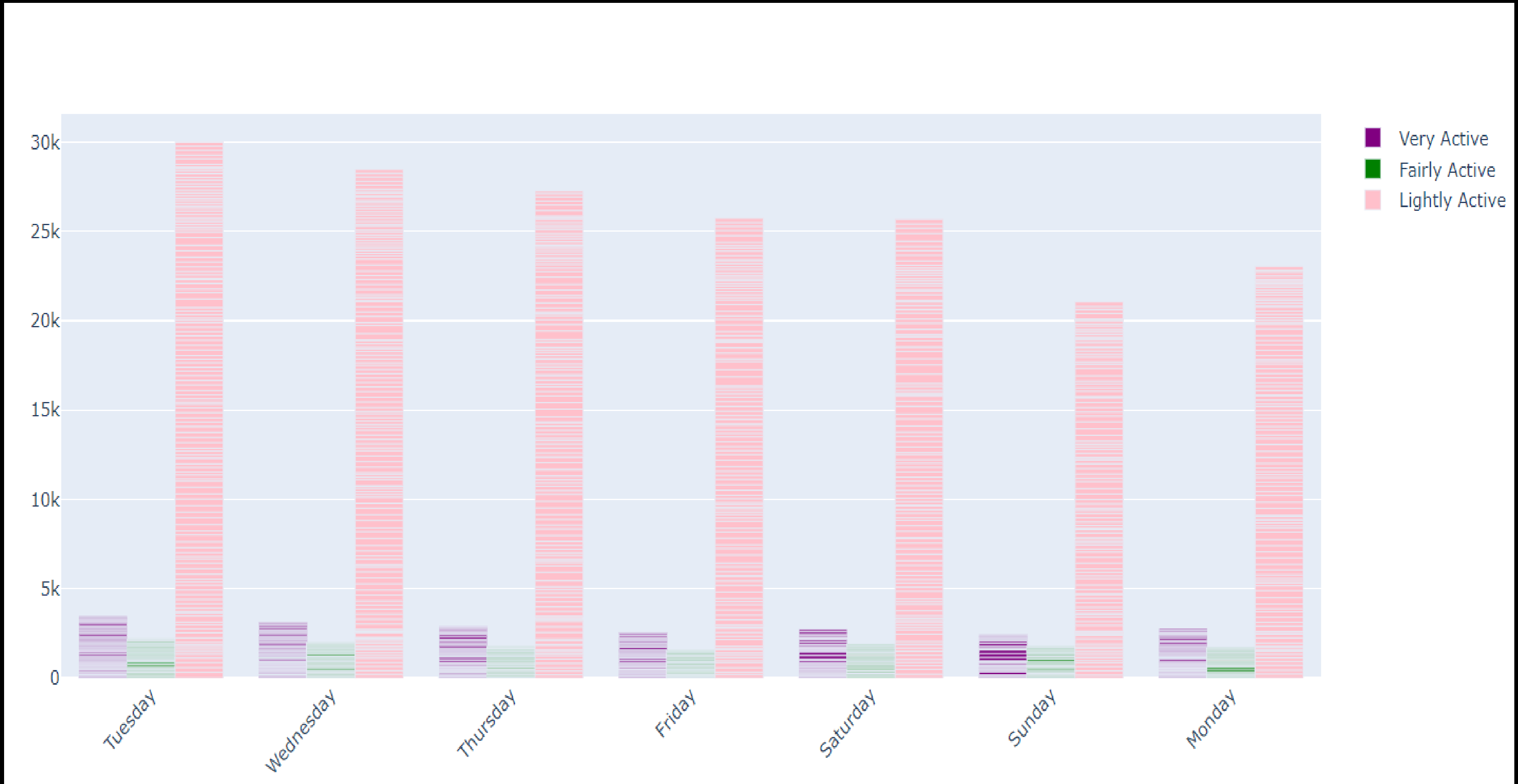
OUTPUT


Total Active Minutes
192.812766
21.16489362
13.56489362
991.2106383
Inactive Minutes
Lightly Active Minutes
Very Active Minutes
Fairly Active Minutes

LET'S SEE GRAPHICALLY THE DISTRIBUTION OF THE DAYS WITH ACTIVITY

```
[ ]  fig = go.Figure()
     fig.add_trace(go.Bar(
         x=data['Day'],
         y=data['VeryActiveMinutes'],
         name='Very Active',
         marker_color='purple'
     ))
     fig.add_trace(go.Bar(
         x=data['Day'],
         y=data['FairlyActiveMinutes'],
         name='Fairly Active',
         marker_color='green'
     ))
     fig.add_trace(go.Bar(
         x=data['Day'],
         y=data['LightlyActiveMinutes'],
         name='Lightly Active',
         marker_color='pink'
     ))
     fig.update_layout(barmode='group', xaxis_tickangle=-45)
     fig.show()
```

