BELLABEAT CASE STUDY

Introduction

What is a smart device and why do we need them? My understanding of a smart device is any tool (wired or wireless) that is capable of connecting to other devices and performs simple or complex operations. Imagine having to stand up, walk over to the T.V, and click a button several times just to change the channel? It might seem like an easy task, but imagine having to do that every single time. Worse part is, some channels are 100, 200, sometimes even a 1000 clicks away. If you love sports like me, and the hockey show is on channel 456, it would mean you have to click the forward button over 400 times to get a desired channel. Not only is this task strenuous on the fingers and legs, but boredom ensues quickly as it is a very monotonous task.

With a remote control - this tool allows the user to perform various operations on the T.V with a simple click of a button. In the case of our sports example earlier, all we have to do on the remote control is click numbers 4-5-6 on the button pad. You can do various other operations like record a show, change the volume settings, increase the aspect ratio, brightness, and a whole lot more. This tool makes using the T.V easy, and by principle is the definition of a smart device. There are various types of smart devices ranging from phones, laptops, appliances, etc. that make our lives easier by performing simple to complex operations for our benefit.

Some applications of smart devices can be seen in the health sector. Tools that help track patient data and use that information to administer treatment would be considered a smart device (*Science Soft*). Hospitals use ECG (electrocardiogram) to measure electrical activity in a patient's heart allowing for medical practitioners to diagnose heart conditions(*Massoomi MR, Handberg EM, 2019*).

Depending on the type of ECG, some can be prompted to alert your healthcare practitioner if irregularities occur in a patient's heart. Actions include sending alert messages to the clinic or automatically dialing 9-1-1 on a patient's phone. These actions make delivering health care more efficient for the healthcare provider and especially for the patient. This application can be expanded on to different aspects of a person's health. Accessories such as watches, rings, necklaces, and even water bottles have features that let them track heart rate, physical activity, sleep pattern, stress levels, and even hydration levels. For specific cases, there are watches that track sugar levels of persons dealing with diabetes. The application of smart devices is endless, and Bellabeat hopes to provide such services to current and new customers.

Bellabeat is a high-tech manufacturer of health-focused products geared towards women. Manufactured in 2013, Bellabeat has become a successful organization providing health & wellness smart products to its valued customers. Bellabeat products include an application that provides information about user health habits, various smart devices (tracker leaf, wellness watch, and water bottle) that track physical activity, sleep and stress levels, & hydration levels. Lastly, Bellabeat has a 24/7 membership that gives users health guidance/advice based on tracked data from smart devices, lifestyle and health goals. Although Bellabeat is a successful company, the COO and co-founder, Urška Sršen, would like the company to expand by increasing its market share in the smart device industry. As part of their marketing analyst department, I have been tasked to study and gain insight from external smart device usage data, and apply those insights into one of Bellabeats many health driven products. The insights I gain from this study will help guide marketing strategies in the future with the aim of expanding Bellabeat as a household name! Sršen believes analyzing customer data would yield more opportunities for growth, but my task is to gain more insight in smart device usage by analyzing external data sets.

Methods Section - Prepare

For this project, I have been asked to use the Fitbit business tracker data from Kaggle. This open source data set consists of personal fitness tracker data from 30 fitbit users. The fitbit users provided consent for this study, and were assigned randomized ID numbers to protect personally identifiable information (PII). This also helps mitigate any sampling or gender bias during data collection.

Personal health data was tracked by using Fitbits to measure minute-level output for physical activity, heart rate, and sleep monitoring. Information about daily activity, steps, and heart rate data are also included. Fitbits make for valid measures of health data as they are worn on the wrist to track physical changes. Data retrieved from these trackers are then used by users to set future health goals.

I retrieved and stored the dataset on a personal computer which is only accessible by me - the analyst tasked with this project. It will be stored for the duration of my analysis and then deleted a few days after my presentation. The dataset consisted of 18 CSV files - each file containing daily,hourly, and minute health data of each participant. CSV file format is the standard format used by majority analysis tools/softwares like Excel, SQL, and Tableau. Long data format was used in all 18 files and each column could be grouped into 3 main categories - Identification column contains assigned ID numbers for each participant; Timestamp column contains day/month/year information for each tracker entry, and the description columns create context for each health data tracked.

Morbius, a Kaggle user, sourced this dataset from a study completed by ***Furberg, Robert; Brinton, Julia; Keating, Michael ; Ortiz, Alexa*** *on Zenodo.org*. From March 12 to May 12 2016, participants responded to an open survey by providing their Fitbit tracker data. The dataset was made public through his account and has been cited by multiple users. The dataset has a Kaggle usability score of 100% as it shows excellent completeness (No null values), credibility, and compatability. The dataset is licensed under CCO:public domain making it accessible to the public. Since its inception, Morbius updates the dataset yearly, as the number of participants have increased by 3 - 33 unique ID's.
3 files were too large to open with MS Excel, so these datasets will be cleaned and processed using SQL. These CSV files tracked health data in seconds that exceeded row lengths of over 1 million entries!

I tested for integrity by ensuring participants were equally represebted (sampling bias), ensuring privacy, and following ethics rules when dealing with consent. I also followed the data lifetime cycle - retrieval, storage, use, and disposal. The validity of the dataset as well as the credibility of the dataset - Kaggle is a trusted site for data analysts. The dataset was recorded in 2019, has been cited by 3 peer reviewed papers, retrieved directly from participants, reliable(accurate, not missing any values, and consistent - no skewed values). ROCCC.

One of the major issues with our dataset is the sample size. 33 participants gave consent for this dataset, but this number of participants is not a significant representation of the population. To mitigate population bias, Urška Sršen advised me to seek other datasets to help supplement this study. Although I found a few datasets capable of supplementing this study, most were behind a paywall. I was able to find one public dataset from The Arctic university of Norway. Henrikson et al (2021) collected physical activity data using fitness trackers from 2019 to 2020. This dataset was intended to help in further epidemiology study during Covid-19. 113 participants in Norway consented to this study where their daily steps and versions of physical activity (PA) was tracked.

The dataset can definitely help improve results in this study, but like most datasets, it does have its flaws. Participants in this dataset were collected in Norway, so results will be biased towards this location/region of the world. Null values were presented in this dataset, but the authors defined these as either not wearing the trackers during these days (i) or not owning a tracker during these days(ii) or participants that wear their Fitbit/Garmin for more than 10h and did not clock over 150 steps (iii).

By combining both dataset, we can reduce population bias in study and get a more representative finding in this study.

Data cleaning - Change log

Fitbit dataset :

- By combining both dataset, we can reduce population bias in study and get a more representative finding in this study.

- For both datasets, I changed the name (included units of measurements, acronyms, and naming schema) and data type (date, time, numerical, string) of each column in Excel.

- The fitbit dataset and raw Norway dataset were in long format, whereas the **analyzed** Norway dataset was in wide format. To ensure congruence, I converted the wide dataset from wide to long.

- I started this process by using the function: `=ARRAYFORMULA(SPLIT(FLATTEN(data.csv!A2:A114&"🦊"&data.csv!B2:B114&"🦊"&data.csv!C1:HR1&"🦊"&data.csv!C2:HR114),"🦊"))` function to convert the wide data into a long data format.

- I filtered-in data that ONLY included the average number of steps each participant took in a month. Measurements of physical activity like total energy expenditure (TEE), activity energy expenditure (AEE), moderate-to-vigorous physical activity (MVPA), light PA (LPA), moderate PA (MPA), vigorous PA (VPA), and sedentary time were removed from this dataset.

- I then used the `Find & replace` tool to remove “-Steps” from each row in the types of steps/physical activity column. The purpose was to create a new column - monthly_date - that only included date data.

- NULL values in the average daily steps column describe participants that did not wear their Fitbit/Garmin for more than 10h and did not clock over 150 steps.

- This Revised version of the Norway dataset includes columns for ID, monthly date, average daily steps each month, and the kind of provider (Fitbit/Garmin).

- Each participant tracked their average daily steps. This column includes data about the following - average daily steps during each month, average daily steps during each year, and between March 1st-12th, and March 13th to 31st (pre- and post COVID-19 lockdown in Norway).

Norway raw dataset :

- The **raw** dataset from the Norway study contains daily step activity presented in long format. I used the `Find & replace` tool on the date column to replace the delimiter " -" with " / ".

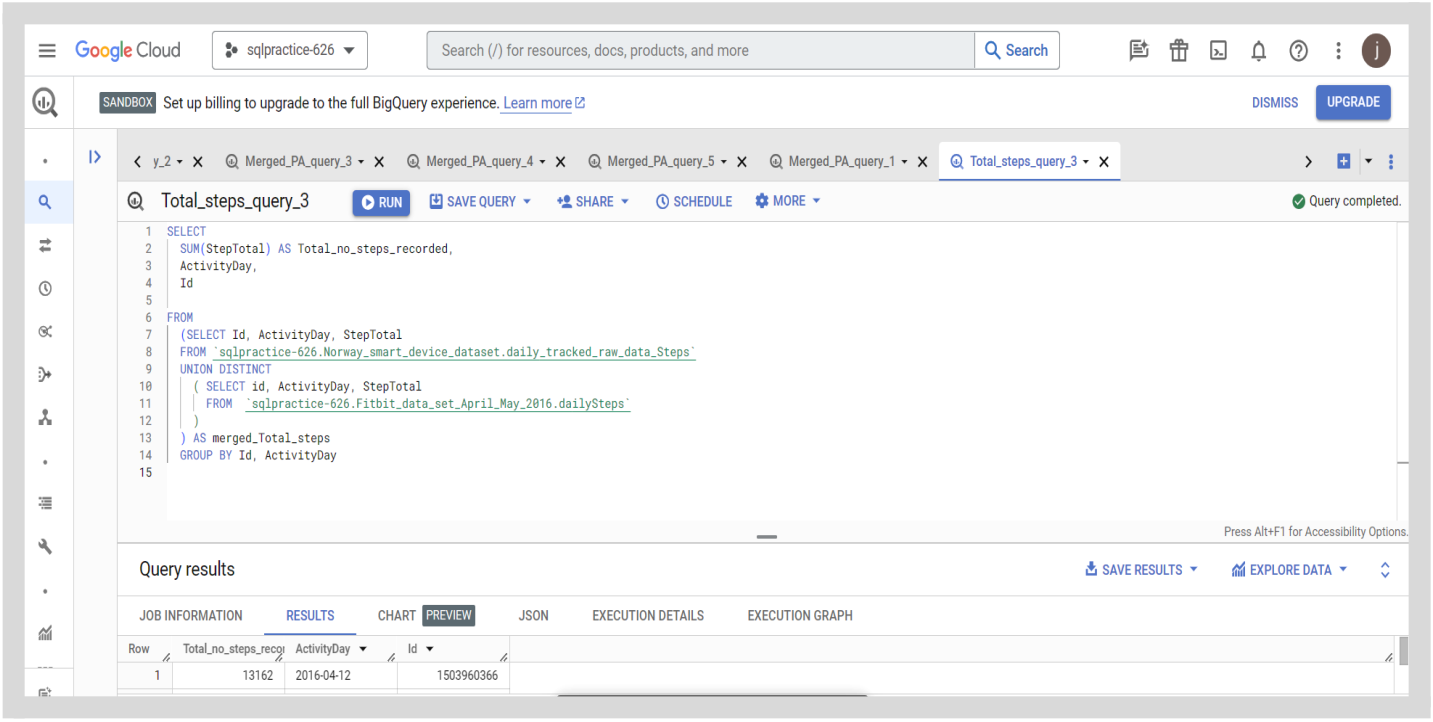
- " / " is the ideal delimiter for formatting date data. I.e. *" 12/25/2023 "*

- Although our data was complete and there are no missing values, certain participants failed to log in tracked fitness data. Some participants logged data daily from January 1st 2019 to December 31st, 2023, while others missed days and months of log entries. ***(Limitation to our dataset)***

- I revised versions of the Norway raw dataset into 2:
  1. This excel sheet contains participants IDs, provider (Fitbit or Garmin), and total number of steps on each day.
  2. The excel sheet contains columns for participant IDs, providers (Fitbit or Garmin), Sedimentary time (i.e sitting, idle.) measured in minutes, 3 categories of physical activities measured in minutes, and calorie expenditure measured in Kcal (kilocalories).

- For this analysis, I will be using SQL to identify trends in smart device usage amongst users.
- Tableau is my tool of choice for visualization.

Analysis



- After importing both datasets (Fitbit & Norway) into SQL, I can now run simple analysis on my dataset.
- I used the UNION DISTINCT function on SQL to combine the Fitbit dataset and Norway dataset together. This merged both datasets, making sure duplicates were removed from the final result.

Total_steps_query_5 RUN SAVE QUERY SHARE SCHEDULE MORE

```sql
--declare variable
DECLARE
  mean_total_step_count FLOAT64;
--set variable value
SET
  mean_total_step_count = 9602.02;



------ 3rd step data visualization table -----------
-----------------------------------------------
SELECT
    Id,

  AVG(StepTotal) AS Average_Steps_per_Id,

  CASE
    WHEN AVG(StepTotal) > mean_total_step_count OR AVG(StepTotal) = mean_total_step_count THEN "Above_AVG_step_count"
    WHEN AVG(StepTotal) < mean_total_step_count THEN "Less_than_AVG_step_count"
  ELSE NULL
  END AS Average_step_summary,

  MIN(StepTotal) AS min_step_count

 FROM
    (SELECT Id, ActivityDay, StepTotal
    FROM `sqlpractice-626.Norway_smart_device_dataset.daily_tracked_raw_data_Steps`
    UNION DISTINCT
      ( SELECT id, ActivityDay, StepTotal
      FROM  `sqlpractice-626.Fitbit_data_set_April_May_2016.dailySteps`
      )
    ) AS merged_Total_steps

GROUP BY Id
```

Total_steps_query_4 RUN SAVE QUERY SHARE SCHEDULE MORE

```sql
SELECT
  AVG(StepTotal) AS Average_no_Steps_recorded,
  SUM(StepTotal) AS Total_no_steps_recorded,
  MAX(StepTotal) AS highest_step_count_day,

FROM
  (SELECT Id, ActivityDay, StepTotal
  FROM `sqlpractice-626.Norway_smart_device_dataset.daily_tracked_raw_data_Steps`
  UNION DISTINCT
    ( SELECT id, ActivityDay, StepTotal
    FROM  `sqlpractice-626.Fitbit_data_set_April_May_2016.dailySteps`
    )
  ) AS merged_Total_steps

-- Average number of steps, total steps recorded and highest ever step count in 1 day.
```

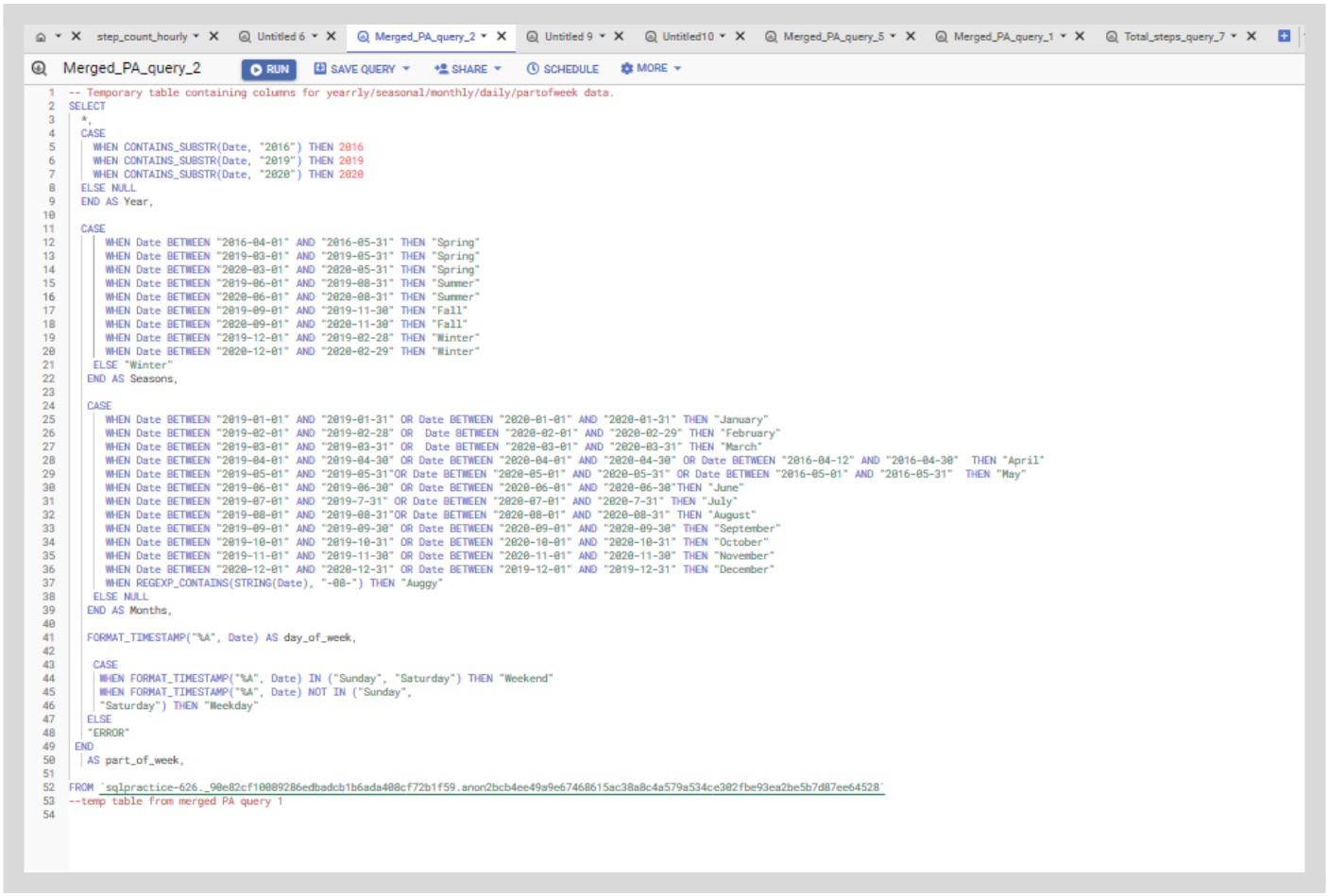
Press Alt+F1 for Accessibility Opt

Query results SAVE RESULTS EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Average_no_Steps_recorded	Total_no_steps_recorded	highest_step_count_day
1	9602.2007376834244	700298102	128172

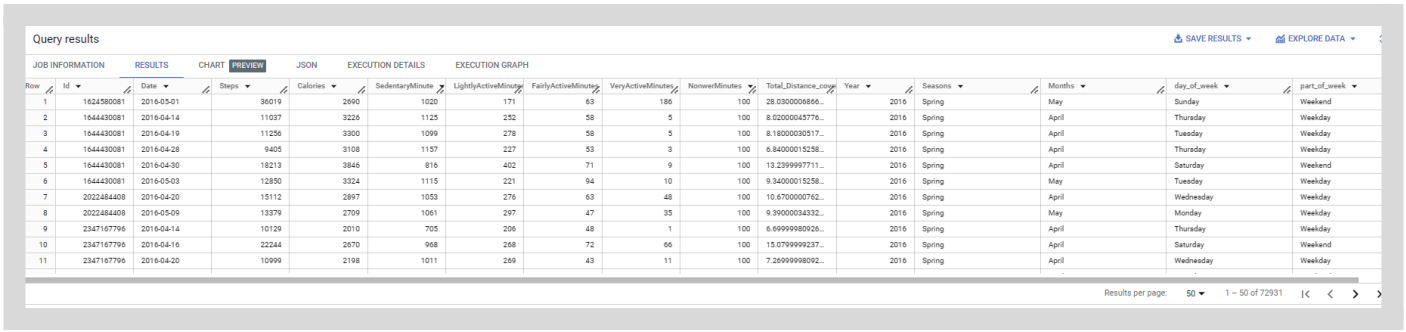
- This function was applied to the total steps table, PA table, and Intensities table.
- Summary data shows 700,298,102 total steps were recorded, with a total average of 9,602.20 steps.
- Out of 146 participants, 145 participants recorded over 5000 steps daily.
- The highest Total step count in 1 day was 128,172 and it was recorded on the 19th of September 2019.
- 61 out of 146 participants recorded average step counts higher than the mean average step count for all participants (9,602.02). **41.8%** participants recorded higher average steps than the overall average for all steps among all paticipants.



