Case Study 2: How Can a Wellness Technology Company Play It Smart?

Coursera // Google Data Analytics Professional Certificate

Author: Eva Gordos, 2023

Certificate:
https://www.credly.com/badges/d758f87c-a3f1-4a02-9fd3-6a31818824a3/linked_in?t=rzfwl6

TASK OVERVIEW

Produce a report with the following deliverables:

1. A clear summary of the business task
2. A description of all data sources used
3. Documentation of any cleaning or manipulation of data
4. A summary of the analysis
5. Supporting visualizations and key findings
6. Top high-level content recommendations based on the analysis

phase 1 // ask

COMPANY BACKGROUND

Bellabeat is a high-tech manufacturer of health-focused products for women; a successful small company with the potential to become a larger player in the global smart device market.

MAIN QUESTIONS

1. What are some trends in smart device usage?
2. How could these trends apply to Bellabeat customers?
3. How could these trends help influence Bellabeat marketing strategy?

BUSINESS TASK

Define the most important trends in smart device usage and based on the findings, provide high-level recommendations to Bellabeat`s marketing strategy.

KEY STAKEHOLDERS

- Urška Sršen: Bellabeat's cofounder and Chief Creative Officer

- Sando Mur: Mathematician and Bellabeat's cofounder; key member of the Bellabeat executive team
- Bellabeat marketing analytics team

phase 2 // prepare

DATA SOURCE

FitBit Fitness Tracker Data (`No Copyright`; CC0: Public Domain, dataset made available through Mobius). This is the copy of a dataset based on this study. The description of the data is available here.

I downloaded the dataset to my desktop and renamed the files, so that they are more readable for me:

- dailyActivity_merged → daily_activity
- dailyCalories_merged → daily_calories
- dailyIntensities_merged → daily_intensities
- dailySteps_merged → daily_steps
- heartrate_seconds_merged → heartrate_seconds
- hourlyCalories_merged → hourly_calories
- hourlyIntensities_merged → hourly_intensities
- hourlySteps_merged → hourly_steps
- minuteCaloriesNarrow_merged → minute_calories_narrow
- minuteCaloriesWide_merged → minute_calories_wide
- minuteIntensitiesNarrow_merged → minute_intensities_narrow
- minuteIntensitiesWide_merged → minute_intensities_wide
- minuteMETsNarrow_merged → minute_METs_narrow
- minuteSleep_merged → minute_sleep
- minuteStepsNarrow_merged → minute_steps_narrow
- minuteStepsWide_merged → minute_steps_wide
- sleepDay_merged → sleep_day
- weightLogInfo_merged → weight_logInfo

phase 3 // process

TOOL USED
SQL // BigQuery: although the data cleaning and analysis were probably more complicated in SQL than in R, I wanted to practice SQL.

DATA INTEGRITY

From data integrity point of view, I considered the following aspects:

- **Data replication compromising data integrity:** I compared the data between 12 April 2016 and 12 May 2016 of the original dataset with FitBit Fitness Tracker Data

and I found that they are 100% identical, so the data is not compromised by this replication.

- **Data transfer compromising data integrity:** While uploading the files to SQL there was a problem with the Date/ActivityHour/ActivityMinute columns in 14 cases, so I had to change the format type to string in order to be able to import them. This issue was later resolved while cleaning the data (the columns have been set to DATETIME format).

- **Data manipulation compromising data integrity:** I carried out data cleaning, manipulation and analysis with the utmost care in order to avoid any compromise.