OUTPUT

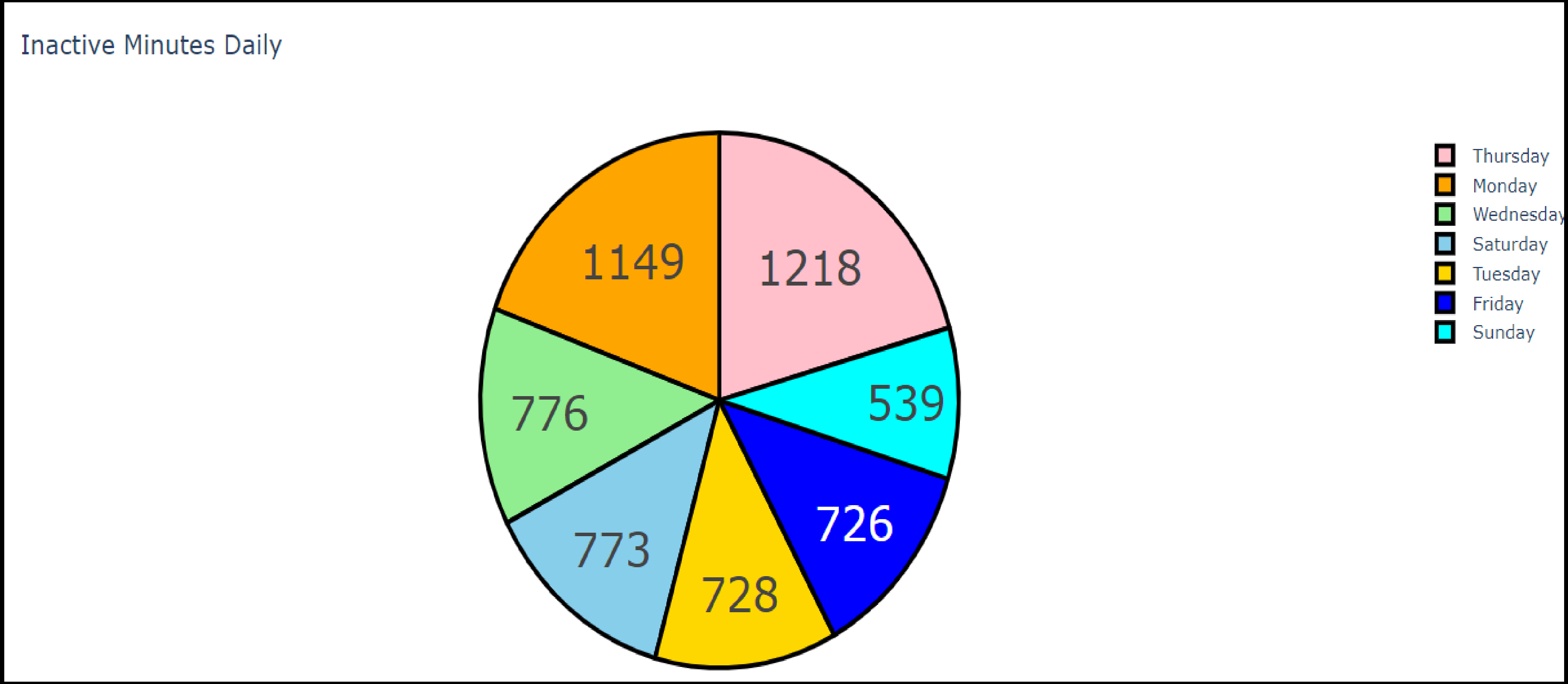


Very Active
Fairly Active
Lightly Active
30k
25k
20k
15k
10k
5k
0
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
Monday

NOW LET'S SEE THE NUMBER OF INACTIVE MINUTES ON EACH DAY OF THE WEEK

```
[ ]  day = data['Day'].value_counts()
     label = day.index
     counts = data['SedentaryMinutes']
     colors = ['gold','lightgreen','pink','blue','skyblue','cyan','orange']

     fig = go.Figure(data=[go.Pie(labels=label, values=counts)])
     fig.update_layout(title_text='Inactive Minutes Daily')
     fig.update_traces(hoverinfo='label+percent', textinfo='value', textfont_size=30,
                       marker=dict(colors=colors, line=dict(color='black', width=3)))
     fig.show()
```