Merged_PA_query_2

RUN | SAVE QUERY | SHARE | SCHEDULE | MORE

```
-- Temporary table containing columns for yearrly/seasonal/monthly/daily/partofweek data.
SELECT
  *,
  CASE
    WHEN CONTAINS_SUBSTR(Date, "2016") THEN 2016
    WHEN CONTAINS_SUBSTR(Date, "2019") THEN 2019
    WHEN CONTAINS_SUBSTR(Date, "2020") THEN 2020
  ELSE NULL
  END AS Year,

  CASE
      WHEN Date BETWEEN "2016-04-01" AND "2016-05-31" THEN "Spring"
      WHEN Date BETWEEN "2019-03-01" AND "2019-05-31" THEN "Spring"
      WHEN Date BETWEEN "2020-03-01" AND "2020-05-31" THEN "Spring"
      WHEN Date BETWEEN "2019-06-01" AND "2019-08-31" THEN "Summer"
      WHEN Date BETWEEN "2020-06-01" AND "2020-08-31" THEN "Summer"
      WHEN Date BETWEEN "2019-09-01" AND "2019-11-30" THEN "Fall"
      WHEN Date BETWEEN "2020-09-01" AND "2020-11-30" THEN "Fall"
      WHEN Date BETWEEN "2019-12-01" AND "2019-02-28" THEN "Winter"
      WHEN Date BETWEEN "2020-12-01" AND "2020-02-29" THEN "Winter"
    ELSE "Winter"
  END AS Seasons,

  CASE
      WHEN Date BETWEEN "2019-01-01" AND "2019-01-31" OR Date BETWEEN "2020-01-01" AND "2020-01-31" THEN "January"
      WHEN Date BETWEEN "2019-02-01" AND "2019-02-28" OR  Date BETWEEN "2020-02-01" AND "2020-02-29" THEN "February"
      WHEN Date BETWEEN "2019-03-01" AND "2019-03-31" OR  Date BETWEEN "2020-03-01" AND "2020-03-31" THEN "March"
      WHEN Date BETWEEN "2019-04-01" AND "2019-04-30" OR Date BETWEEN "2020-04-01" AND "2020-04-30" OR Date BETWEEN "2016-04-12" AND "2016-04-30"  THEN "April"
      WHEN Date BETWEEN "2019-05-01" AND "2019-05-31"OR Date BETWEEN "2020-05-01" AND "2020-05-31" OR Date BETWEEN "2016-05-01" AND "2016-05-31"  THEN "May"
      WHEN Date BETWEEN "2019-06-01" AND "2019-06-30" OR Date BETWEEN "2020-06-01" AND "2020-06-30"THEN "June"
      WHEN Date BETWEEN "2019-07-01" AND "2019-7-31" OR Date BETWEEN "2020-07-01" AND "2020-7-31" THEN "July"
      WHEN Date BETWEEN "2019-08-01" AND "2019-08-31"OR Date BETWEEN "2020-08-01" AND "2020-08-31" THEN "August"
      WHEN Date BETWEEN "2019-09-01" AND "2019-09-30" OR Date BETWEEN "2020-09-01" AND "2020-09-30" THEN "September"
      WHEN Date BETWEEN "2019-10-01" AND "2019-10-31" OR Date BETWEEN "2020-10-01" AND "2020-10-31" THEN "October"
      WHEN Date BETWEEN "2019-11-01" AND "2019-11-30" OR Date BETWEEN "2020-11-01" AND "2020-11-30" THEN "November"
      WHEN Date BETWEEN "2020-12-01" AND "2020-12-31" OR Date BETWEEN "2019-12-01" AND "2019-12-31" THEN "December"
      WHEN REGEXP_CONTAINS(STRING(Date), "-08-") THEN "Auggy"
    ELSE NULL
  END AS Months,

  FORMAT_TIMESTAMP("%A", Date) AS day_of_week,

   CASE
    WHEN FORMAT_TIMESTAMP("%A", Date) IN ("Sunday", "Saturday") THEN "Weekend"
    WHEN FORMAT_TIMESTAMP("%A", Date) NOT IN ("Sunday",
    "Saturday") THEN "Weekday"
  ELSE
   "ERROR"
END
  AS part_of_week,

FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anon2bcb4ee49a9e67468615ac38a8c4a579a534ce302fbe93ea2be5b7d87ee64528`
--temp table from merged PA query 1
```



Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Id	Date	Steps	Calories	SedentaryMinute	LightlyActiveMinute	FairlyActiveMinutes	VeryActiveMinutes	NonwerMinutes	Total_Distance_cor	Year	Seasons	Months	day_of_week	part_of_week
1	1624580081	2016-05-01	36019	2690	1020	171	63	186	100	28.0300006866...	2016	Spring	May	Sunday	Weekend
2	1644430081	2016-04-14	11037	3226	1125	252	58	5	100	8.02000045776...	2016	Spring	April	Thursday	Weekday
3	1644430081	2016-04-19	11256	3300	1099	278	58	5	100	8.18000030517...	2016	Spring	April	Tuesday	Weekday
4	1644430081	2016-04-28	9405	3108	1157	227	53	3	100	6.84000015258...	2016	Spring	April	Thursday	Weekday
5	1644430081	2016-04-30	18213	3846	816	402	71	9	100	13.2399997711...	2016	Spring	April	Saturday	Weekend
6	1644430081	2016-05-03	12850	3324	1115	221	94	10	100	9.34000015258...	2016	Spring	May	Tuesday	Weekday
7	2022484408	2016-04-20	15112	2897	1053	276	63	48	100	10.6700000762...	2016	Spring	April	Wednesday	Weekday
8	2022484408	2016-05-09	13379	2709	1061	297	47	35	100	9.39000034332...	2016	Spring	May	Monday	Weekday
9	2347167796	2016-04-14	10129	2010	705	206	48	1	100	6.69999980926...	2016	Spring	April	Thursday	Weekday
10	2347167796	2016-04-16	22244	2670	968	268	72	66	100	15.0799999237...	2016	Spring	April	Saturday	Weekend
11	2347167796	2016-04-20	10999	2198	1011	269	43	11	100	7.26999998092...	2016	Spring	April	Wednesday	Weekday

Results per page: 50 | 1 – 50 of 72931

- Agregate the data into seasonal/monthly/daily/part of week information.

STEPS AND CALORIES

Seasonal data:

```sql
--Temp table from merged PA query 2 (`sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`)

SELECT
  Seasons,
  MAX(SedentaryMinute) AS max_rest_time,
  AVG(SedentaryMinute) AS avg_rest_time,
  MIN(SedentaryMinute) AS min_rest_time,
  AVG(Steps) AS avg_step_count,
  SUM(Steps) AS total_step_count,
  MAX(Calories) AS max_calory_exp,
  MIN(Calories) AS min_Calory_exp,
  AVG(Calories) AS average_calories_season,
  SUM(Calories) AS Total_calories_season,
  corr(steps, Calories) AS correlation_bet_steps_calories
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY Seasons;

--- Steps, Calories per season
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Seasons	max_rest_time	avg_rest_time	min_rest_time	avg_step_count	total_step_count	max_calory_exp	min_Calory_exp	average_calories_season	Total_calories_season	correlation_bet_steps_calories
1	Fall	1440	251.2169129371...	0	9776.710268948...	183939027	28379	0	3219.9582757521052	60580295	0.678512423919531 26
2	Spring	1440	290.3162167975...	0	9308.657973975...	173122421	12300	0	3169.5733412194927	58947725	0.661039084670877 91
3	Summer	1440	239.6679027156...	0	10405.88412040...	190823103	16096	0	3299.6380739448186	60508763	0.673464699410950 66
4	Winter	1440	251.4317560095...	0	8871.052383446...	152413551	12512	0	3135.6106163785594	53872926	0.682475502820202 08

- The highest average of sedentary time/rest time occurred during the spring at 290.33 (04 hrs:50 minutes:18 seconds). Summer was the least with 239.667 (03:49:40), followed by winter(04:11:25) and fall (04:11:12) respectively. We can infer participants were least idle during the summer season.
- Participants averaged 10,405.88 daily steps during the summer, the highest amongst all 4 seasons. Fall season had the 2nd highest with 9,776.71 average daily steps, with spring(9,308.65) and winter(8,871.05) following up respectively. The weather during the winter season may play a role into why average daily step counts were low compared to other seasons.
- During the summer, participants on average lost 3,299.63 kilocalories a day - the highest out of all 4 seasons.Fall came in 2nd with an average of 3,219.95 kilocalories burned a day. Participants exhausted the 3rd highest average in spring (3,169.57 kilocalories), while winter experienced the least energy expenditure with 3,135.61 kilocalories.
- We compared average step count and average calories count and found a positive correlation amongst all 4 seasons. Although the average correlation coefficient was low at 0.6735, we can infer a positive relation between calories burned and step count.

Monthly:

```sql
SELECT
  Months,
  MAX(SedentaryMinute) AS max_rest_time,
  MIN(SedentaryMinute) AS min_rest_time,
  AVG(SedentaryMinute) AS avg_rest_time,
  AVG(Steps) AS avg_step_count,
  SUM(Steps) AS total_step_count,
  MAX(Calories) AS max_calory_exp,
  MIN(Calories) AS min_Calory_exp,
  AVG(Calories) AS average_calories_month,
  SUM(Calories) AS Total_calories_month,
  corr(steps, Calories) AS correlation_bet_steps_calories,
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY Months;
-- Steps, Calories Per month

```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Months	max_rest_time	min_rest_time	avg_rest_time	avg_step_count	total_step_count	max_calory_exp	min_Calory_exp	average_calories_mo	Total_calories_month	correlation_bet_steps_calories
1	May	1440	0	281.93382584093...	9690.524206224...	61651115	12300	0	3233.861678717...	20573828	0.66021054439642224
2	April	1440	0	325.46804481615...	9221.665456840...	58437694	10301	0	3110.907369417...	19713820	0.65086580045686548
3	September	1440	0	247.64950462887...	10066.65242813...	61980379	28379	0	3246.008445671...	19985674	0.694477165368765
4	October	1440	0	253.68991885143...	9948.8061797753	63751950	18856	0	3265.747659176...	20926911	0.66810583218239228
5	November	1440	0	252.19587133941...	9314.562009921...	58206698	13954	0	3147.337173947...	19667710	0.67042652399656733
6	December	1440	0	251.49533780467...	9033.568951415...	55222207	9265	0	3123.910191395...	19096463	0.65791030688689767
7	January	1440	0	251.43026086956...	8582.080695652...	49346964	12512	0	3104.542086956...	17851117	0.6954823954954823
8	February	1440	0	251.36028582173...	8996.686724332...	47844380	9349	0	3182.652500940...	16925346	0.7003757038824595
9	March	1440	0	261.5946770639087	8990.271571452...	53033612	9082	0	3163.261061196...	18660077	0.6746484923653510
10	June	1440	0	233.66099983167...	10223.76822083...	60739407	16096	0	3267.987039218...	19415111	0.6676061758820746
11	July	1440	0	239.85413631212...	10729.33300955...	66275090	10709	0	3326.296422211...	20546533	0.6865563484610093
12	August	1440	0	245.22041800643...	10258.61832797...	63808606	14900	0	3303.395337620...	20547119	0.6658837398976843

- April, May, and March (Spring season) recorded on average the highest sedentary time/rest time amongst all months with 05:25:27 per day, 04:41:55 per day, and 04:21:35 per day.
  The least were the summer months - August(04:05:13), July(03:59:51), and June(03:53:39) . Comparing the highest and lowest rest time, participants spent a considerable amount of time being idle, sitting, or resting in spring months.
- The summer months - June, July, and August recorded the highest average step count per day. July was the highest with 10,729.33 steps, followed by August (10,258.61) and June (10,223.76). Winter months - January and February - were the least at 8,582.08 & 8,996.68 respectively. March - Spring season- was also amongst the lowest with 8,990.27 steps on average.
- July, August, and June -Summer months - were periods in which participants lost on average the highest kilocalories - 3,326.29kcal, 3,303.39kcal, and 32,67.98kcal.The least was January with 3,104.54 kilocalories per day.
- Although the summer months recorded the highest calories lost on average, October - fall season - recorded the highest total calories lost with 20,926,911 kilocalories.
- Similar to seasonal data, our monthly query shows there was a positive correlation between steps and calories expended.

Daily:

```sql
SELECT
  day_of_week,
  MAX(SedentaryMinute) AS max_rest_time,
  MIN(SedentaryMinute) AS min_rest_time,
  AVG(SedentaryMinute) AS avg_rest_time,
  AVG(Steps) AS avg_step_count,
  SUM(Steps) AS total_step_count,
  MAX(Calories) AS max_calory_exp,
  MIN(Calories) AS min_Calory_exp,
  AVG(Calories) AS average_calories_day,
  SUM(Calories) AS Total_calories_day,
  corr(steps, Calories) AS correlation_bet_steps_calories,
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY day_of_week;
-- Steps, Calories Per Day

```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	day_of_week	max_rest_time	min_rest_time	avg_rest_time	avg_step_count	total_step_count	max_calory_exp	min_Calory_exp	average_calories_day	Total_calories_day	correlation_bet_steps_calories
1	Sunday	1440	0	261.4437361151...	9246.497730126...	95728991	13954	0	3162.8707620979203	32745201	0.685624015128978
2	Thursday	1440	0	260.3960698273...	9665.155776018...	101319828	10431	0	3204.6160450252828	33593990	0.6605614068717687
3	Tuesday	1440	0	259.4118037764...	9488.5080178385	99999386	10917	0	3189.5516652433803	33614685	0.6621589026586240 9
4	Saturday	1440	0	247.6857335776...	10261.79338284...	106384012	28379	0	3337.3498601331207	34598306	0.6843051904480528 2
5	Wednesday	1440	0	261.5266679411...	9563.468361689...	100053006	10435	0	3195.6974765819127	33433387	0.6620176908431737 5
6	Monday	1440	0	259.2650706662...	9717.905009133...	101075930	12300	0	3205.0330737429126	33335549	0.6745126390241673 6
7	Friday	1440	0	258.5424171993...	9271.445767964...	95736949	13518	0	3155.9743366260022	32588591	0.6761445126846257 2

- **Wednesdays** recorded on average the highest sedentary/rest time amongst all days with (04:21:31) per day. The lowest average sedentary time was on **Saturdays** with 04:07:40 per day. Sundays were the 2nd highest on average with 04:21:26 per day, while Fridays were 1st from bottom with 04:21:26 per day.
- On **Saturdays**, participants recorded on average the highest step count with 10,261.79 steps/day. The least overall was **Sundays** with an average of 9,246.49 steps/day. During the weekdays, participants recorded the highest average step count with 9,717.90 steps/day on Mondays, while Fridays were the least with 9,271.44 steps/day.
- The highest average calorie expenditure was during **Saturdays** with 3337.34 kilocalories, while **Fridays** were the least with an average of 3,155.97 kilocalories.**Mondays** were the second highest on average with 3,205.03 kilocalories, and Sundays were 1st from bottom with 3,162.87 kilocalories.
- Factoring in days of the week, there was a positive correlation between steps and calories expended.

Part of the week:

```sql
SELECT
  part_of_week,
  MAX(SedentaryMinute) AS max_rest_time,
  MIN(SedentaryMinute) AS min_rest_time,
  AVG(SedentaryMinute) AS avg_rest_time,
  AVG(Steps) AS avg_step_count,
  SUM(Steps) AS total_step_count,
  MAX(Calories) AS max_calory_exp,
  MIN(Calories) AS min_Calory_exp,
  AVG(Calories) AS average_calories_week,
  SUM(Calories) AS Total_calories_week,
  corr(steps, Calories) AS correlation_bet_steps_calories,
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY part_of_week;
-- Steps, Calories per Part of the week
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	part_of_week	max_rest_time	min_rest_time	avg_rest_time	avg_step_count	total_step_count	max_calory_exp	min_Calory_exp	average_calories_week	Total_calories_week	correlation_bet_steps_calories
1	Weekend	1440	0	254.5600868725...	9754.48856177...	202113003	28379	0	3250.1692567567579	67343507	0.6861374131355551
2	Weekday	1440	0	259.8320277336...	9541.765126122...	498185099	13518	0	3190.2511348183116	166566202	0.6671054290058265 9

- Weekdays recorded the highest sedentary time/rest time with 04:19:49 per day, and weekends were the lowest with 04:14:33 per day. The difference is negligible.
- Participants were recorded taking more steps and burning more calories during the weekend.
- Although weekdays recorded the highest total step count with 498,185,099 steps, there are more weekdays (5) than there are weekends (2). The averages will give a better depiction of activity.
- Weekends (Saturday and Sundays) recorded the highest average step count with 9,754.48 steps. Weekdays had less with 9,541.76 steps/day.
- During the weekends, participants recorded on average 3,250.16 kilocalories burned a day - highest,, the weekdays came in 2nd with 3,190.25 kilocalories.
- Factoring in parts of the week, there was a positive correlation between steps and calories expended.