DATA CLEANING & MANIPULATION DOCUMENTATION

```
-- Column `Id` can be found in every file of the dataset; Checking if all Ids
consist of 10 characters; Query executed for all files, all Ids consist of 10
characters.

SELECT
 LENGTH(CAST(Id AS STRING))
FROM `tribal-marker-389314.daily_activity.daily_activity`
WHERE LENGTH(CAST(Id AS STRING)) != 10;

-- Transforming Ids into string format (in case of daily_activity,
daily_calories, daily_intensities and daily_steps tables)

SELECT DISTINCT(CAST(Id AS STRING)) as distinct_id
FROM `tribal-marker-389314.daily_activity.daily_activity`
ORDER BY distinct_id;

-- Transforming Time/Date/ActivityHour/ActivityMinute into DATETIME format (in
case of 14 files, except daily_activity, daily_calories, daily_intensities and
daily_steps tables)

SELECT
PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p', Time) AS date_time
FROM `tribal-marker-389314.heartrate_seconds.heartrate_seconds`

-- Verifying if `daily_activity` dataset consists of data in `daily_calories`,
`daily_intensities` and `daily_steps; Result: yes.

SELECT
  daily_activity.Id
  ActivityDate, ActivityDay,
  daily_activity.Calories,
  daily_calories.Calories,
  daily_activity.Calories-daily_calories.Calories AS diff_cal
FROM daily_activity.daily_activity
JOIN
  daily_calories.daily_calories ON daily_activity.Id = daily_calories.Id AND
daily_activity.ActivityDate = daily_calories.ActivityDay
WHERE daily_activity.Calories-daily_calories.Calories !=0;

SELECT
  daily_activity.Id,
```

```sql
  ActivityDate,
  ActivityDay,
  daily_activity.SedentaryMinutes,
  daily_intensities.SedentaryMinutes,
  daily_activity.LightlyActiveMinutes,
  daily_intensities.LightlyActiveMinutes,
  daily_activity.FairlyActiveMinutes,
  daily_intensities.FairlyActiveMinutes,
  daily_activity.VeryActiveMinutes,
  daily_intensities.VeryActiveMinutes,
  daily_activity.SedentaryActiveDistance,
  daily_intensities.SedentaryActiveDistance,
  daily_activity.LightActiveDistance,
  daily_intensities.LightActiveDistance,
  daily_activity.ModeratelyActiveDistance,
  daily_intensities.ModeratelyActiveDistance,
  daily_activity.VeryActiveDistance,
  daily_intensities.VeryActiveDistance,
  daily_activity.SedentaryMinutes-daily_intensities.SedentaryMinutes AS
diff_SM,
  daily_activity.LightlyActiveMinutes-daily_intensities.LightlyActiveMinutes
AS diff_LAM,
  daily_activity.FairlyActiveMinutes-daily_intensities.FairlyActiveMinutes AS
diff_FAM,
  daily_activity.VeryActiveMinutes-daily_intensities.VeryActiveMinutes AS
diff_VAM,
  daily_activity.SedentaryActiveDistance-daily_intensities.SedentaryActiveDi
stance AS diff_SAD,
  daily_activity.LightActiveDistance-daily_intensities.LightActiveDistance AS
diff_LAD,
  daily_activity.ModeratelyActiveDistance-daily_intensities.ModeratelyActive
Distance AS diff_MAD,
  daily_activity.VeryActiveDistance-daily_intensities.VeryActiveDistance AS
diff_VAD
FROM daily_activity.daily_activity
JOIN  daily_intensities.daily_intensities ON daily_activity.Id
=  daily_intensities.Id AND daily_activity.ActivityDate
=  daily_intensities.ActivityDay
WHERE
  daily_activity.SedentaryMinutes-daily_intensities.SedentaryMinutes !=0
  oR
  daily_activity.LightlyActiveMinutes-daily_intensities.LightlyActiveMinutes
!=0
  OR
  daily_activity.FairlyActiveMinutes-daily_intensities.FairlyActiveMinutes
!=0
  oR
  daily_activity.VeryActiveMinutes-daily_intensities.VeryActiveMinutes !=0
  OR
  daily_activity.SedentaryActiveDistance-daily_intensities.SedentaryActiveDi
stance !=0
  OR
  daily_activity.LightActiveDistance-daily_intensities.LightActiveDistance
!=0
  OR
  daily_activity.ModeratelyActiveDistance-daily_intensities.ModeratelyActive
Distance !=0
  OR
```

```
  daily_activity.VeryActiveDistance-daily_intensities.VeryActiveDistance !=0;

SELECT
  daily_activity.Id,
  ActivityDate,
  ActivityDay,
  daily_activity.TotalSteps,
  daily_steps.StepTotal,
  daily_activity.TotalSteps-daily_steps.StepTotal AS diff_step
FROM daily_activity.daily_activity
JOIN daily_steps.daily_steps ON daily_activity.Id = daily_steps.Id AND
daily_activity.ActivityDate = daily_steps.ActivityDay
WHERE daily_activity.TotalSteps-daily_steps.StepTotal !=0;