OUTPUT

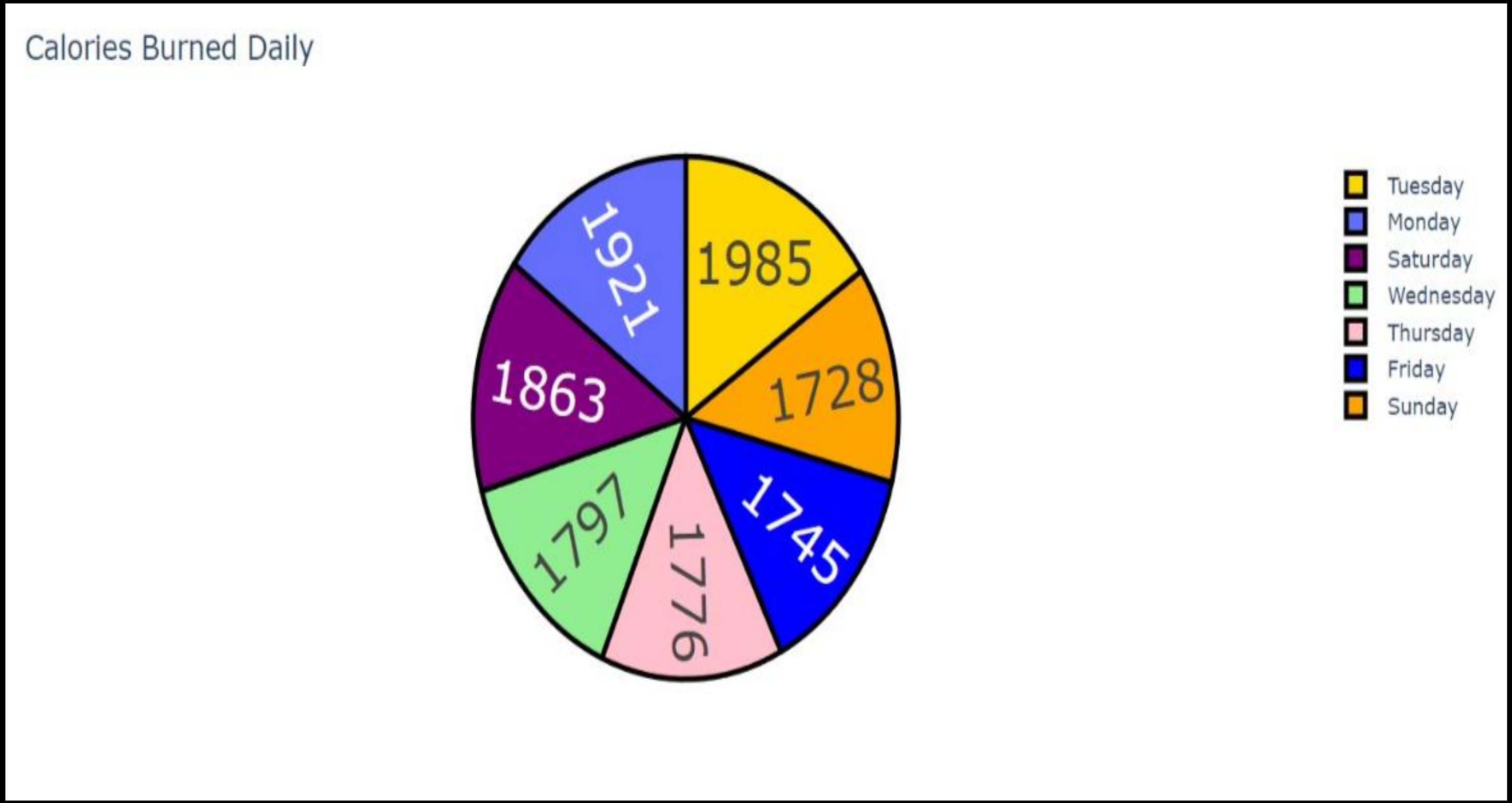


Inactive Minutes Daily
1149
1218
776
539
773
728
726
Thursday
Monday
Wednesday
Saturday
Tuesday
Friday
Sunday

WITH THE INFORMATION COLLECTED FROM SEVERAL PEOPLE WE CAN REACH THE CONCLUSION THAT ON THURSDAYS ARE THE LEAST PRODUCTIVE DAYS IN GENERAL NOW LET'S SEE THE NUMBER OF CALORIES BURNED FOR EACH DAY OF THE WEEK

```
calories = data['Day'].value_counts()
label = calories.index
counts = data['Calories']
colors = ['gold','lightgreen','pink','blue','purple','orange']

fig = go.Figure(data=[go.Pie(labels=label, values=counts)])
fig.update_layout(title_text='Calories Burned Daily')
fig.update_traces(hoverinfo='label+percent', textinfo='value', textfont_size=30,
                  marker=dict(colors=colors, line=dict(color='black', width=3)))
fig.show()
```

OUTPUT



Calories Burned Daily
1985
1921
1863
1728
1745
1776
1797
Tuesday
Monday
Saturday
Wednesday
Thursday
Friday
Sunday

CONCLUSION

- In conclusion, this analysis aims to shed light on the significance of caloric expenditure tracking through smartwatches, providing valuable insights for users seeking to enhance their health and fitness journeys. By exploring the technology behind these wearables, their applications, and their limitations, we hope to offer a comprehensive understanding of how smartwatches contribute to a healthier and more active lifestyle. As the smartwatch market continues to evolve, this study will serve as a foundation for future advancements in fitness technology, empowering users to make more informed decisions about their well-being.

PRESENTED BY:

21BCE9094
M.YATHEESWAR

21BCE9180
V.KALYAN HANUMAN CHOWDARY

21BCE8976
P.LEELA KRISHNA