Time of day/ Hour of day:

```sql
SELECT
    Id,
    Calories,
    ActivityHour,
   CASE
      WHEN ActivityHour BETWEEN "2016-04-01" AND "2016-05-31" THEN "Spring"
    ELSE NULL
   END AS Seasons,

   CASE
      WHEN ActivityHour BETWEEN "2016-04-12" AND "2016-04-30"  THEN "April"
      WHEN ActivityHour BETWEEN "2016-05-01" AND "2016-05-31"  THEN "May"
      WHEN REGEXP_CONTAINS(STRING(ActivityHour), "-08-") THEN "Auggy"
    ELSE NULL
   END AS Months,

   FORMAT_TIMESTAMP("%A", ActivityHour) AS day_of_week,

    CASE
     WHEN FORMAT_TIMESTAMP("%A", ActivityHour) IN ("Sunday", "Saturday") THEN "Weekend"
     WHEN FORMAT_TIMESTAMP("%A", ActivityHour) NOT IN ("Sunday",
     "Saturday") THEN "Weekday"
   ELSE
   "ERROR"
 END
   AS part_of_week,

  CASE
     WHEN TIME(ActivityHour) BETWEEN TIME(6, 0, 0) AND TIME(12, 0, 0) THEN "Morning"
     WHEN TIME(ActivityHour) BETWEEN TIME(12, 0, 0) AND TIME(18, 0, 0) THEN "Afternoon"
     WHEN TIME(ActivityHour) BETWEEN TIME(18, 0, 0) AND TIME(21, 0, 0) THEN "Evening"
     WHEN TIME(ActivityHour) >= TIME(21,0,0) OR TIME(TIMESTAMP_TRUNC(ActivityHour, MINUTE)) <= TIME(6,0,0) THEN "Night"
    ELSE NULL
  END AS time_of_day,
 FROM `sqlpractice-626.Fitbit_data_set_April_May_2016.hourly_Calories`
 GROUP BY Id,ActivityHour, Calories
```

```sql
SELECT
  SUM(Calories) as Total_Calories_expenditure,
  AVG(Calories) AS Avg_Calories_expenditure_hour,
  time_of_day
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anon4752b4bd752cf2ad58a23329af932cf4375b30106151b425ef3a4a4fca113b10`
GROUP BY  time_of_day


----Time of Day - version 3
```

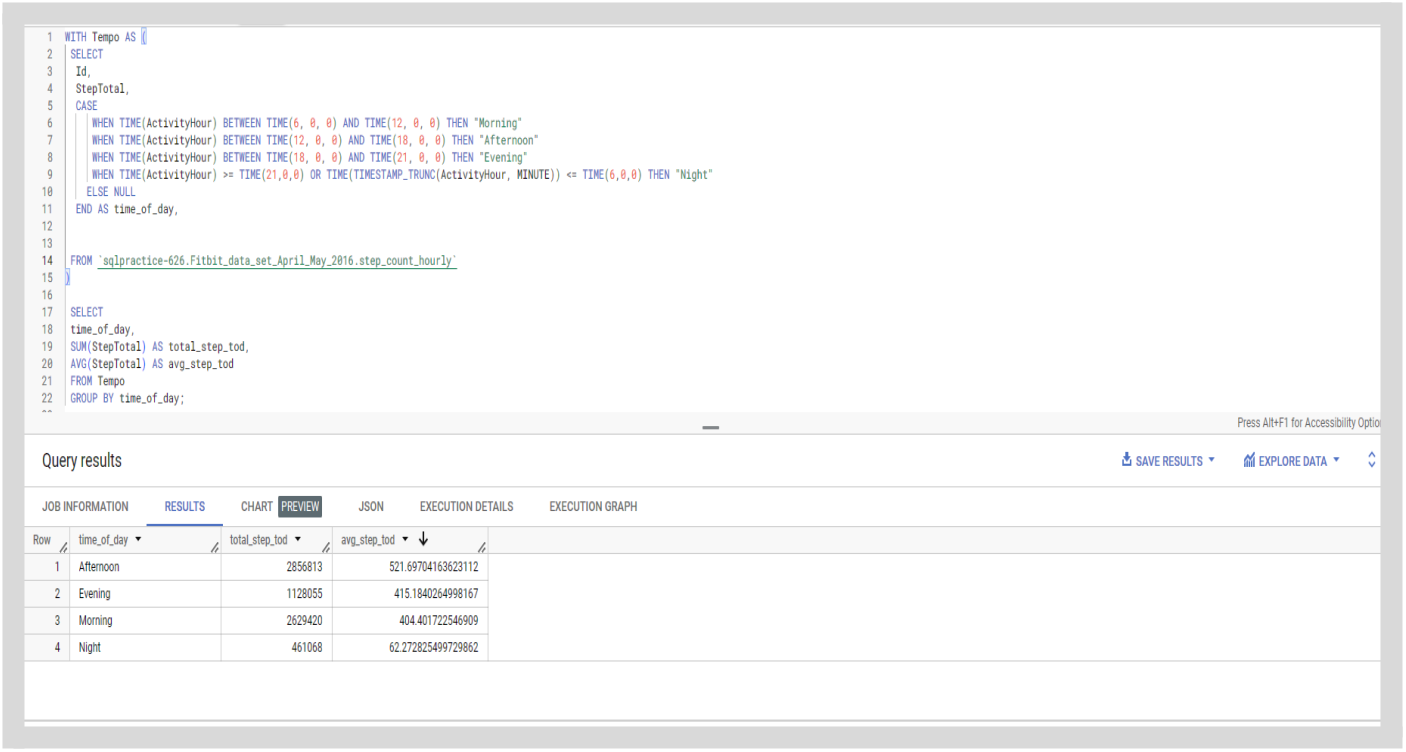
Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Total_Calories_exper	Avg_Calories_expenditure_hour	time_of_day
1	636249	116.18864134404662	Afternoon
2	289732	106.63673168936317	Evening
3	676444	104.03629652414618	Morning
4	549725	74.247028633171553	Night

- I aggregated the hourly dataset for calories to include seasonal information - month,day,part of week, time of day.
- The most calories were burned in the **morning (6AM to 12 noon)**, but on average (per hour), the **afternoons (12 noon to 6PM)** saw the most calories burned - 116.18kcal.
- The least calories burned every hour was at **night (9PM to 6AM)** with 74.24kcal.
- **Evenings (6PM to 9PM)** recorded the 2nd most calories burned on average with 106.3 kcal per hour, followed by the **mornings (6AM to 12 noon)** with 104.03kcal per hour.



```
WITH Tempo AS (
 SELECT
  Id,
  StepTotal,
  CASE
     WHEN TIME(ActivityHour) BETWEEN TIME(6, 0, 0) AND TIME(12, 0, 0) THEN "Morning"
     WHEN TIME(ActivityHour) BETWEEN TIME(12, 0, 0) AND TIME(18, 0, 0) THEN "Afternoon"
     WHEN TIME(ActivityHour) BETWEEN TIME(18, 0, 0) AND TIME(21, 0, 0) THEN "Evening"
     WHEN TIME(ActivityHour) >= TIME(21,0,0) OR TIME(TIMESTAMP_TRUNC(ActivityHour, MINUTE)) <= TIME(6,0,0) THEN "Night"
    ELSE NULL
  END AS time_of_day,


FROM `sqlpractice-626.Fitbit_data_set_April_May_2016.step_count_hourly`
)

SELECT
time_of_day,
SUM(StepTotal) AS total_step_tod,
AVG(StepTotal) AS avg_step_tod
FROM Tempo
GROUP BY time_of_day;
```

Press Alt+F1 for Accessibility Optio

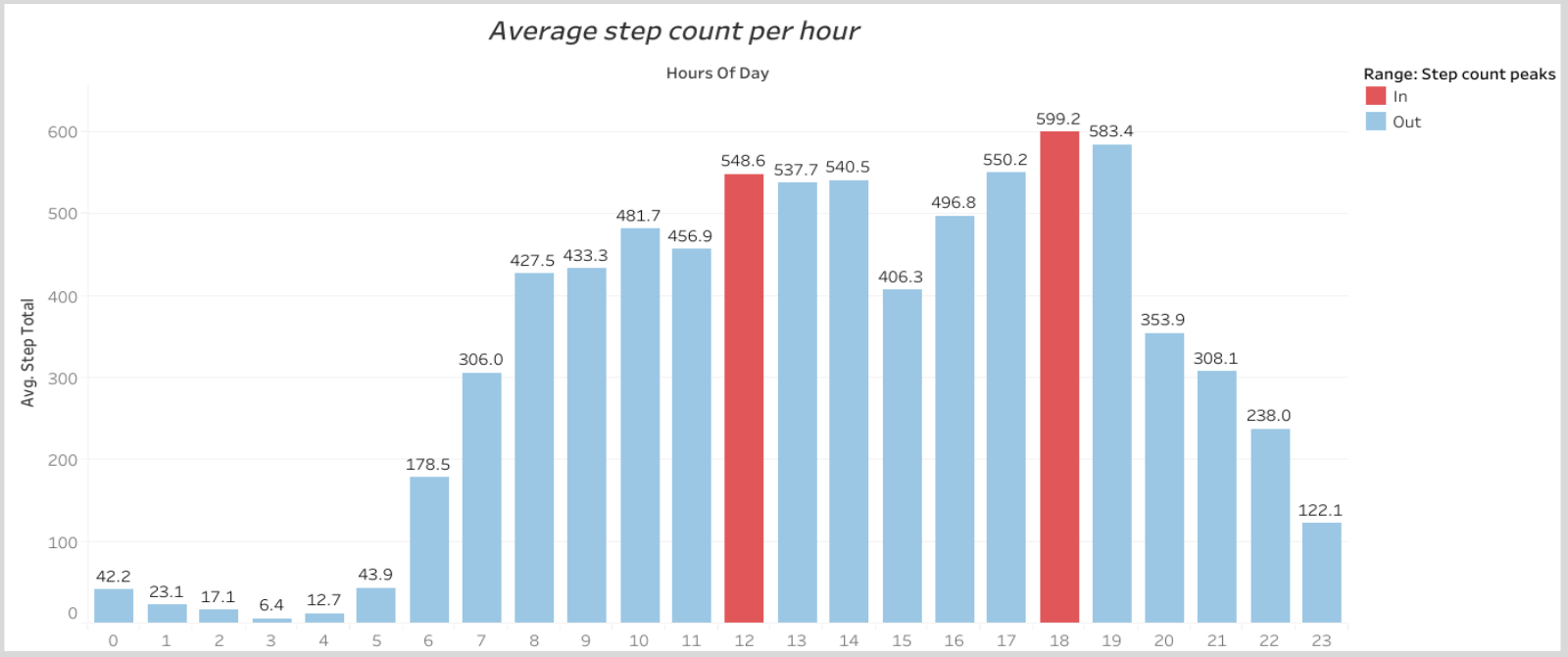
Query results

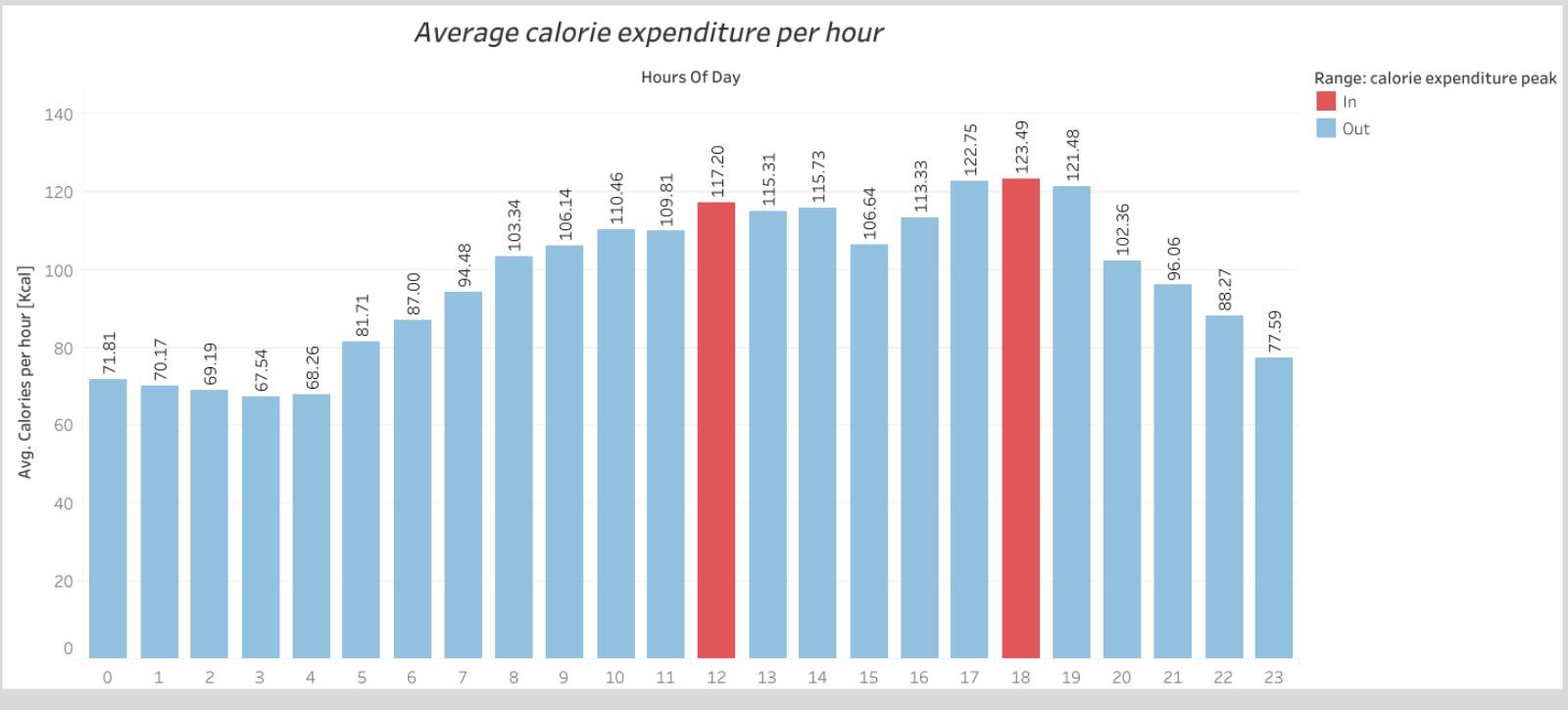
SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	time_of_day	total_step_tod	avg_step_tod
1	Afternoon	2856813	521.69704163623112
2	Evening	1128055	415.1840264998167
3	Morning	2629420	404.401722546909
4	Night	461068	62.272825499729862

- I aggregated the hourly dataset for steps to include seasonal information - time of day.
- On average, the highest daily step count occurred during the **afternoons (12 noon to 6PM)** with 521 steps hourly.
- The least steps taken per hour occurred at **night (9PM to 6AM)** with 62 steps.





- Throughout the course of a day (24 hours), step count and calorie expenditure peaked at 12:00 PM noon and 6:00 PM.

PHYSICAL ACTIVITIES (Light, Fair, Vigorous)

Seasonal data:

```
--Temp table from merged PA query 2 (`sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`)

SELECT
  Seasons,SUM(LightlyActiveMinutes) AS total_LAM_time, MAX(LightlyActiveMinutes) AS max_LAM_time,MIN(LightlyActiveMinutes) AS min_LAM_time, AVG(LightlyActiveMinutes) AS avg_LAM_time,  SUM(FairlyActiveMinutes) AS total_FAM_count, AVG
(FairlyActiveMinutes) AS avg_FAM_count, MAX(FairlyActiveMinutes) AS max_FAM_time, MIN(FairlyActiveMinutes) AS min_FAM_time, SUM(VeryActiveMinutes) AS total_VAM_time, AVG(VeryActiveMinutes) AS avg_VAM_time, MAX(VeryActiveMinutes) AS
max_VAM_time,MIN(VeryActiveMinutes) AS min_VAM_time, corr(steps,VeryActiveMinutes) AS correlation_bet_steps_VA_minutes, (AVG(FairlyActiveMinutes) + AVG(VeryActiveMinutes) + AVG(VeryActiveMinutes)) AS total_avg_active_minutes,
  (AVG(FairlyActiveMinutes) + AVG(VeryActiveMinutes) + AVG(VeryActiveMinutes))/3 AS avg_sum_avg_physical_activites

FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY Seasons;

---Grouped PA per season
```

Query results

SAVE RESULTS EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Seasons	total_LAM_time	max_LAM_time	min_LAM_time	avg_LAM_time	total_FAM_count	avg_FAM_count	max_FAM_time	min_FAM_time	total_VAM_time	avg_VAM_time	max_VAM_time	min_VAM_time	correlation_bet_steps_VA_minutes	total_avg_active_minutes	avg_sum_avg_physical_activites
1	Spring	1626027	686	0	87.4302075491…	293369	15.7742230347…	451	0	502388	27.0130121518…	630	0	0.6328822919452772 3	69.800247338423191	23.26674911280773
2	Fall	1498453	730	0	79.6456362283…	300404	15.9670458169…	934	0	483109	25.6781651961…	1440	0	0.6293329866923230 7	67.323376209205776	22.441125403068593
3	Winter	1393907	729	0	81.1307258017…	236732	13.7787090390…	326	0	399564	23.2561550550…	375	0	0.6344669467791528 3	60.291019149060205	20.097006838020069
4	Summer	1555481	1080	0	84.8228269167…	355034	19.3605627658…	620	0	504833	27.5293379866…	633	0	0.6509877957427248 1	74.419238739229925	24.806412913076642

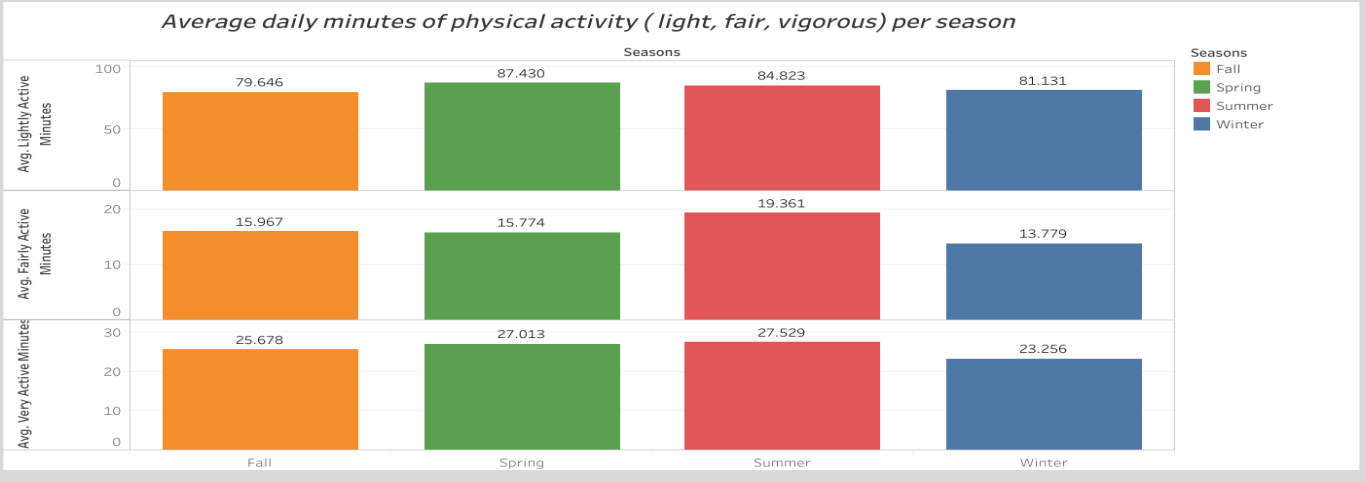
```
--Temp table from merged PA query 2 (`sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`)


SELECT
  Seasons,
  corr(Steps, LightlyActiveMinutes) AS rel_steps_light_act,
  corr(Steps, FairlyActiveMinutes) AS rel_steps_fairly_act,
  corr(Steps, VeryActiveMinutes) AS rel_steps_very_act,

  corr(LightlyActiveMinutes, Calories) AS rel_light_act_calories,
  corr(FairlyActiveMinutes, Calories)AS rel_fairly_act_calories,
  corr(VeryActiveMinutes, Calories)AS rel_very_act_calories,

FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY Seasons;

--- correlation between PA, calories, steps per season
```

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Seasons	rel_steps_light_act	rel_steps_fairly_act	rel_steps_very_act	rel_light_act_calories	rel_fairly_act_calories	rel_very_act_calories
1	Spring	0.13276253641744118	0.36879854251537569	0.63288229194527723	0.37056778855342376	0.42649482796491273	0.61554783646734679
2	Fall	0.093758169869225486	0.41271723998046267	0.62933298669232307	0.38648665020152134	0.42354075769873606	0.58541303366904429
3	Winter	0.1441016352717111	0.38157843454907459	0.63446694677915283	0.39520760250591686	0.42901319394027382	0.60032037056492993
4	Summer	0.1152405864580808	0.40522005706267561	0.65098779574272481	0.42205386029097314	0.45380123186913207	0.62263806509166886



light physical activities was during the **fall** season with an average of 79.64 minutes (01:19:38). Summer was 2nd with 84.82 minutes (01:24:49) , while winter was 3rd with 81.13 minutes (01:21:07).