-- Verifying if `TotalDistance` or `TrackerDistance` equals to the sum of every
Distance type: 636 results are not equal in both cases, but there is slight
difference. In case of TotalDistance, there are 13 results where the difference
is more than 1. The maximum difference is 9,37 km. In case of TrackerDistance,
there are 20 results where the difference is more than 1. The maximum difference
is 9,37 km.

SELECT
 ABS(TotalDistance - (SedentaryActiveDistance + LightActiveDistance +
ModeratelyActiveDistance + VeryActiveDistance)) as diff_dist
FROM daily_activity.daily_activity
WHERE ABS(TotalDistance - (SedentaryActiveDistance + LightActiveDistance +
ModeratelyActiveDistance + VeryActiveDistance)) != 0
ORDER BY ABS(TotalDistance - (SedentaryActiveDistance + LightActiveDistance +
ModeratelyActiveDistance + VeryActiveDistance)) DESC;

SELECT
 ABS(TrackerDistance - (SedentaryActiveDistance + LightActiveDistance +
ModeratelyActiveDistance + VeryActiveDistance)) as diff_dist
FROM daily_activity.daily_activity
WHERE ABS(TrackerDistance - (SedentaryActiveDistance + LightActiveDistance +
ModeratelyActiveDistance + VeryActiveDistance)) != 0
ORDER BY ABS(TrackerDistance - (SedentaryActiveDistance + LightActiveDistance
+ ModeratelyActiveDistance + VeryActiveDistance)) DESC;

-- Verifying Calories; There are 12 cases where Calories are less than 1000.

SELECT
  Id,
  ActivityDate,
  Calories,
  FROM `tribal-marker-389314.daily_activity.daily_activity`
WHERE LENGTH(CAST(Calories AS STRING)) != 4;

-- Verifying TotalSteps; There are 77 cases where TotalSteps are 0.

SELECT
  Id,
  ActivityDate,
  TotalSteps,
FROM `tribal-marker-389314.daily_activity.daily_activity`
WHERE TotalSteps = 0;

-- Verifying TotalDistance; There are 78 cases where TotalDistance is 0.
```

```sql
SELECT
  Id,
  ActivityDate,
  TotalDistance,
FROM `tribal-marker-389314.daily_activity.daily_activity`
WHERE TotalDistance = 0;

-- Verifying TrackerDistance; There are 78 cases where TrackerDistance is 0.

SELECT
  Id,
  ActivityDate,
  TrackerDistance,
FROM `tribal-marker-389314.daily_activity.daily_activity`
WHERE TrackerDistance = 0;

-- Checking data where TotalSteps, TotalDistance or TrackerDistance equal 0;
There are 78 cases.

SELECT
  *
FROM `tribal-marker-389314.daily_activity.daily_activity`
WHERE TrackerDistance = 0 OR TotalDistance = 0 OR TotalSteps = 0;

-- Verifying consistency between daily, hourly and minute data

-- 1 daily_calories-hourly_calories; Less than 3000 calories difference per day.

SELECT hc.ActivityDate, ABS(hc.sum_cal - da.sum_cal) AS diff_da_hc
FROM
  (SELECT SUM(Calories) AS sum_cal, ActivityDate
  FROM `tribal-marker-389314.daily_activity.daily_activity`
  GROUP BY ActivityDate) as da
JOIN
  (SELECT SUM(Calories) AS sum_cal, DATE(PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p',
ActivityHour)) AS ActivityDate
  FROM `tribal-marker-389314.hourly_calories.hourly_calories`
  GROUP BY ActivityDate) AS hc
ON da.ActivityDate = hc.ActivityDate
ORDER BY diff_da_hc DESC;

-- 2 daily_steps-hourly_steps; In certain cases there is considerable difference
(max.17945).

SELECT hs.ActivityDate, ABS(hs.sum_step - da.sum_step) AS diff_da_hs
FROM
  (SELECT SUM(TotalSteps) AS sum_step, ActivityDate
  FROM `tribal-marker-389314.daily_activity.daily_activity`
  GROUP BY ActivityDate) as da
JOIN
  (SELECT SUM(StepTotal) AS sum_step, DATE(PARSE_DATETIME('%m/%d/%Y %I:%M:%S
%p', ActivityHour)) AS ActivityDate
  FROM `tribal-marker-389314.hourly_steps.hourly_steps`
  GROUP BY ActivityDate) AS hs
ON da.ActivityDate = hs.ActivityDate
ORDER BY diff_da_hs DESC;
```

```sql
-- 3 daily_calories-minute_calories; Less than 3000 calories difference per day.

SELECT mc.ActivityDate, ABS(mc.sum_cal - da.sum_cal) AS diff_da_mc
FROM
  (SELECT SUM(Calories) AS sum_cal, ActivityDate
  FROM `tribal-marker-389314.daily_activity.daily_activity`
  GROUP BY ActivityDate) as da
JOIN
  (SELECT SUM(Calories) AS sum_cal, DATE(PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p',
ActivityMinute)) AS ActivityDate
  FROM `tribal-marker-389314.minute_calories_narrow.minute_calories_narrow`
  GROUP BY ActivityDate) AS mc