- The highest average minutes spent during fairly physical activities was in the **Summer** with 19.36 minutes (00:19:21). The lowest average was during **winter** with 13.77 minutes (00:13:46). Spring came in 2nd with 15.77 minutes (00:15:46), and fall came in third with 15.96 minutes (00:15:57).
- Similar to fairly physical activities, **summer** recorded the highest average time spent doing vigorous physical activities with 27.52 minutes (00:27:31). **Winter** recorded the lowest average time spent with 23.15 minutes (00:23:15). Spring was 2nd with 27.01 minutes (00:27:00), and fall was third with 25.67 minutes (00:25:40).
- When comparing time spent amongst all 3 types of physical activities (Light, fair, and vigorous), there was a large disparity between light physical activities and fair / vigorous activities. Participants were seen to average over 1 hour a day engaging in light physical activities compared to fair or vigorous activities - both were sub or less than 30 minutes on average.
- Seeing as participants spent the most time doing light activities, I hypothesized the average time spent will follow in order of difficulty - lowest to highest difficulty. However, participants spent the least time - range of 13 to 19 minutes - engaging in fairly physical activities.
- The total sum of average minutes spent doing all 3 categories of physical activities show participants spent the most time during summer - 74. 41 minutes (01:14:24). The least time spent was during winter with 60.29 minutes (01:00:17). Spring came in 2nd with 69.80 (01:09:48), and then fall was 3rd with 67.32 minutes (01:07:19).

Monthly data:

```
12
13  SELECT  Months, SUM(LightlyActiveMinutes) AS total_LAM_time, MAX(LightlyActiveMinutes) AS max_LAM_time,MIN(LightlyActiveMinutes) AS min_LAM_time, AVG(LightlyActiveMinutes) AS avg_LAM_time,  SUM(FairlyActiveMinutes) AS total_FAM_count, AVG
    (FairlyActiveMinutes) AS avg_FAM_count, MAX(FairlyActiveMinutes) AS max_FAM_time, MIN(FairlyActiveMinutes) AS min_FAM_time, SUM(VeryActiveMinutes) AS total_VAM_time, AVG(VeryActiveMinutes) AS avg_VAM_time, MAX(VeryActiveMinutes) AS
    max_VAM_time,MIN(VeryActiveMinutes) AS min_VAM_time, corr(steps,VeryActiveMinutes) AS correlation_bet_steps_VA_minutes, (AVG(FairlyActiveMinutes) + AVG(VeryActiveMinutes) + AVG(VeryActiveMinutes)) AS total_avg_active_minutes,(AVG
    (FairlyActiveMinutes) + AVG(VeryActiveMinutes) + AVG(VeryActiveMinutes))/3 AS avg_sum_avg_physical_activites
14  FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
15  GROUP BY Months;
16  --Grouped PA Per month
17
18
```

Query results

SAVE RESULTS EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

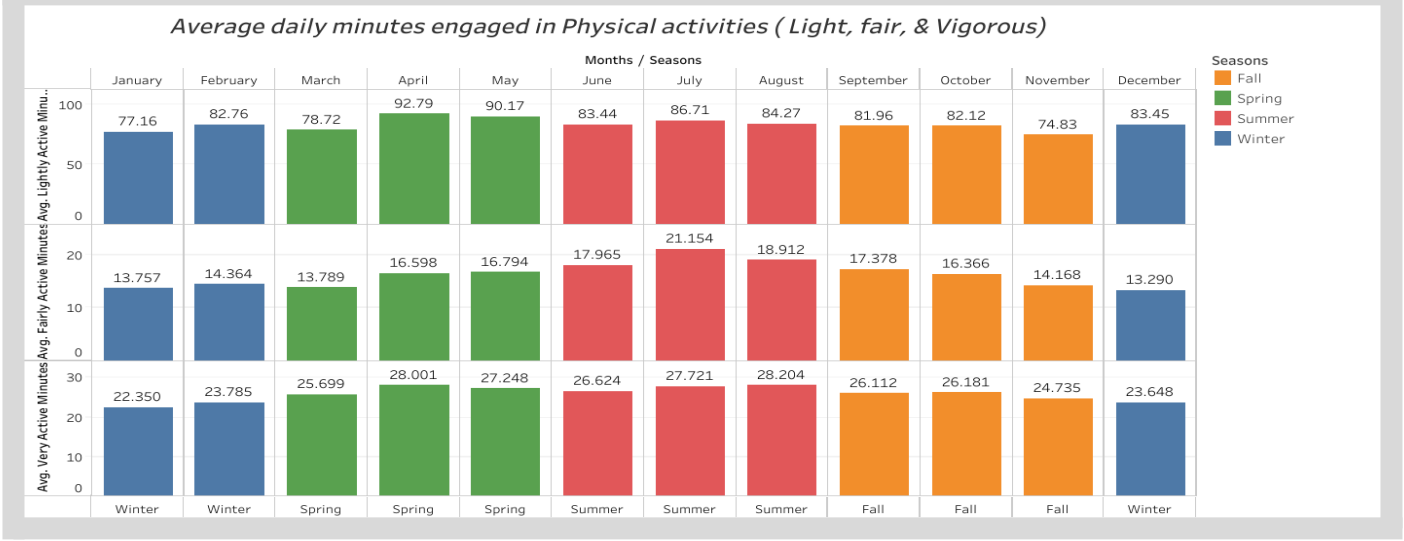
Row	Months	total_LAM_time	max_LAM_time	min_LAM_time	avg_LAM_time	total_FAM_count	avg_FAM_count	max_FAM_time	min_FAM_time	total_VAM_time	avg_VAM_time	max_VAM_time	min_VAM_time	correlation_bet_steps_VA_minut	total_avg_active_minutes	avg_sum_avg_physical_activites
1	May	573667	640	0	90.1708582206...	106844	16.7940899088...	451	0	173349	27.2475636592...	630	0	0.6197107450199171 9	71.289217227286883	23.763072409095
2	April	587999	686	0	92.7882278680...	105184	16.5983904055...	388	0	177440	28.0006312135...	532	0	0.6301136425160791 5	72.5996528325705	24.19988427752
3	September	504608	634	0	81.9567971414...	106994	17.3776189702...	934	0	160773	26.1122299821...	1440	0	0.5687544954659075 1	69.602078934545759	23.200692978181
4	October	526238	574	0	82.1220349563...	104872	16.3657927590...	334	0	167766	26.1807116104...	627	0	0.6860008577934848 5	68.727215980025	22.909071993341
5	November	467607	730	0	74.8290926548...	88538	14.1683469355...	312	0	154570	24.7351576252...	333	0	0.6497836887308526	63.638662185949833	21.212887395316
6	December	510109	729	0	83.4465892360...	81244	13.2903647963...	326	0	144559	23.6477997709...	319	0	0.6455639985478061 1	60.585964338295412	20.1953214460984
7	January	443692	667	0	77.1638260869...	79102	13.7568695652...	303	0	128514	22.3502608695...	375	0	0.6029835694839547 1	58.457391304347695	19.485797101449
8	February	440106	672	0	82.7578036855...	76386	14.3636705528...	227	0	126491	23.7854456562...	309	0	0.6550039809332112 2	61.934561865363015	20.644853955121
9	March	464361	513	0	78.7185963722...	81341	13.7889472791...	242	0	151599	25.6991015426...	367	0	0.6561394985439263 9	65.187150364468565	21.729050121489
10	June	495712	656	0	83.4391516579...	106732	17.9653257027...	456	0	158172	26.6238007069...	633	0	0.6227356760120642 5	71.21292711664708	23.737642372215
11	July	535599	680	0	86.7085964060...	130667	21.1537963412...	620	0	171235	27.7213857859...	355	0	0.6509053434331333	76.596567913226565	25.532189304408
12	August	524170	1080	0	84.2717041800 64	117635	18.9123794212...	461	0	175426	28.2035397744...	608	0	0.6792372802097216	75.319453376205843	25.106484587852

```
18
19
20  SELECT  Months,
21    corr(Steps, LightlyActiveMinutes) AS rel_steps_light_act,
22    corr(Steps, FairlyActiveMinutes) AS rel_steps_fairly_act,
23    corr(Steps, VeryActiveMinutes) AS rel_steps_very_act,
24
25    corr(LightlyActiveMinutes, Calories) AS rel_light_act_calories,
26    corr(FairlyActiveMinutes, Calories)AS rel_fairly_act_calories,
27    corr(VeryActiveMinutes, Calories)AS rel_very_act_calories,
28
29  FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
30  GROUP BY Months;
31  ----- correlation between PA, calories, steps per season Per month
32
```

Query results

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Months	rel_steps_light_act	rel_steps_fairly_act	rel_steps_very_act	rel_light_act_calories	rel_fairly_act_calories	rel_very_act_calories
1	May	0.11710410085534693	0.34931315411842712	0.61971074501991719	0.373517431813479	0.435234563445096	0.61694276654784053
2	April	0.13721564225388733	0.387012334127041	0.63011364251607915	0.33823553648011284	0.40037406253472696	0.6231081192088308
3	September	0.053250102303810734	0.46002896021125544	0.56875449546590751	0.36591604327399768	0.46649900319444032	0.52050791154919251
4	October	0.10597490651688053	0.38802455319034107	0.68600085779348485	0.40396514121221438	0.41063862099908743	0.63922706388005457
5	November	0.12453192695866133	0.36934916865282735	0.64978364887308526	0.3876938654891251	0.38217119781451797	0.6133143957210756
6	December	0.11742687283319209	0.37412331423493711	0.64556399854780611	0.382810521268926	0.39806385710265146	0.57748920485984978
7	January	0.16335519816851046	0.37767375198142394	0.60298356948395471	0.39159506153903967	0.44488464237488756	0.62408582464255313
8	February	0.15420755012065079	0.39572767563811068	0.65500398093321122	0.41649657362941406	0.44620305347490286	0.60177309100006671
9	March	0.14407125185460781	0.37291815083161162	0.65613949854392639	0.40708698125263554	0.46451618786136539	0.60968719343011812
10	June	0.12818893400188139	0.3589506396516694	0.62273567601206425	0.42642822520939488	0.44399143403918423	0.63159454079217714
11	July	0.13709309281747539	0.44425805494543019	0.65090534343313133	0.44270607015533692	0.45653521561748983	0.62705302269288887
12	August	0.0776755372177269	0.40141621989223741	0.67923728802097216	0.39497759404481164	0.4648079729060593	0.61109689469068607



- **April (Spring), May (Spring) and July (Summer)** were the top 3 months averaging the highest minutes of light physical activities - 92.78 minutes (01:32:46), 90.17 minutes (01:30:10), and 86.70 minutes (01:26:42) respectively. The least time was spent during **March (Spring), January (Winter), and November (Fall)** - 78.71 minutes (01:18:42), 77.16 minutes (01:17:09), and 74.82 minutes (01:14:49) respectively.
- **July, August, and June -Summer** - recorded the highest average minutes engaging in fairly physical activities - 21.15 minutes (00:21:09), 18.91 minutes (00:18:54), and 17.96 minutes (00:17:57) respectively. The least minutes spent doing fairly physical activities was during **March (Spring), January (Winter) and December (Winter)** - 13.78 minutes (00:13:46), 13.75 minutes (00:13:45), and 13.29 minutes (00:13:17) respectively.
- **August (Summer), April (Spring) and July(Summer)** - recorded the highest average minutes engaging in vigorous physical activities - 28.20 minutes (00:28:12), 28.00 minutes (00:28:00), and 27.72 minutes (00:27:43) respectively. The least minutes spent doing vigorous physical activities was during **February (Spring), December(Winter) and January (Winter)** - 23.78 minutes (00:23:46), 23.45 minutes (00:23:27), and 22.35 minutes (00:22:21) respectively.
- July and April are the most common months experiencing high minutes of physical activities.
- When comparing time spent amongst all 3 types of physical activities (Light, fair, and vigorous), there was a large disparity between light physical activities and fair & vigorous activities. Participants were seen to average over 1 hour a day engaging in light physical activities compared to fair or vigorous activities - both were sub or less than 30 minutes on average.
- Seeing as participants spent the most time doing light activities, I hypothesized the average time spent will follow in order of difficulty - lowest to highest difficulty. However, participants spent the least time - range of 13 to 21 minutes - engaging in fairly physical activities.
- The total sum of average minutes spent doing all 3 categories of physical activities show participants spent the most time in July, August, and April respectively. The least time spent was during February, December and January respectively.
- We can infer participants spend more time doing physical activities during the summer and spring, and the least during winter - specifically in January.

Day of the week:

```
SELECT  day_of_week, SUM(LightlyActiveMinutes) AS total_LAM_time, MAX(LightlyActiveMinutes) AS max_LAM_time,MIN(LightlyActiveMinutes) AS min_LAM_time, AVG(LightlyActiveMinutes) AS avg_LAM_time,  SUM
(FairlyActiveMinutes) AS total_FAM_count, AVG(FairlyActiveMinutes) AS avg_FAM_count, MAX(FairlyActiveMinutes) AS max_FAM_time, MIN(FairlyActiveMinutes) AS min_FAM_time, SUM(VeryActiveMinutes) AS total_VAM_time,
AVG(VeryActiveMinutes) AS avg_VAM_time, MAX(VeryActiveMinutes) AS max_VAM_time,MIN(VeryActiveMinutes) AS min_VAM_time, corr(steps,VeryActiveMinutes) AS correlation_bet_steps_VA_minutes, (AVG
(FairlyActiveMinutes) + AVG(VeryActiveMinutes) + AVG(VeryActiveMinutes)) AS total_avg_active_minutes, (AVG(FairlyActiveMinutes) + AVG(VeryActiveMinutes) + AVG(VeryActiveMinutes))/3 AS
avg_sum_avg_physical_activites
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY day_of_week;
--Grouped PA Per Day
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	day_of_week	total_LAM_time	max_LAM_time	min_LAM_time	avg_LAM_time	total_FAM_count	avg_FAM_count	max_FAM_time	min_FAM_time	total_VAM_time	avg_VAM_time	max_VAM_time	min_VAM_time	correlation_bet_steps_VA_minutes	total_avg_active_minutes	avg_sum_avg_physical_activites
1	Sunday	780324	596	0	75.3717762967...	176913	17.0880904085...	379	0	298256	28.8086544962...	633	0	0.6914408831577662	74.7053994011398 86	24.901799800379962
2	Thursday	888974	680	0	84.8014881236...	165852	15.8210435943...	461	0	258876	24.6948392635...	330	0	0.6148551534324101 5	65.210722121529955	21.736907373843319
3	Tuesday	875250	598	0	83.0486763450...	160667	15.2449947812...	268	0	263543	25.0064522250...	1412	0	0.5811456314911904 7	65.257899231426 1	21.752693077142033
4	Saturday	898601	730	0	86.6789813832...	202384	19.5219446320...	934	0	313553	30.2452975788...	627	0	0.6720018709154378	80.012539789717351	26.670846596572449
5	Wednesday	873343	674	0	83.4776333997...	162496	15.5320206461...	620	0	260349	24.8852035939...	380	0	0.630035919698859	65.302427834066322	21.767475944688773
6	Monday	863965	609	0	83.0655706182...	160408	15.4223692343...	374	0	268929	25.8560715315...	1440	0	0.5989809274474289 2	67.134506297471319	22.378168765823773
7	Friday	893411	1080	0	86.5205306992...	156819	15.1868099941...	444	0	226388	21.9240751501...	592	0	0.6324484364489046	59.034960294402488	19.678320098134161

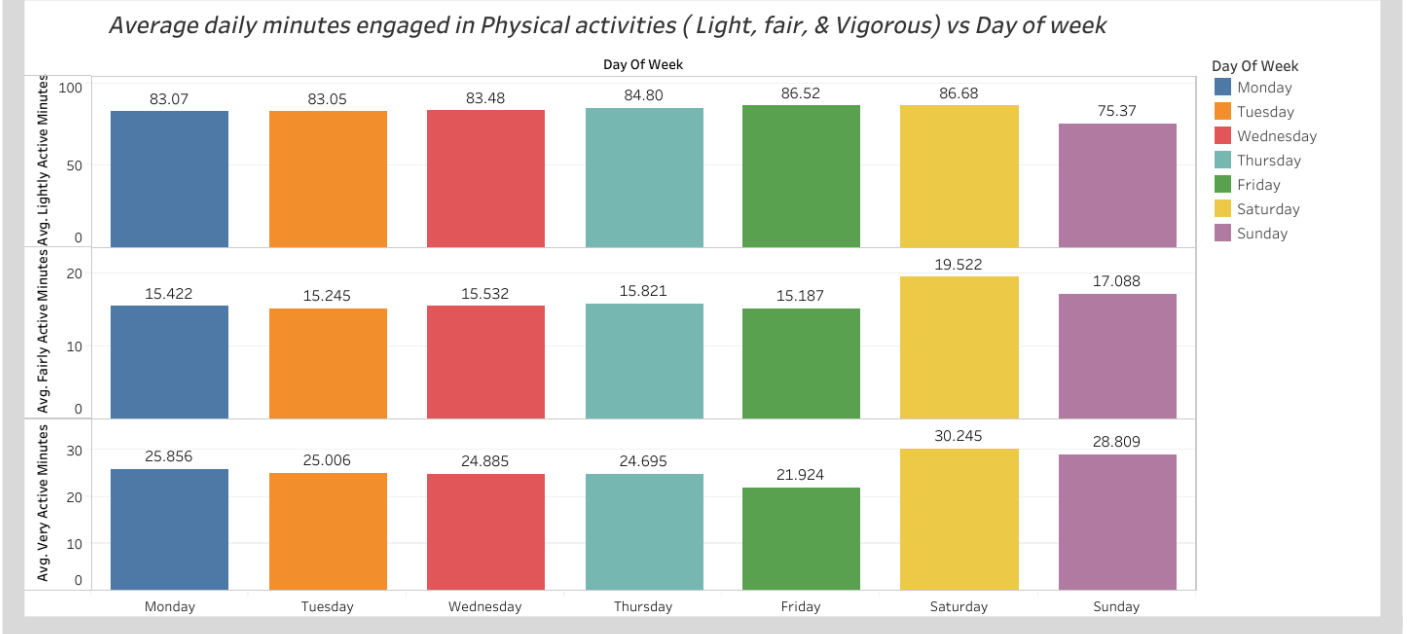
```
SELECT  day_of_week,
  corr(Steps, LightlyActiveMinutes) AS rel_steps_light_act,
  corr(Steps, FairlyActiveMinutes) AS rel_steps_fairly_act,
  corr(Steps, VeryActiveMinutes) AS rel_steps_very_act,

  corr(LightlyActiveMinutes, Calories) AS rel_light_act_calories,
  corr(FairlyActiveMinutes, Calories)AS rel_fairly_act_calories,
  corr(VeryActiveMinutes, Calories)AS rel_very_act_calories,

FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY day_of_week;
----- correlation between PA, calories, steps per season Per Day
```

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	day_of_week	rel_steps_light_act	rel_steps_fairly_act	rel_steps_very_act	rel_light_act_calories	rel_fairly_act_calories	rel_very_act_calories
1	Sunday	0.059635623714522031	0.37632735192738281	0.6914408831577662	0.32595339769269716	0.43716665121021109	0.654087870796521
2	Thursday	0.14484154185572	0.38887600938630368	0.61485515343241015	0.41368731658348862	0.4219058022865888	0.59533589811302179
3	Tuesday	0.15211259816078393	0.39261675671700558	0.5811456314911904 7	0.41235986011000664	0.436868140723830 4	0.56281103159365242
4	Saturday	0.059093101603060895	0.42678085680561273	0.67200180709154378	0.346463398093635	0.444906669703619	0.63390900591527766
5	Wednesday	0.13486094139761423	0.3679400035048187	0.630035919698859	0.4126950198702567	0.41465936310985868	0.60091493931535722
6	Monday	0.12950175549788795	0.40047992275304251	0.59898092744742892	0.40589759767454242	0.41850161168105016	0.57889187885461235
7	Friday	0.18297922191382163	0.40797100994244923	0.63244843644489046	0.44877823636777392	0.45441672645159464	0.59288661518021035



- Participants recorded the highest average minutes engaging in light physical activities - 87.67 minutes (01:27:40)- on **Saturdays**. The least time spent doing light physical activities was on **Sundays** with 75.37 minutes (01:15:22). During weekdays, Fridays were the most active with 86.52 minutes (01:26:31), while Tuesdays were the least with 83.04 minutes (01:23:02).
- Participants recorded the highest average minutes engaging in fairly physical activities - 19.52 minutes (00:19:31) - on **Saturdays**. The least time spent doing fairly physical activities was on **Fridays** with 15.18 minutes (00:15:18). During weekdays, Thursdays were the most active with 15.82 minutes (00:15:49).
- Participants recorded the highest average minutes engaging in vigorous physical activities - 30.34 minutes (00:30:20) - on **Saturdays**. The least time spent doing vigorous physical activities was on **Fridays** with 21.92 minutes (00:21:55). During weekdays, Mondays were the most active with 25.85 minutes 00:25:51).
- Results show Saturdays are when participants spend most minutes engaging in all kinds of physical activities (light, fair, and vigorous). Fridays and Sundays recorded the least time/minutes on average spent engaging in physical activities.
- When comparing time spent amongst all 3 types of physical activities (Light, fair, and vigorous), there was a large disparity between light physical activities and fair & vigorous activities. Participants were seen to average over 1 hour a day engaging in light physical activities compared to fair or vigorous activities - both were sub or less than 30 minutes on average.
- Seeing as participants spent the most time doing light activities, I hypothesized the average time spent will follow in order of difficulty - lowest to highest difficulty. However, participants spent the least time - range of 15 to 19 minutes - engaging in fairly physical activities.
- The max values for each level of physical activity was considerably higher than the averages. Taking into account participants recorded zero values in each category of physical activity.
- The total sum of average minutes spent doing all 3 categories of physical activities show participants spent the most time on Saturdays. The least time spent on Fridays.
- We can infer participants spend the most time doing activities on Saturdays and the least time on Fridays.

Part of the week:

```sql
SELECT  part_of_week, SUM(LightlyActiveMinutes) AS total_LAM_time, MAX(LightlyActiveMinutes) AS max_LAM_time,MIN(LightlyActiveMinutes) AS min_LAM_time, AVG(LightlyActiveMinutes) AS avg_LAM_time,  SUM
(FairlyActiveMinutes) AS total_FAM_count, AVG(FairlyActiveMinutes) AS avg_FAM_count, MAX(FairlyActiveMinutes) AS max_FAM_time, MIN(FairlyActiveMinutes) AS min_FAM_time, SUM(VeryActiveMinutes) AS total_VAM_time,
AVG(VeryActiveMinutes) AS avg_VAM_time, MAX(VeryActiveMinutes) AS max_VAM_time,MIN(VeryActiveMinutes) AS min_VAM_time, corr(steps,VeryActiveMinutes) AS correlation_bet_steps_VA_minutes, (AVG
(FairlyActiveMinutes) + AVG(VeryActiveMinutes) + AVG(VeryActiveMinutes)) AS total_avg_active_minutes, (AVG(FairlyActiveMinutes) + AVG(VeryActiveMinutes) + AVG(VeryActiveMinutes))/3 AS
avg_sum_avg_physical_activites
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY part_of_week;
--Grouped PA per Part of the week
```

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	part_of_week	total_LAM_time	max_LAM_time	min_LAM_time	avg_LAM_time	total_FAM_count	avg_FAM_count	max_FAM_time	min_FAM_time	total_VAM_time	avg_VAM_time	max_VAM_time	min_VAM_time	correlation_bet_steps_VA_minutes	total_avg_active_minutes	avg_sum_avg_physical_activites
1	Weekend	1678925	730	0	81.0291988416...	379297	18.3058397683...	934	0	611809	29.5274613899...	633	0	0.6803265840940641	77.360762548262386	25.786920849420795
2	Weekday	4394943	1080	0	84.1765719867...	806242	15.4419949819...	620	0	1278085	24.4792285150...	1440	0	0.6106245907105987	64.40045201202814	21.466817337342714

```sql
SELECT  part_of_week,
  corr(Steps, LightlyActiveMinutes) AS rel_steps_light_act,
  corr(Steps, FairlyActiveMinutes) AS rel_steps_fairly_act,
  corr(Steps, VeryActiveMinutes) AS rel_steps_very_act,

  corr(LightlyActiveMinutes, Calories) AS rel_light_act_calories,
  corr(FairlyActiveMinutes, Calories)AS rel_fairly_act_calories,
  corr(VeryActiveMinutes, Calories)AS rel_very_act_calories,

FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anona70e7e9d6139b21d40d8dfa673dfa16c778aa6ee871ed88db1e46889a924f52b`
GROUP BY part_of_week;
----- correlation between PA, calories, steps per season per Part of the week -------------------
```

Query results

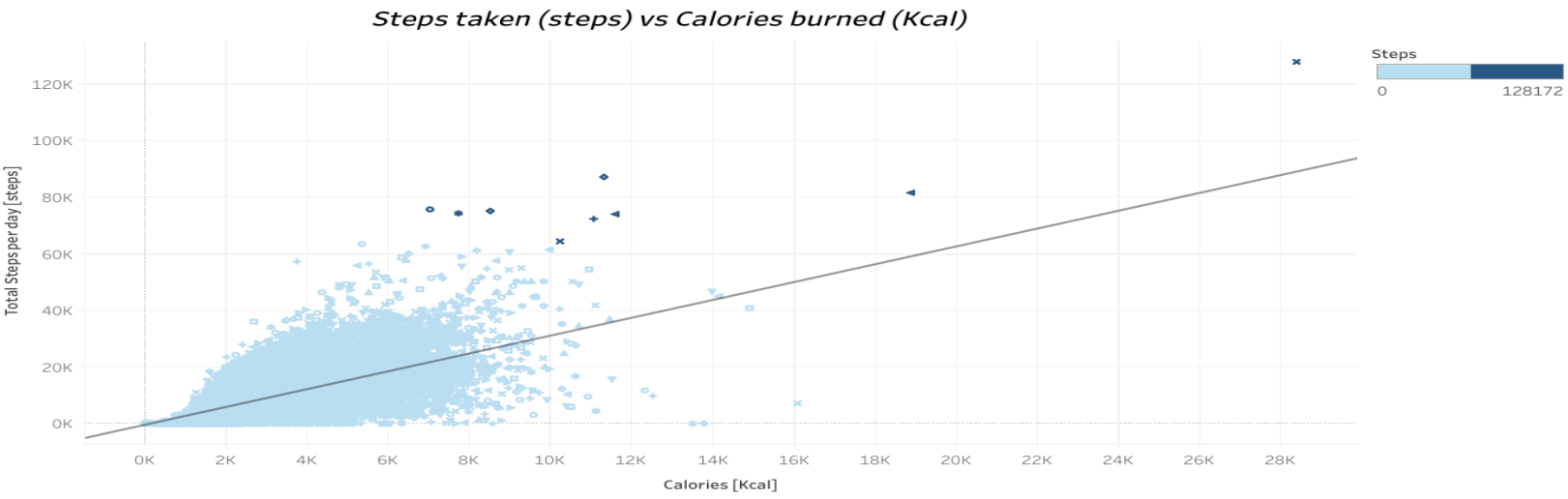
SAVE RESULTS

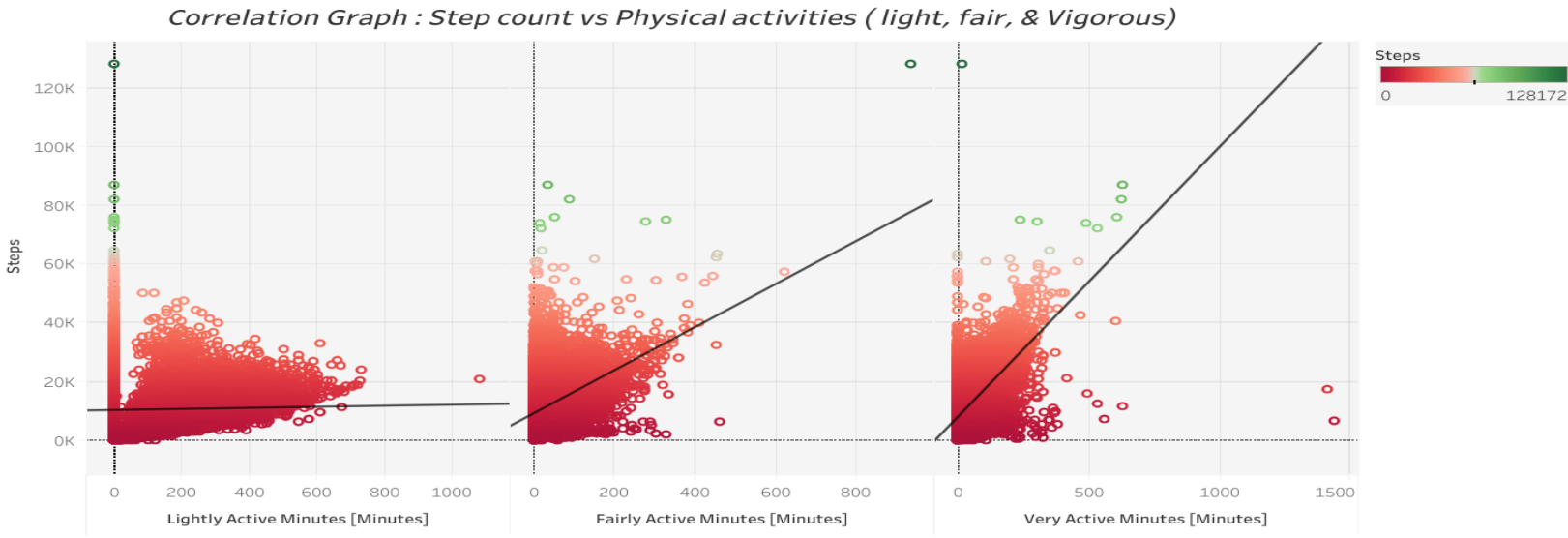
JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

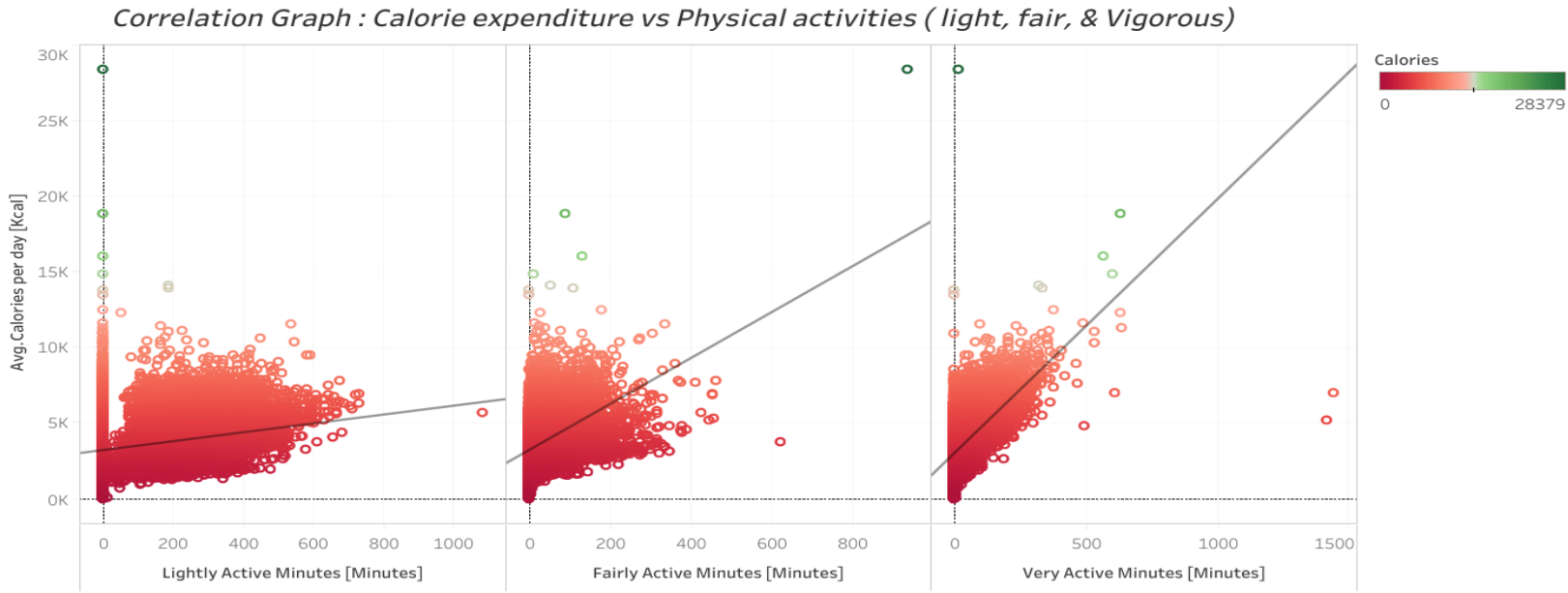
Row	part_of_week	rel_steps_light_act	rel_steps_fairly_act	rel_steps_very_act	rel_light_act_calories	rel_fairly_act_calories	rel_very_act_calories
1	Weekend	0.06219622939876277 2	0.40560051723265578	0.68032658840940641	0.33910437047694181	0.442155911934902 39	0.64277257672394483
2	Weekday	0.14871722745814711	0.39133190487372327	0.61062459707105987	0.41886981465159495	0.42914690584085435	0.58518487668830987

- **Weekdays** recorded the highest average minutes spent doing light physical activities - 84.17 minutes (01:24:10).
- **Weekends** recorded the highest average minutes spent doing fairly physical activities - 18.30 minutes (00:18:18).
- **Weekends** recorded the highest average minutes spent doing vigorous physical activities - 29.52 minutes (00:29:31).
- When comparing all 3 classes of physical activities, participants spent the most minutes during the **weekends** - 77.36 (01:17:21). The weekdays recorded 64.40 minutes (01:04:24).
- Although participants spent more time doing light activities on the weekdays, overall participants spent the most time doing all 3 kinds of physical activities on the weekend.

Correlation between PA - steps and calories:







correlation coefficient

This shows a possitive correlation between calorie expenditure and steps taken

0.6730

P-value: less than 0.0001 making this a significant test

correlation coefficient

This shows a possitive correlation between steps taken and physical activities

Light: 0.1198

Fair: 0.3970

Vigorous: 0.6368

P-value: less than 0.0001 making this a significant test

correlation coefficient

This shows a possitive correlation between calorie expenditure and physical activities

Light: 0.3939

Fair: 0.4334

Vigorous: 0.6046

P-value: less than 0.0001 making this a significant test

- There was positive correlation between calorie expenditure and all 3 types (light, fairly, and very) of physical activities.
- There was also positive correlation between the step count and all 3 types (light, fairly, and very) of physical activities.
- The strength in correlation coefficient was dependent on the difficulty of the physical category, with vigorous physical activities having the highest and vice versa.
- The higher in difficulty the physical activity, the more calories were burned and steps taken.

SLEEP ACTIVITY

Monthly data:

```sql
SELECT
  Id,
  SleepDay,
  TotalTimeInBed,
  TotalMinutesAsleep,
  (TotalTimeInBed-TotalMinutesAsleep)AS lay_time,
  TIME(div(TotalMinutesAsleep,60),MOD(TotalMinutesAsleep,60),MOD(MOD(TotalMinutesAsleep*60,3600),60) ) AS sleep_time,
  TIME(div(TotalTimeInBed,60),MOD(TotalTimeInBed,60),MOD(MOD(TotalTimeInBed*60,3600),60)) AS Bed_time,
  TIME( div((TotalTimeInBed-TotalMinutesAsleep),60),
        MOD((TotalTimeInBed-TotalMinutesAsleep),60),
        MOD(MOD((TotalTimeInBed-TotalMinutesAsleep)*60,3600),60)
      ) AS Laydown_time,
  COUNT(TotalSleepRecords) AS number_naps,
  CASE
    WHEN DATE(SleepDay) BETWEEN "2016-04-01" AND "2016-04-30" THEN "April"
    WHEN DATE(SleepDay) BETWEEN "2016-05-01" AND "2016-05-31" THEN "May"
    ELSE NULL
  END AS Months,

  FORMAT_TIMESTAMP("%A", SleepDay) AS day_of_week,
  CASE
     WHEN FORMAT_TIMESTAMP("%A", SleepDay) IN ("Sunday", "Saturday") THEN "Weekend"
     WHEN FORMAT_TIMESTAMP("%A", SleepDay) NOT IN ("Sunday",
     "Saturday") THEN "Weekday"
    ELSE
    "ERROR"
  END
    AS part_of_week,
FROM `sqlpractice-626.Fitbit_data_set_April_May_2016.sleepDay`
GROUP BY Id, SleepDay, TotalMinutesAsleep, TotalTimeInBed, TotalSleepRecords;
```

```sql
--- Temporary table retrieved from aggregated query
SELECT
  Months,
  MAX(Bed_time) AS max_bed_time,
  MAX(sleep_time) AS max_sleep_time,
  MAX(Laydown_time) AS max_laydown_time,
  AVG(TotalTimeInBed) AS avg_bed_time_min,
  AVG(TotalMinutesAsleep) AS avg_sleep_time_min,
  AVG(lay_time) AS avg_lay_time_min,

  TIME(CAST(FLOOR( CAST(FLOOR(avg(TotalTimeInBed)) AS INT64)/60) AS INT64), CAST(FLOOR(MOD(CAST(FLOOR(avg(TotalTimeInBed)) AS INT64),60))AS INT64), CAST(FLOOR(MOD((MOD( CAST(FLOOR(avg(TotalTimeInBed)) AS INT64),60)),1)*60) AS INT64) ) AS avg_bed_time_t,

  TIME(CAST(FLOOR( CAST(FLOOR(avg(TotalMinutesAsleep)) AS INT64)/60) AS INT64), CAST(FLOOR(MOD(CAST(FLOOR(avg(TotalMinutesAsleep)) AS INT64),60))AS INT64), CAST(FLOOR(MOD((MOD( CAST(FLOOR(avg(TotalMinutesAsleep)) AS INT64),60)),1)*60) AS INT64) ) AS avg_sleep_time_t,

  TIME(CAST(FLOOR( CAST(FLOOR(avg(lay_time)) AS INT64)/60) AS INT64), CAST(FLOOR(MOD(CAST(FLOOR(avg(lay_time)) AS INT64),60))AS INT64), CAST(FLOOR(MOD((MOD( CAST(FLOOR(avg(lay_time)) AS INT64),60)),1)*60) AS INT64) ) AS avg_laydown_time_t,