ON da.ActivityDate = mc.ActivityDate
ORDER BY diff_da_mc DESC;

-- 4 daily_steps-minute_steps; In certain cases there is considerable difference
(max. 17945).

SELECT ms.ActivityDate, ABS(ms.sum_step - da.sum_step) AS diff_da_ms
FROM
  (SELECT SUM(TotalSteps) AS sum_step, ActivityDate
  FROM `tribal-marker-389314.daily_activity.daily_activity`
  GROUP BY ActivityDate) as da
JOIN
  (SELECT SUM(Steps) AS sum_step, DATE(PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p',
ActivityMinute)) AS ActivityDate
  FROM `tribal-marker-389314.minute_steps_narrow.minute_steps_narrow`
  GROUP BY ActivityDate) AS ms
ON da.ActivityDate = ms.ActivityDate
ORDER BY diff_da_ms DESC;

-- 5 hourly_calories-minute_calories; There is some difference in each row (max
311,44).

SELECT mc.ActivityHour, ABS(mc.sum_cal - hc.sum_cal) AS diff_hc_mc
FROM
  (SELECT SUM(Calories) AS sum_cal, PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p',
ActivityHour) AS ActivityHour
  FROM `tribal-marker-389314.hourly_calories.hourly_calories`
  GROUP BY ActivityHour) AS hc
JOIN
  (SELECT SUM(Calories) AS sum_cal, DATETIME_TRUNC(PARSE_DATETIME('%m/%d/%Y
%I:%M:%S %p', ActivityMinute), HOUR) AS ActivityHour
  FROM `tribal-marker-389314.minute_calories_narrow.minute_calories_narrow`
  GROUP BY ActivityHour) AS mc
ON hc.ActivityHour = mc.ActivityHour
ORDER BY diff_hc_mc DESC;

-- 6 hourly_steps-minute-steps; 2016-05-12T01:00:00: 487 steps difference, in
other cases the difference is 0.

SELECT ms.ActivityHour, ABS(ms.sum_step - hs.sum_step) AS diff_hs_ms
FROM
  (SELECT SUM(StepTotal) AS sum_step, PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p',
ActivityHour) AS ActivityHour
  FROM `tribal-marker-389314.hourly_steps.hourly_steps`
  GROUP BY ActivityHour) AS hs
JOIN
```

```sql
  (SELECT SUM(Steps) AS sum_step, DATETIME_TRUNC(PARSE_DATETIME('%m/%d/%Y
%I:%M:%S %p', ActivityMinute), HOUR) AS ActivityHour
  FROM `tribal-marker-389314.minute_steps_narrow.minute_steps_narrow`
  GROUP BY ActivityHour) AS ms
ON hs.ActivityHour = ms.ActivityHour;

-- 7 hourly_intensities-minute_intensities; The difference is 0 in every row.

SELECT mi.ActivityHour, ABS(mi.sum_int - hi.sum_int) AS diff_hi_mi
FROM
  (SELECT SUM(TotalIntensity) AS sum_int, PARSE_DATETIME('%m/%d/%Y %I:%M:%S
%p', ActivityHour) AS ActivityHour
  FROM `tribal-marker-389314.hourly_intensities.hourly_intensities`
  GROUP BY ActivityHour) AS hi
JOIN
  (SELECT SUM(Intensity) AS sum_int, DATETIME_TRUNC(PARSE_DATETIME('%m/%d/%Y
%I:%M:%S %p', ActivityMinute), HOUR) AS ActivityHour
  FROM
`tribal-marker-389314.minute_intensities_narrow.minute_intensities_narrow`
  GROUP BY ActivityHour) AS mi
ON hi.ActivityHour = mi.ActivityHour;

-- Creating `hourly_activity` table by joining `hourly_calories`,
`hourly_intensities` and `hourly_steps` tables

SELECT
  hourly_calories.Id,
  PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p', hourly_calories.ActivityHour) AS
Activity_hour,
  Calories,
  TotalIntensity,
  AverageIntensity,
  StepTotal
FROM hourly_calories.hourly_calories
JOIN hourly_intensities.hourly_intensities ON hourly_calories.Id =
hourly_intensities.Id AND hourly_calories.ActivityHour =
hourly_intensities.ActivityHour
JOIN hourly_steps.hourly_steps ON hourly_calories.Id = hourly_steps.Id AND
hourly_calories.ActivityHour = hourly_steps.ActivityHour;

-- Creating `minute_activity` table by joining `minute_calories_narrow`,
`minute_intensities_narrow` and `minute_steps_narrow` tables

SELECT
  minute_calories_narrow.Id,
  PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p',
minute_calories_narrow.ActivityMinute) AS Activity_min,
  minute_calories_narrow.Calories,
  minute_intensities_narrow.Intensity,
  minute_steps_narrow.Steps
FROM minute_calories_narrow.minute_calories_narrow
JOIN minute_intensities_narrow.minute_intensities_narrow ON
minute_calories_narrow.Id = minute_intensities_narrow.Id AND
minute_calories_narrow.ActivityMinute =
minute_intensities_narrow.ActivityMinute
JOIN minute_steps_narrow.minute_steps_narrow ON minute_calories_narrow.Id =
minute_steps_narrow.Id AND minute_calories_narrow.ActivityMinute =
minute_steps_narrow.ActivityMinute
```