  /*TIME(div(CAST(AVG(TotalMinutesAsleep)AS INT64),60),MOD(CAST(AVG(TotalMinutesAsleep)AS INT64),60), MOD(MOD(CAST(AVG(TotalMinutesAsleep)AS INT64)*60, 3600),60)) AS avg_sleepy_time  FLOOR((MOD((AVG(TotalTimeInBed)),60) % 1)*60)
  CAST(CAST(AVG(CAST(CAST(sleep_time AS STRING FORMAT "HH")AS FLOAT64))AS STRING ) AS TIME) AS avg_Bedd_time,
  CAST(CAST(AVG(CAST(CAST(sleep_time AS STRING)AS FLOAT64))AS STRING) AS TIME) AS avg_Sleep_time,
  CAST(CAST(AVG(CAST(CAST(Laydown_time AS STRING)AS FLOAT64))AS STRING) AS TIME) AS avg_Laydown_time*/
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anon5d5676cfe2c442ed23f83c7d9d7fff726a7748a6ec180c1f26a1e388aaa35cb8`
GROUP BY Months;

--Sleep data per Month
```

Press Alt+F1 for Accessibility Opti

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Months	max_bed_time	max_sleep_time	max_laydown_time	avg_bed_time_min	avg_sleep_time_min	avg_lay_time_min	avg_bed_time_t	avg_sleep_time_t	avg_laydown_time_t
1	April	16:01:00	12:55:00	05:17:00	458.35984848484844	419.63257575757586	38.727272727272712	07:38:00	06:59:00	00:38:00
2	May	16:01:00	13:16:00	06:11:00	458.70547945205487	418.34246575342456	40.363013698630155	07:38:00	06:58:00	00:40:00

- **April** and **May** months had comparable averages for time in bed per day - 458.35 (07:38:21) and 458.70 minutes (07:38:00).
- **April** and **May** months had comparable averages for sleep time per day- 419.63 (06:59:00) and 418.34 minutes (06:58:00)
- The average laydown time - time in bed not sleeping - was 38 minutes in April, and 40 minutes in May.
- Participants on average spent less than 7 hours asleep in both months.

Day of week data:

```sql
SELECT
  day_of_week,
  MAX(Bed_time) AS max_bed_time,
  MAX(sleep_time) AS max_sleep_time,
  MAX(Laydown_time) AS max_laydown_time,
  AVG(TotalTimeInBed) AS avg_bed_time_min,
  AVG(TotalMinutesAsleep) AS avg_sleep_time_min,
  AVG(lay_time) AS avg_lay_time_min,

  TIME(CAST(FLOOR( CAST(FLOOR(avg(TotalTimeInBed)) AS INT64)/60) AS INT64), CAST(FLOOR(MOD(CAST(FLOOR(avg(TotalTimeInBed)) AS INT64),60))AS INT64), CAST(FLOOR(MOD((MOD( CAST(FLOOR(avg(TotalTimeInBed)) AS INT64),60)),1)*60) AS INT64) ) AS avg_bed_time_t,

  TIME(CAST(FLOOR( CAST(FLOOR(avg(TotalMinutesAsleep)) AS INT64)/60) AS INT64), CAST(FLOOR(MOD(CAST(FLOOR(avg(TotalMinutesAsleep)) AS INT64),60))AS INT64), CAST(FLOOR(MOD((MOD( CAST(FLOOR(avg(TotalMinutesAsleep)) AS INT64),60)),1)*60) AS INT64) ) AS avg_sleep_time_t,

  TIME(CAST(FLOOR( CAST(FLOOR(avg(lay_time)) AS INT64)/60) AS INT64), CAST(FLOOR(MOD(CAST(FLOOR(avg(lay_time)) AS INT64),60))AS INT64), CAST(FLOOR(MOD((MOD( CAST(FLOOR(avg(lay_time)) AS INT64),60)),1)*60) AS INT64) ) AS avg_laydown_time_t,
  /*CAST(CAST(AVG(CAST(CAST(sleep_time AS STRING FORMAT "HH24")AS FLOAT64))AS STRING) AS TIME FORMAT "HH24") AS avg_Bed_time,
  CAST(CAST(AVG(CAST(CAST(sleep_time AS STRING)AS FLOAT64))AS STRING) AS TIME) AS avg_Sleep_time,
  CAST(CAST(AVG(CAST(CAST(Laydown_time AS STRING)AS FLOAT64))AS STRING) AS TIME) AS avg_Laydown_time*/
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anon5d5676cfe2c442ed23f83c7d9d7fff726a7748a6ec180c1f26a1e388aaa35cb8`
GROUP BY day_of_week;

-- Day of week data
```

Press Alt+F1 for Accessibility Optio

Query results

SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	day_of_week	max_bed_time	max_sleep_time	max_laydown_time	avg_bed_time_min	avg_sleep_time_min	avg_lay_time_min	avg_bed_time_t	avg_sleep_time_t	avg_laydown_time_t
1	Tuesday	12:55:00	12:30:00	03:28:00	443.2923076923...	404.5384615384...	38.7538461538461...	07:23:00	06:44:00	00:38:00
2	Friday	16:01:00	10:58:00	05:17:00	445.0526315789...	405.4210526315...	39.6315789473684...	07:25:00	06:45:00	00:39:00
3	Sunday	16:01:00	11:40:00	06:11:00	503.5090909090...	452.7454545454...	50.7636363636363...	08:23:00	07:32:00	00:50:00
4	Wednesday	11:19:00	10:58:00	03:11:00	470.0303030303...	434.6818181818...	35.3484848484848...	07:50:00	07:14:00	00:35:00
5	Thursday	09:28:00	09:05:00	03:00:00	434.8750000000...	401.2968749999...	33.5781249999999...	07:14:00	06:41:00	00:33:00
6	Saturday	16:01:00	12:55:00	03:59:00	459.8421052631...	419.0701754385...	40.7719298245614...	07:39:00	06:59:00	00:40:00
7	Monday	16:01:00	13:16:00	03:25:00	457.3478260869...	419.5000000000...	37.8478260869565...	07:37:00	06:59:00	00:37:00

- Participants recorded the highest average time in bed on **Sundays** with 503.50 minutes (08:23:00). The least amount of time spent in bed was 434.87 minutes (7:14:00) on **Thursdays**. The days inbetween are as follows - Wednesdays, Saturdays , Mondays, Fridays and Tuesdays.
- Participants recorded the highest average sleep time on **Sundays** with 452.74 minutes (07:32:00). The least amount of time sleeping was 401.29 minutes (06:41:00) on **Thursdays**. The days inbetween are as follows - Wednesdays, Mondays, Saturdays, Fridays and Tuesdays.
- Participants recorded the highest average time layed in bed was on **Sundays** with 50.76 minutes (00:50:00). The least amount of time layed in bed was 33.57 minutes (00:33:00) on **Thursdays**. The days inbetween are as follows - Saturdays, Fridays, Tuesdays, Mondays, and Wednesdays.
- The range between the highest and least average time spent sleeping is 51 minutes. It was 17 minutes for laydown time and 69 minutes for bedtime.
- For laydown time and bedtime, there was a large drop off in minutes when comparing the highest average from the corresponding days. I.e the largest difference in minutes occurred within the highest and second highest average.
- Across all 3 categories (Time in bed, sleep time, and laydown time), **Sundays** and **Thursdays** maintained their positions as the average highest and lowest minutes.

Part of week data:

```sql
SELECT
  part_of_week,
  MAX(Bed_time) AS max_bed_time,
  MAX(sleep_time) AS max_sleep_time,
  MAX(Laydown_time) AS max_laydown_time,
  AVG(TotalTimeInBed) AS avg_bed_time_min,
  AVG(TotalMinutesAsleep) AS avg_sleep_time_min,
  AVG(lay_time) AS avg_lay_time_min,

  TIME(CAST(FLOOR( CAST(FLOOR(avg(TotalTimeInBed)) AS INT64)/60) AS INT64), CAST(FLOOR(MOD(CAST(FLOOR(avg(TotalTimeInBed)) AS INT64),60))AS INT64), CAST(FLOOR(MOD((MOD( CAST(FLOOR(avg(TotalTimeInBed)) AS INT64),60)),1)*60) AS INT64) ) AS avg_bed_time_t,

  TIME(CAST(FLOOR( CAST(FLOOR(avg(TotalMinutesAsleep)) AS INT64)/60) AS INT64), CAST(FLOOR(MOD(CAST(FLOOR(avg(TotalMinutesAsleep)) AS INT64),60))AS INT64), CAST(FLOOR(MOD((MOD( CAST(FLOOR(avg(TotalMinutesAsleep)) AS INT64),60)),1)*60) AS INT64) ) AS avg_sleep_time_t,

  TIME(CAST(FLOOR( CAST(FLOOR(avg(lay_time)) AS INT64)/60) AS INT64), CAST(FLOOR(MOD(CAST(FLOOR(avg(lay_time)) AS INT64),60))AS INT64), CAST(FLOOR(MOD((MOD( CAST(FLOOR(avg(lay_time)) AS INT64),60)),1)*60) AS INT64) ) AS avg_laydown_time_t,
  /*CAST(CAST(AVG(CAST(CAST(sleep_time AS STRING FORMAT "HH24")AS FLOAT64))AS STRING) AS TIME FORMAT "HH24") AS avg_Bed_time,
  CAST(CAST(AVG(CAST(CAST(sleep_time AS STRING)AS FLOAT64))AS STRING) AS TIME) AS avg_Sleep_time,
  CAST(CAST(AVG(CAST(CAST(Laydown_time AS STRING)AS FLOAT64))AS STRING) AS TIME) AS avg_Laydown_time*/
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anon5d5676cfe2c442ed23f83c7d9d7fff726a7748a6ec180c1f26a1e388aaa35cb8`
GROUP BY part_of_week;

--Part of week - weekend or weekday
```

Press Alt+F1 for Accessibility Optio

Query results

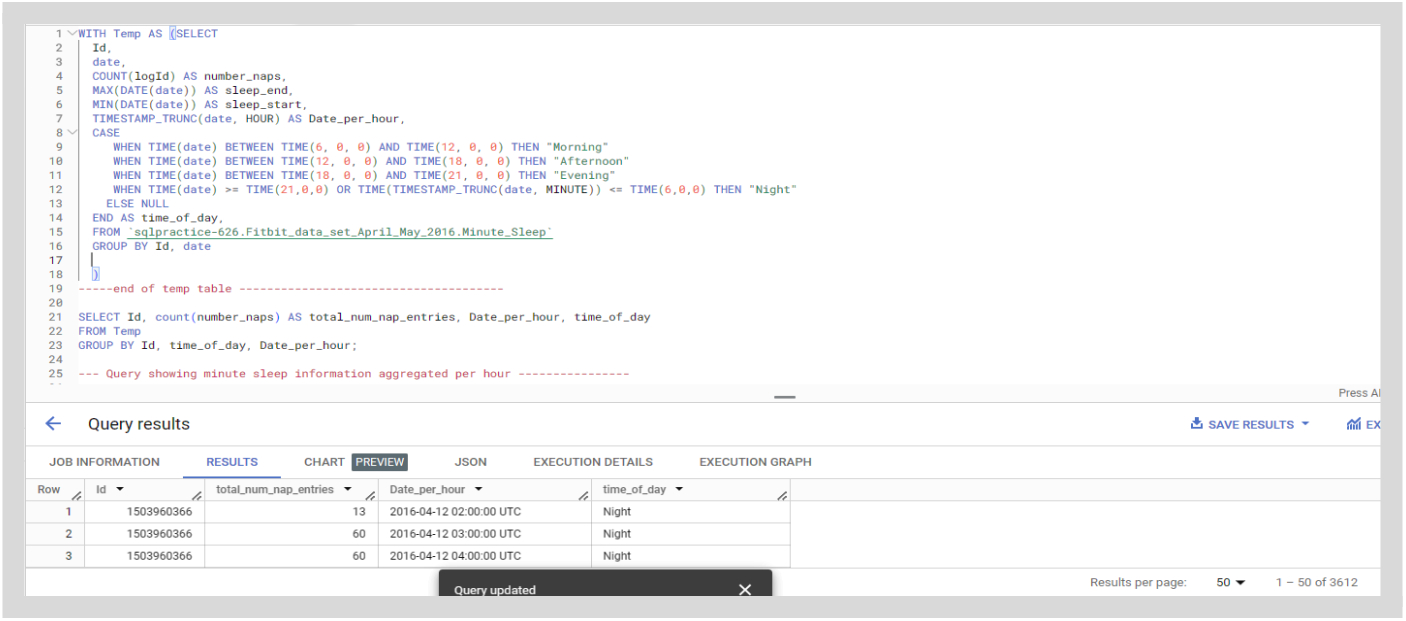
SAVE RESULTS | EXPLORE DATA

JOB INFORMATION | RESULTS | CHART | PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	part_of_week	max_bed_time	max_sleep_time	max_laydown_time	avg_bed_time_min	avg_sleep_time_min	avg_lay_time_min	avg_bed_time_t	avg_sleep_time_t	avg_laydown_time_t
1	Weekday	16:01:00	13:16:00	05:17:00	449.91275167785221	412.99664429530225	36.916107382550337	07:29:00	06:52:00	00:36:00
2	Weekend	16:01:00	12:55:00	06:11:00	481.28571428571428	435.60714285714283	45.678571428571438	08:01:00	07:15:00	00:45:00

- On average, participants spent more time in bed during the **weekends** (481.28 minutes - 08:01:00) as opposed to the **weekdays** (449.91 minutes - 07:29:00).
- On average, participants spent more time sleeping during the **weekends** (435.60 minutes - 07:15:00) as opposed to the **weekdays** (412.99 minutes - 06:52:00)
- On average, participants spent more laying in bed during the **weekends** (45.67 minutes - 00:45:00) as opposed to the **weekdays** (36.91 minutes - 00:36:00)
- Results show participants get more sleep, lay in bed, and spend more in bed during the weekends.
- Somewhere less than 7 hours has been shown to increase health risks like high blood pressure, and obesity (Yong Liu et. al, 2016).

Time of day data:



```
WITH Temp AS (SELECT
  Id,
  date,
  COUNT(logId) AS number_naps,
  MAX(DATE(date)) AS sleep_end,
  MIN(DATE(date)) AS sleep_start,
  TIMESTAMP_TRUNC(date, HOUR) AS Date_per_hour,
  CASE
     WHEN TIME(date) BETWEEN TIME(6, 0, 0) AND TIME(12, 0, 0) THEN "Morning"
     WHEN TIME(date) BETWEEN TIME(12, 0, 0) AND TIME(18, 0, 0) THEN "Afternoon"
     WHEN TIME(date) BETWEEN TIME(18, 0, 0) AND TIME(21, 0, 0) THEN "Evening"
     WHEN TIME(date) >= TIME(21,0,0) OR TIME(TIMESTAMP_TRUNC(date, MINUTE)) <= TIME(6,0,0) THEN "Night"
    ELSE NULL
  END AS time_of_day,
  FROM `sqlpractice-626.Fitbit_data_set_April_May_2016.Minute_Sleep`
  GROUP BY Id, date
  )
-----end of temp table ---------------------------------------

SELECT Id, count(number_naps) AS total_num_nap_entries, Date_per_hour, time_of_day
FROM Temp
GROUP BY Id, time_of_day, Date_per_hour;

--- Query showing minute sleep information aggregated per hour ----------------
```

Query results

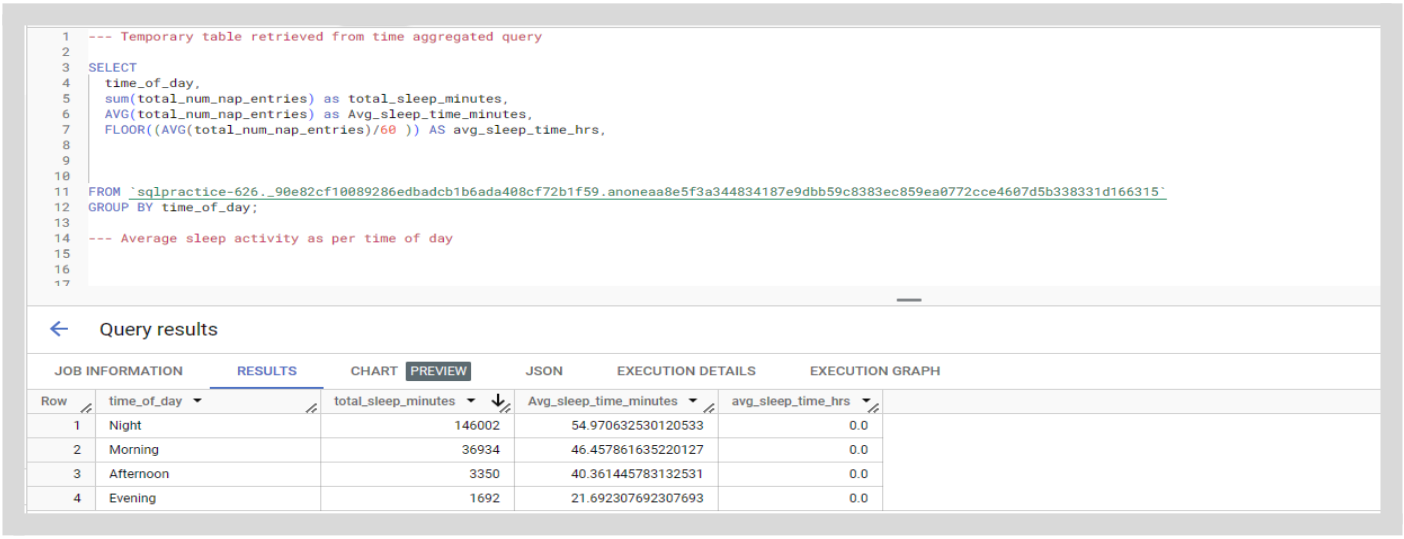
SAVE RESULTS

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Id	total_num_nap_entries	Date_per_hour	time_of_day
1	1503960366	13	2016-04-12 02:00:00 UTC	Night
2	1503960366	60	2016-04-12 03:00:00 UTC	Night
3	1503960366	60	2016-04-12 04:00:00 UTC	Night

Query updated

Results per page: 50 | 1 – 50 of 3612



```
--- Temporary table retrieved from time aggregated query

SELECT
  time_of_day,
  sum(total_num_nap_entries) as total_sleep_minutes,
  AVG(total_num_nap_entries) as Avg_sleep_time_minutes,
  FLOOR((AVG(total_num_nap_entries)/60 )) AS avg_sleep_time_hrs,


FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anoneaa8e5f3a344834187e9dbb59c8383ec859ea0772cce4607d5b338331d166315`
GROUP BY time_of_day;

--- Average sleep activity as per time of day
```

Query results

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	time_of_day	total_sleep_minutes	Avg_sleep_time_minutes	avg_sleep_time_hrs
1	Night	146002	54.970632530120533	0.0
2	Morning	36934	46.457861635220127	0.0
3	Afternoon	3350	40.361445783132531	0.0
4	Evening	1692	21.692307692307693	0.0

- Participants recorded the most time in bed/ sleeping/laying down at **night (9PM to 6AM)** when compared to other times of the day - **Morning(6AM to 12 noon)** , **evening (6PM to 9PM)** and **afternoon (12 noon to 6PM)**. The least time spent in bed/ sleeping/laying down was during the evening.
- At night, participants recorded **146,002 minutes** of sleep time. This was the highest sleep activity compared to other times during the day (morning : 36,934 minutes, afternoon: 3,350 minutes, and evening: 1,692 minutes).
- We can infer participants were asleep at night, and most active during the evening.

Per hour:

```sql
SELECT
  CASE
    WHEN TIME(Date_per_hour) BETWEEN TIME(6, 0, 0) AND TIME(12, 0, 0) THEN "Morning"
    WHEN TIME(Date_per_hour) BETWEEN TIME(12, 0, 0) AND TIME(18, 0, 0) THEN "Afternoon"
    WHEN TIME(Date_per_hour) BETWEEN TIME(18, 0, 0) AND TIME(21, 0, 0) THEN "Evening"
    WHEN TIME(Date_per_hour) >= TIME(21,0,0) OR TIME(TIMESTAMP_TRUNC(Date_per_hour, MINUTE)) <= TIME(6,0,0) THEN "Night"
    ELSE NULL
  END AS time_of_day,
  FORMAT_TIMESTAMP("%H", Date_per_hour) AS hours_of_day,
  sum(total_num_nap_entries) as total_sleep_entries,
  COUNT(DISTINCT Id)AS No_participants_recorded_sleep,
  COUNT(DISTINCT Date_per_hour) AS No_days_sleep_recorded,

FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anon6c29c86709d6a3afc0d88b796cac511b0fa15333dbc5eab8d153dcdccf164da0`
GROUP BY time_of_day, FORMAT_TIMESTAMP("%H", Date_per_hour);


--Sleep activity per hour advanced
```

Query results

SAVE RESULTS

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	time_of_day	hours_of_day	total_sleep_entries	No_participants_recorded_sleep	No_days_sleep_recorded
1	Night	03	22620	21	31
2	Night	04	22069	21	31
3	Night	02	21069	22	31
4	Night	05	20196	21	31
5	Night	01	18760	21	31
6	Night	00	16732	20	31
7	Morning	06	15276	21	31

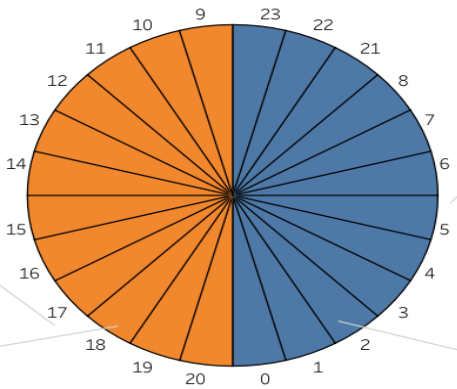
Results per page: 30 1 – 24 of 24

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	time_of_day	hours_of_day	total_sleep_entries	No_participants_recorded_sleep	No_days_sleep_recorded
8	Night	23	13442	19	31
9	Morning	07	9672	21	31
10	Night	22	7649	17	31
11	Morning	08	5945	18	31
12	Evening	21	3493	17	31
13	Morning	09	3346	15	30
14	Morning	10	1858	13	24
15	Evening	20	1081	12	22
16	Morning	11	828	13	19
17	Afternoon	14	707	11	14
18	Afternoon	16	633	8	11
19	Afternoon	15	610	9	11
20	Afternoon	13	553	8	12
21	Morning	12	527	8	12
22	Evening	19	330	4	6
23	Afternoon	17	328	7	9
24	Afternoon	18	254	2	5

Results per page: 30 1 – 24 of 24

Sleep activity clock - Number of days and number of participants sleep activity was recorded per hour



05:00 PM, 06:00 PM and 7:00 PM.
Participants recorded single digit days of sleep activity during these hours.

(April 11th to May 12th, 2016)

09:00 PM to 8:00AM
Participants recorded 31 days of sleep activity within this time frame
(April 11th to May 12th, 2016)

06:00 PM.

Out of 24 participants, only 2 participants recorded sleep activity during these hours.

Making this the lowest turnout for sleep activity.

02:00 AM
22 out of 24 participants recorded sleep activity at 2 AM.
The highest turnout overall

- **09:00 PM** to **8:00 AM**. Participants recorded 31 days of sleep activity within this time frame (April 11th to May 12th, 2016)
- **05:00 PM, 06:00 PM** and **7:00 PM**. Participants recorded single digit days of sleep activity during these hours.
- **02:00 AM**. 22 out of 24 participants recorded sleep activity at 2 AM.The highest turnout overall
- **06:00 PM**. Out of 24 participants, only 2 participants recorded sleep activity during these hours. Making this the lowest turnout for sleep activity.

HEART RATE / STRESS LEVEL ANALYSIS

Monthly data:

```sql
SELECT
  Id,
  Time,
  Value AS Heart_rate_read,
  CASE
      WHEN DATE(Time) BETWEEN "2016-04-01" AND "2016-05-31" THEN "Spring"
    ELSE NULL
  END AS Seasons,
  MAX(DATE(Time)) AS Day_end,
  MIN(DATE(Time)) AS Day_start,
  CASE
      WHEN DATE(Time) BETWEEN "2016-04-12" AND "2016-04-30"  THEN "April"
      WHEN DATE(Time) BETWEEN "2016-05-01" AND "2016-05-31"  THEN "May"
    ELSE NULL
  END AS Months,

  FORMAT_TIMESTAMP("%A", Time) AS day_of_week,

   CASE
    WHEN FORMAT_TIMESTAMP("%A", Time) IN ("Sunday", "Saturday") THEN "Weekend"
    WHEN FORMAT_TIMESTAMP("%A", Time) NOT IN ("Sunday",
    "Saturday") THEN "Weekday"
  ELSE
  "ERROR"
 END
  AS part_of_week,

  CASE
    WHEN TIME(Time) BETWEEN TIME(6, 0, 0) AND TIME(12, 0, 0) THEN "Morning"
    WHEN TIME(Time) BETWEEN TIME(12, 0, 0) AND TIME(18, 0, 0) THEN "Afternoon"
    WHEN TIME(Time) BETWEEN TIME(18, 0, 0) AND TIME(21, 0, 0) THEN "Evening"
    WHEN TIME(Time) >= TIME(21,0,0) OR TIME(TIMESTAMP_TRUNC(Time, MINUTE)) <= TIME(6,0,0) THEN "Night"
   ELSE NULL
  END AS time_of_day,

FROM `sqlpractice-626.Fitbit_data_set_April_May_2016.heartRate_Stress`
GROUP BY Id, Time, Value
```

```sql
---using Temp table from time aggregated query for stress level (sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anonba49ecab15cd8414786672a0e81616bea0b2188a7cb36897095a16ef2ad5e17b)

SELECT
  Months,
  AVG(Heart_rate_read) AS avg_heart_rate_sec,
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anonba49ecab15cd8414786672a0e81616bea0b2188a7cb36897095a16ef2ad5e17b`
GROUP BY Months;


-- Heart rate / month
```

Query results

SAVE RESULTS EXPLORE DA

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	Months	avg_heart_rate_sec
1	April	77.610836939693456
2	May	76.825736614238508

- **April** and **May** months had comparable average heart rates of 77.61 and 76.82 respectively. *Heart rate is measured in beats per minute (BPM)*

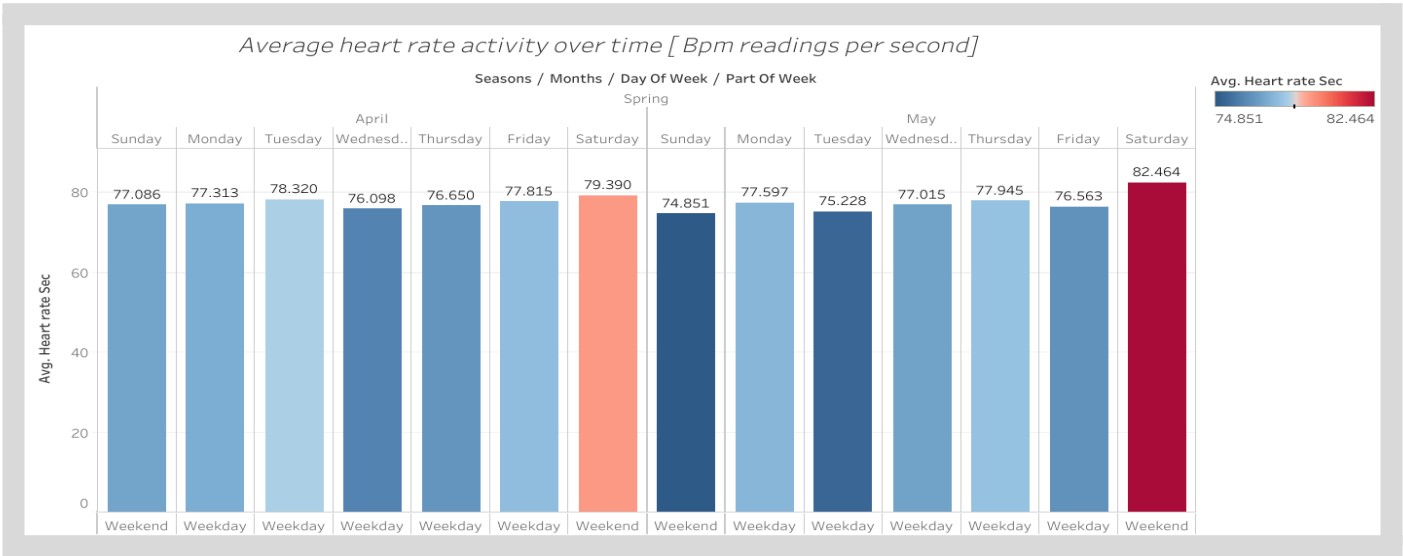
Day of week data:

```sql
SELECT
  day_of_week,
  SUM(Heart_rate_read) AS total_heart_rate,
  AVG(Heart_rate_read) AS avg_heart_rate_sec,
  --DATE(Time) AS Day,
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anonba49ecab15cd8414786672a0e81616bea0b2188a7cb36897095a16ef2ad5e17b`
GROUP BY day_of_week;--DATE(Time),

--Heart rate per day
```

Query results

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	day_of_week	total_heart_rate	avg_heart_rate_sec
1	Saturday	27062659	79.9738145475389
2	Friday	27720986	77.520836479358948
3	Monday	22443478	77.4543352521363
4	Thursday	27667136	77.035901833795066
5	Tuesday	34839545	77.013722945924542
6	Wednesday	30521917	76.4515795327026
7	Sunday	21801637	75.925003569599852



- On **Saturdays**, participants recorded the highest average heart rate per 5-10 seconds with - 79.97. The lowest average was during **Sundays** with a heart rate 75.92 every 5 to 10 seconds. Friday was the highest weekday and came in second with a heart rate of 77.52 every 5 to 10 seconds, while Wednesday was the lowest weekday with a heart rate of 76.45 every 5 to 10 seconds.
- Following the trend of sleep data and physical activity data, Sundays appear to be the day participants rest the most, engage in the least physical activity and thereby have the lowest heart rate value.

Part of week data:

```
74  SELECT
75    part_of_week,
76    SUM(Heart_rate_read) AS total_heart_rate,
77    AVG(Heart_rate_read) AS avg_heart_rate_sec,
78    --DATE(Time) AS Day,
79  FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anonba49ecab15cd8414786672a0e81616bea0b2188a7cb36897095a16ef2ad5e17b`
80  GROUP BY part_of_week;--DATE(Time),
81
82  ----Heart rate per part of week
83
84
85
86
```

Query results

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	part_of_week	total_heart_rate	avg_heart_rate_sec
1	Weekday	143193062	77.0635336741447
2	Weekend	48864296	78.115257033512

- The **weekends** were observed to have the highest average heart rate of 78.11 every 5 to 10 seconds. **Weekdays** were the least with 77.06 every 5 to 10 seconds.

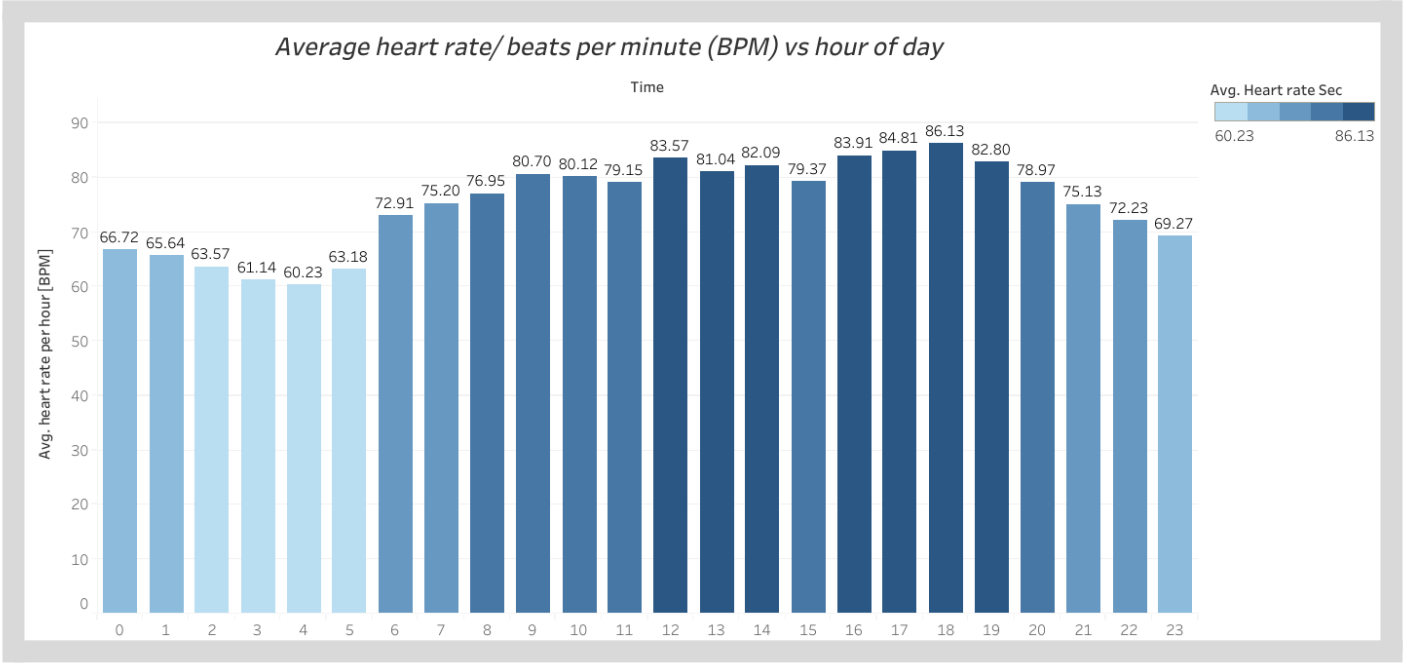
Time of day data:

```sql
SELECT
  time_of_day,
  SUM(Heart_rate_read) AS total_heart_rate,
  AVG(Heart_rate_read) AS avg_heart_rate_sec,
  --DATE(Time) AS Day,
FROM `sqlpractice-626._90e82cf10089286edbadcb1b6ada408cf72b1f59.anonba49ecab15cd8414786672a0e81616bea0b2188a7cb36897095a16ef2ad5e17b`
GROUP BY time_of_day;--DATE(Time),