phase 4 // analyze & phase 5 // share

The dataset contains data of max. 33 users for the time period between 2016-04-12 00:00 and 2016-05-12 15:59 (incl.). The data of 2016-05-12 is therefore incomplete, that explains the considerably lower numbers in results in many queries compared to other days of the research period.

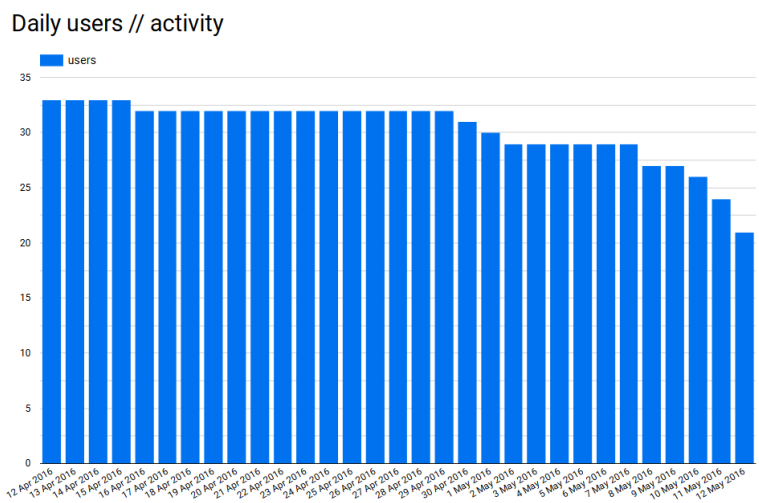
```
SELECT *
FROM `tribal-marker-389314.minute_activity.minute_activity`
ORDER BY Activity_min
LIMIT 100;

SELECT *
FROM `tribal-marker-389314.minute_activity.minute_activity`
ORDER BY Activity_min DESC
LIMIT 100;
```