---Heart rate per time of day
```

Query results

JOB INFORMATION | RESULTS | CHART PREVIEW | JSON | EXECUTION DETAILS | EXECUTION GRAPH

Row	time_of_day	total_heart_rate	avg_heart_rate_sec
1	Evening	30369623	82.885847004528131
2	Afternoon	65769032	82.491865457387362
3	Morning	53297501	77.951207276567317
4	Night	42621202	66.988452597811261

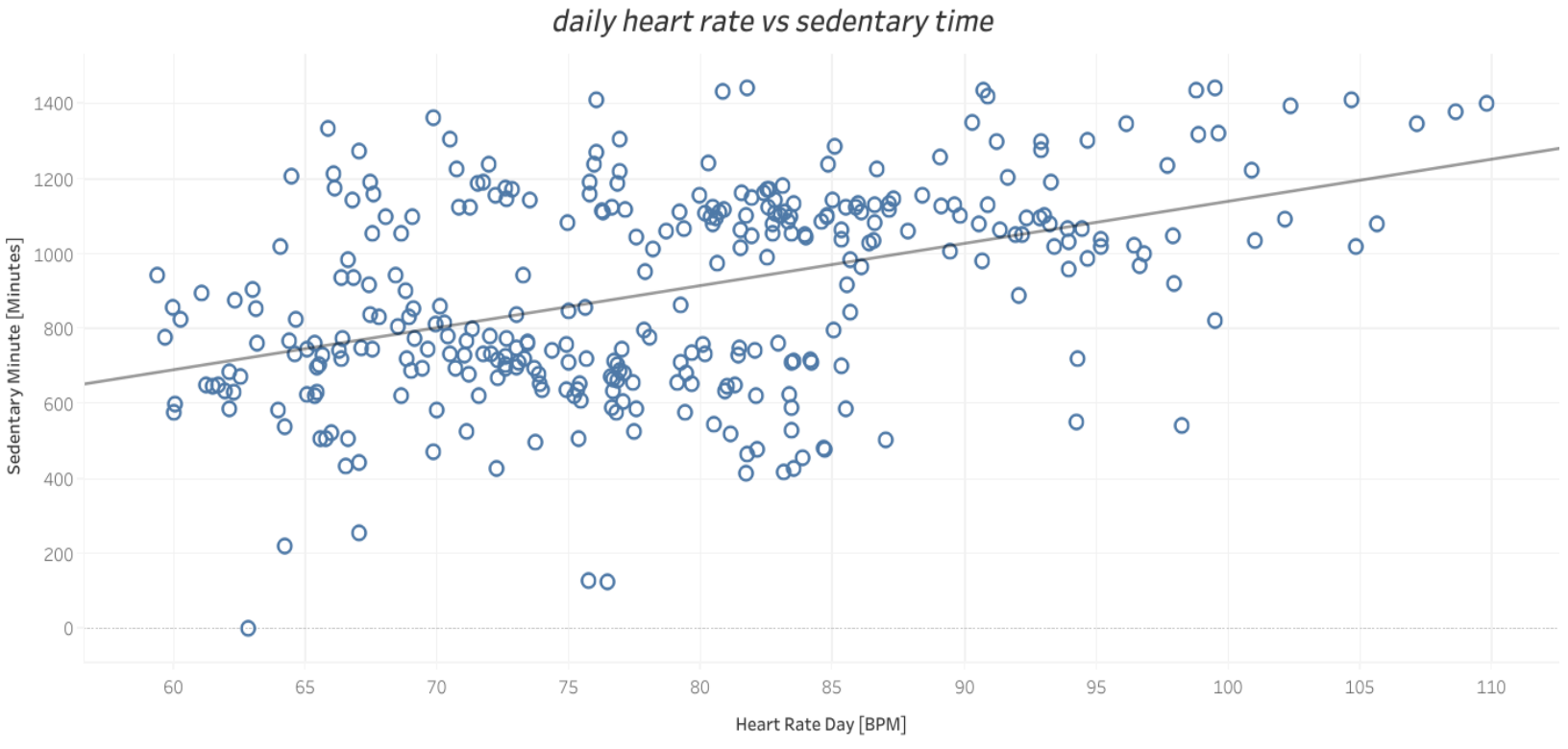


- From 6 to 9 pm (**Evenings**), participants recorded the highest average heart rate every 5 to 10 seconds - 82.88. The least heart rate activity was during the hours of 9PM to 6AM (**Night**) - 66.98.
- 12 noon to 6pm (**Afternoon**) saw the 2nd most heart rate activity with 82.49.
- Both **evenings** and **afternoons** are comparable as there is a difference of .39.
- Morning (6AM to 12 noon) was 3 with an average heart of 77.95 every 5 to 10 seconds.
- Low heart rate at night, but it picks up in the morning (6:00AM) and peaks at 12:00PM noon (83.57) BPM. Heart rate drops in the early afternoon, then picks up 4:00PM and peaks at 6:00PM (86.13)BPM. It gradually drops throughout the night.

Is stress level/heart rate related to sedentary time, calorie expenditure, step count, or physical activity?

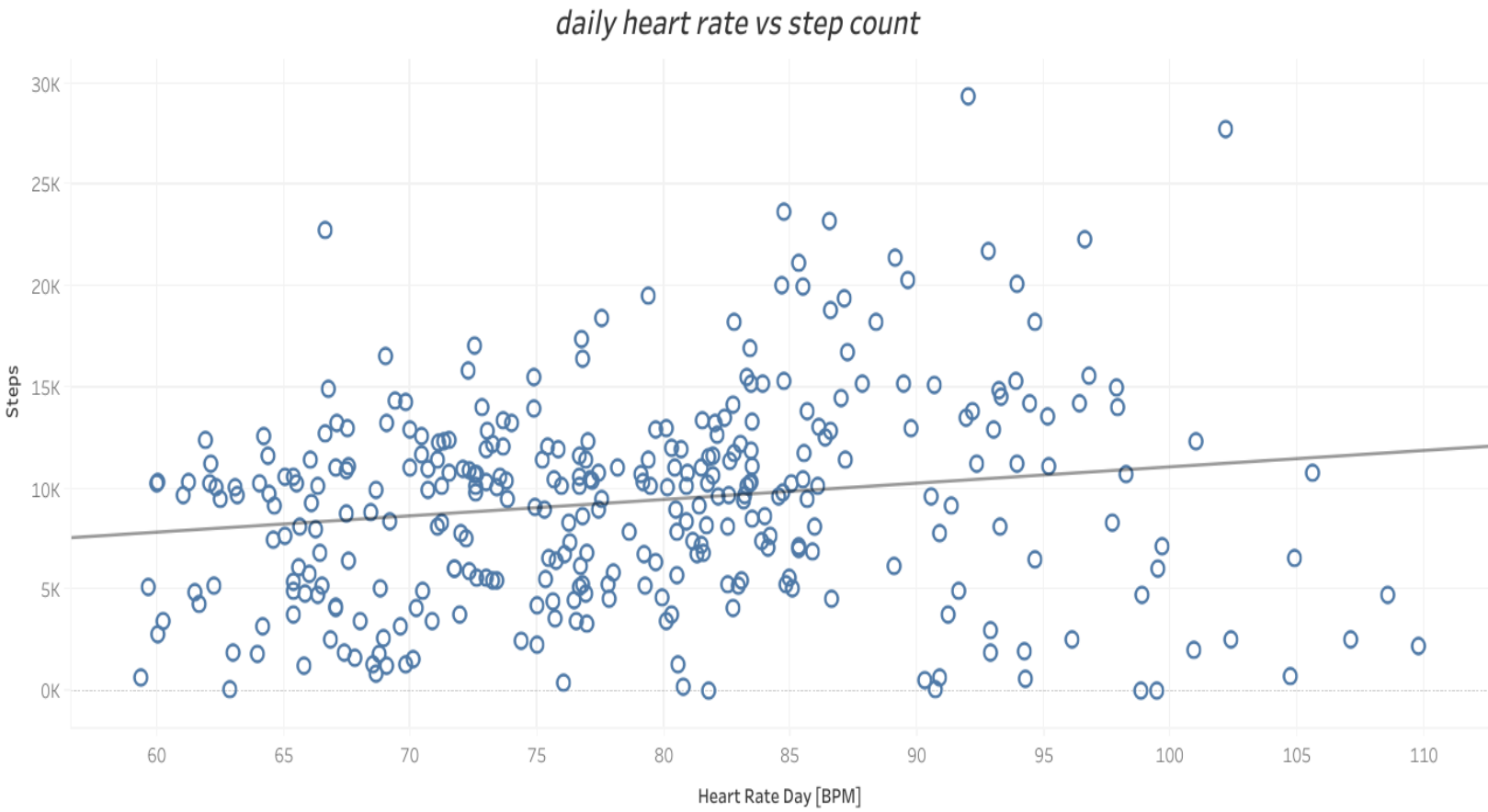
correlation coefficient

* 0.4384



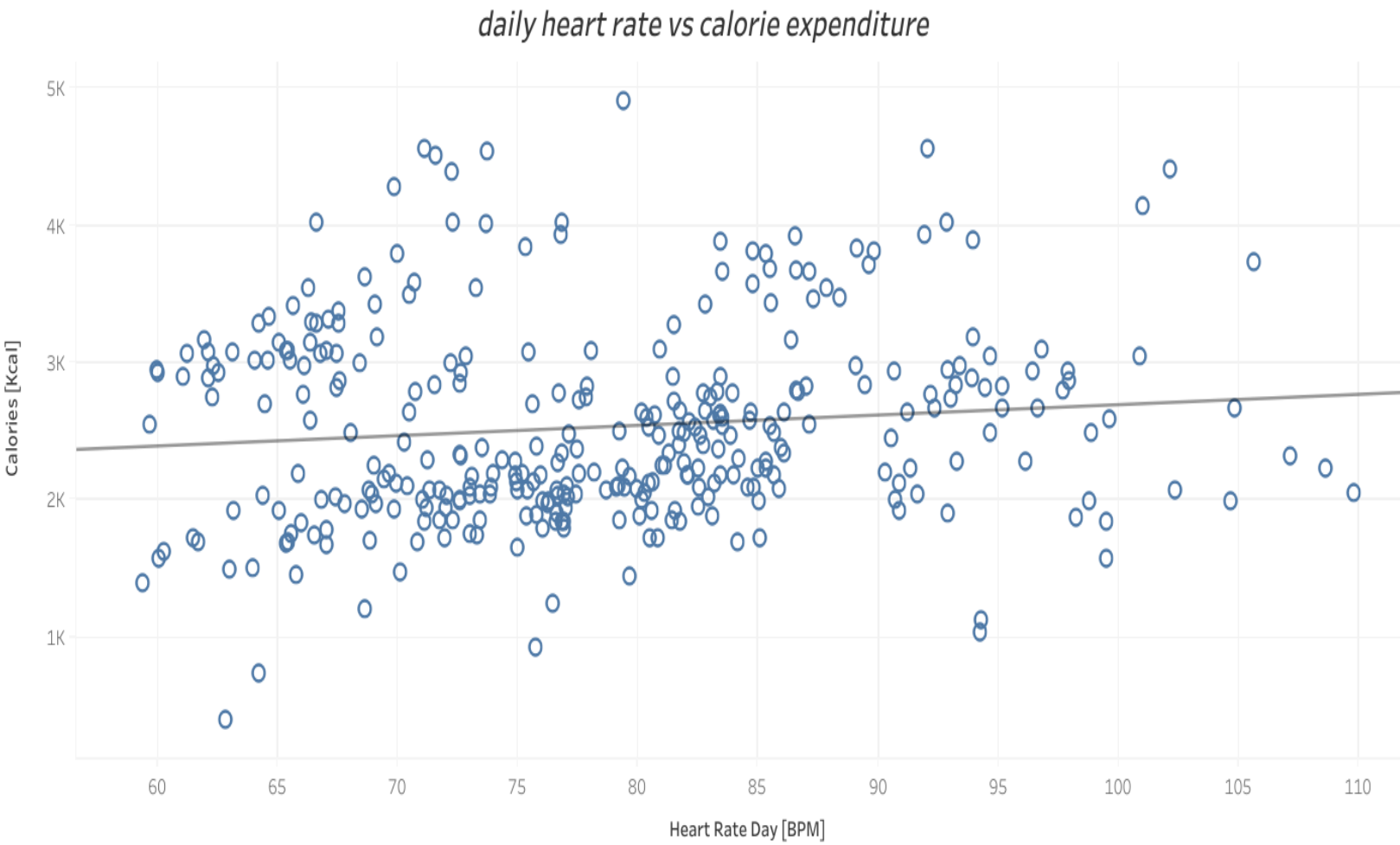
correlation coefficient

*0.1688



correlation coefficient

*0.1108



correlation coefficient

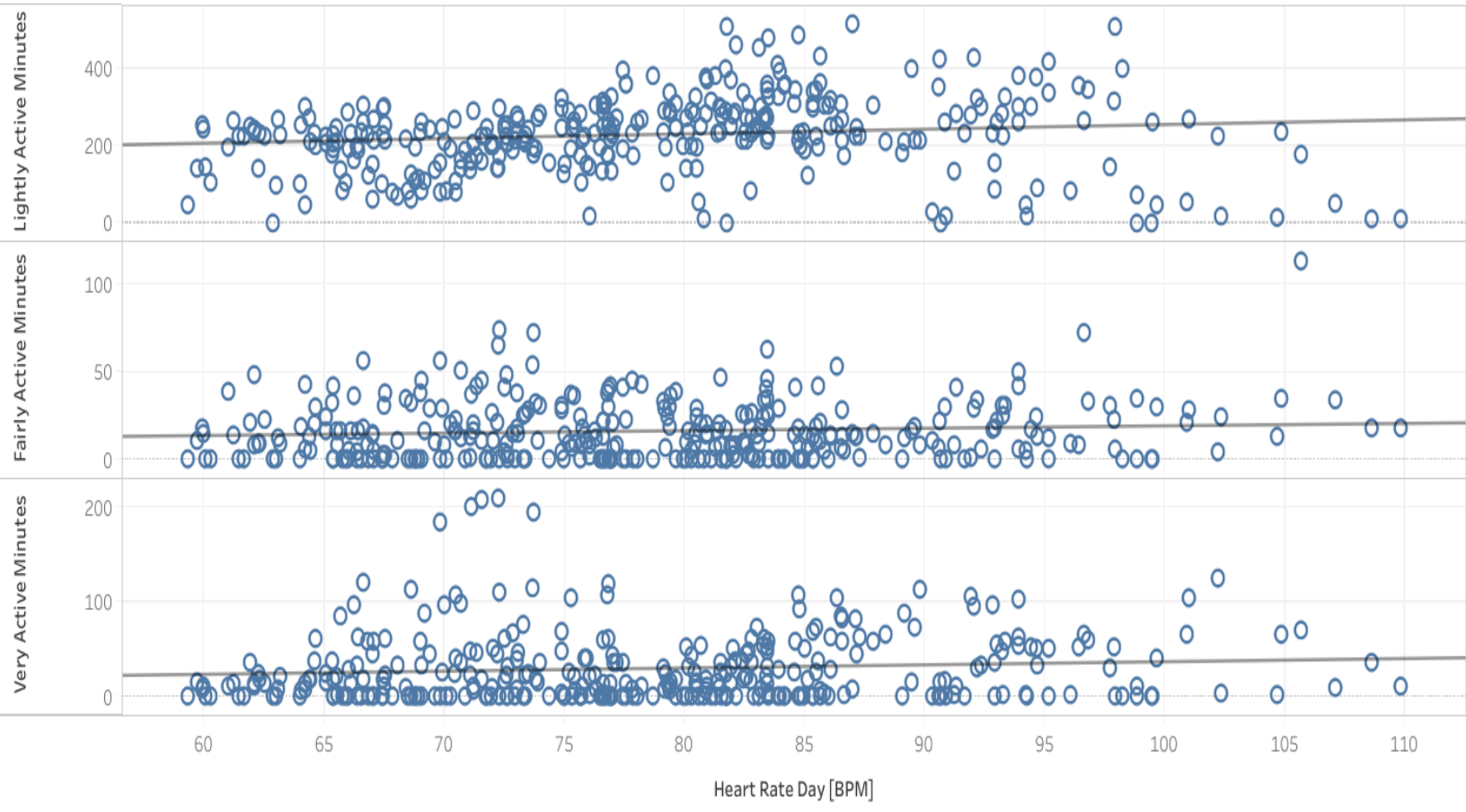
Light: *0.08869

Fair: *0.09494

Vigorous: *0.1297

Since p-value was greater than 0.05, we cannot reject null hypthesis (there no relation between both factors).

daily heart rate vs 3 categories of physical activities



- Our graphs show no correlation between heart rate activity and fairly/vigorous physical activities.
- There was a positive correlation between heart rate and sedimentary time/steps/calorie expenditure/light physical activities. However, the correlation coefficient amongst all 4 factors was less than 0.5.
- We can infer a positive relation exists between heart rate and the 4 factors (sedentary,steps, calories, physical activities - light,fair, vigorous), but that relation is weak.

Key Findings

- Our data shows participants averaged anywhere from 8 to 10k steps a day, and just over 3k calories burned daily.
- Participants were still/idle/or sitting for 3 to 5 hours a day.
- Participants were observed to be more physically active, slept longer, and had higher heart rates in the weekend, as opposed to the weekend days.
- The summer season and summer months **(June, July, August)** saw participants take the most time engaged in steps taken and calories burned. **Winter** (**January and February**) season was the least active month in terms of steps taken and calories burned.
- **Saturdays** experienced the most activity for step count and calories burned. **Sundays** were the day participants were least active in steps taken and calories burned.
- The **afternoons** totaled and averaged the most activity for step count and calories burned. During the **night**, participants did not engage as much in steps taken and calories burned.
- The most idle time spent by participants was during the **spring** season (March, April, May). Specifically, Wednesdays.
- There is a possitive correlation between steps taken and calories, all 3 kinds of physical activities and step count, and all 3 kinds of physical activities and calories burned.
- Throughout the course of a day (24 hours), step count and calorie expenditure peaked at 12:00 PM noon and 6:00 PM.

- **Summer** saw participants engage in light and vigorous physical activities the most, whereas **winter** was least active with these kinds of

activities. **Spring** was the most active month for fairly physical activities with **fall** being the least active month.

- Participants averaged close to **90** minutes engaged in light physical activities, close to **30** minutes doing vigorous physical activities, and around **15 to 20** minutes doing fair physical activities.
- In **July**, participants averaged the highest time during all 3 categories of physical activities.
- Participants seemed to engage in more physical activities during **summer and spring** months.
- In **January**, participants spent the least amount of time on average during all 3 categories of physical activities. **Winter** months were amongst the lowest in minutes spent during all 3 categories of physical activities.
- We can infer participants spend more time doing physical activities during the summer (especially **July**) and spring, and the least during winter - specifically in January.
- Participants averaged the most time doing all 3 categories of physical activities on **Saturdays**, and the least time spent was on **Fridays**.
- There was also positive correlation between the step count and all 3 types (light, fairly, and very) of physical activities.
- The higher in difficulty the physical activity, the higher in correlation there was to step count and calories burned.

- When comparing time in bed, lay down time, and actual sleep time, participants averaged less than **8 hours** of sleep. Participants were seen to spend on average **30** minutes (laydown time) laying down before sleep time. ***Sleep time was just under 7 hours daily***.
- Across all 3 categories of sleep (sleep time, time in bed, and laydown time), participants averaged the most minutes on **Sundays** and the least on **Thursdays**.

- Participants recorded the most time in bed/ sleeping/laying down at **Night**, and least time in the evenings.
- Majority of sleep activity occurred from 9:00 PM to 8:00 AM. A small number of participants spent time sleeping at 5:00PM, 6:00 PM and 7:00 PM.

- Heart rate levels were the highest on **Saturdays**, and the lowest on **Sundays**.
- Heart rate activity was highest during the **evenings** (6:00 PM to 9:00 PM). At **night**, heart rate activity was the lowest.
- Analysis on step count and calorie expenditure shows participants are more engaged in the **afternoons** (12:00 noon to 6:00 PM) and **evenings**(6:00PM to 9:00PM).
- Throughout the course of a day (24 hours), stress levels/heart rate peaked at 12:00 PM noon and 6:00 PM. This was a similar trend in step count and calorie expenditure.
- Although Sundays had the highest sleep minutes and lowest heart rate values, there was no correlation between sleep minutes and heart rate.
- Although sleep activity was lowest in the evenings and heart rate was highest then, there was no correlation between sleep activity and heart rate.
- Similar to calorie expenditure and step count,there are 2 peak hours (12:00 noon and 6:00PM) of heart rate.

Recommendations

1. The leaf fitness tracker alongside the bellabeat app could be used in conjunction to educate customers on common health related practices using our findings. They can then use this data to create and meet their health goals.
2. The leaf can be applied on the wrist as a bracelet, on the neck as a necklace,and on hair or clothing as a clip. Not only does it serve as a fitness and health tracker, we can market its stylish design to our fashion conscious audience.
3. Based on physical, sedentary, and sleep activity, our target audience should be towards working class individuals who work during weekdays. Data shows people with this kind of lifestyle (stand/idle for an average of 3 to 5 hours a day, exhibit high sleep and physical activity during the weekends) purchase/own smart devices to track fitness
4. We should advertise the leaf during the Summer (June, July, August) period as this was when participants engaged in the most step count, calorie expenditure, and physical activity. Saturdays are when participants tend to be most active and Sundays are when sleep activity is highest, so we should advertise the leaf fitness tracker on the weekends.
5. For current users, I recommend we create fitness communities using the leaf fitness tracker and app. Users can share and gain knowledge of health goals and activities with others. Weekly challenges and games could encourage current users to increase usage of the leaf.

   To expand on the 5th recommendation, we can do the following:

6. For weekly or monthly challenges, we can set interval points throughout the day at 12 noon and 6pm (data shows this is when

participants peak in heart rate/calorie expenditure/ step count), and on the weekends (Saturdays are when participants tend to be most active and Sundays are when sleep activity is highest)

Limitations:

- The sample size for stress level and sleep activity was low - there were only 14 unique IDs available for the analysis.
- Some excel files had daily readings for 31 days, while others had daily recordings for 2 years. There will be inconsistencies with the overall results.
- I noted earlier some participants did not wear their tracker for the entirety of the study, hence we had some zero value entries.
- Although heart rate was high in the evenings, our analysis cannot infer it is a result of solely physical activities, step count or calorie loss. Other factors not accounted for could be the issue.
-

References

Relevant Paper:

1. Fanaee-T, Hadi, and Gama, Joao, 'Event labeling combining ensemble detectors and background knowledge', Progress in Artificial Intelligence (2013): pp. 1-15, Springer Berlin Heidelberg

2. Massoomi MR, Handberg EM. Increasing and Evolving Role of Smart Devices in Modern Medicine. Eur Cardiol. 2019;14(3):181-186. Published 2019 Dec 18. doi:10.15420/ecr.2019.02

3. Science Soft (n.d.). Smart Medical Devices: Software Architecture, Features, and Tech Stack. Science Soft Professional Software Development. https://www.scnsoft.com/healthcare/medical-devices/smart#market-overview

4. *Henriksen, André; Johannessen, Erlend; Hartvigsen, Gunnar; Grimsgaard, Sameline; Hopstock, Laila Arnesdatter, 2021, "Replication Data for: Dataset of Consumer-Based Activity Trackers as a Tool for Physical Activity Monitoring in Epidemiological Studies During the COVID-19 Pandemic", https://doi.org/10.18710/TGGCSZ, DataverseNO, V3*

5. *Liu, Yong, et al. "Prevalence of healthy sleep duration among adults—United States, 2014." Morbidity and Mortality Weekly Report 65.6 (2016): 137-141. (sleep notes)*