In my analysis, I have primarily used the daily data and in certain cases the hourly data.

PURPOSE OF SMART DEVICE USAGE

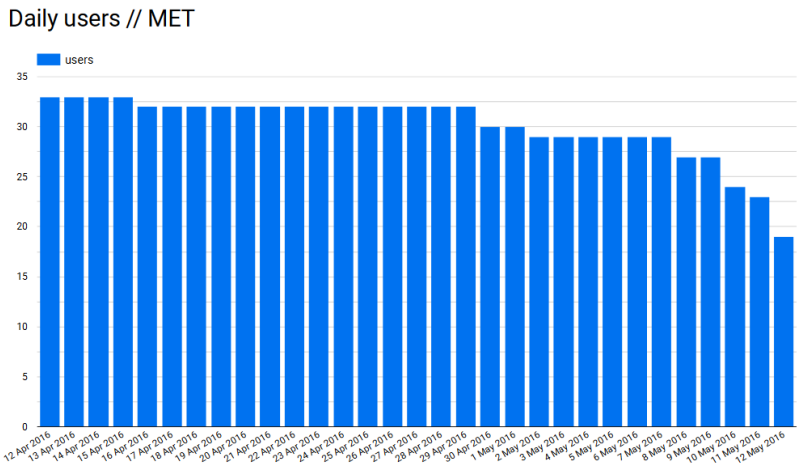
1. **What do customers use the smart device for (activity, MET, weight, sleep or heartbeat recording)?**

```
SELECT
  COUNT(DISTINCT(Id)) as users
FROM `tribal-marker-389314.daily_activity.daily_activity`;

SELECT
  ActivityDate,
  COUNT(DISTINCT(Id)) as users
FROM `tribal-marker-389314.daily_activity.daily_activity`
GROUP BY ActivityDate
ORDER BY ActivityDate;
```



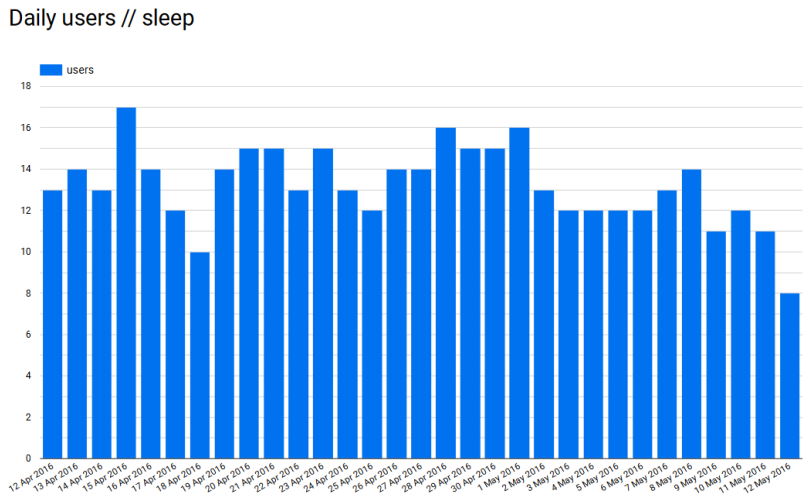
In total, 33 smart device users participated in the research. In the beginning, activity data (intensities, calories and steps) were provided by all participants, but by the end of the research, only 21 participants provided data.

```sql
SELECT COUNT(DISTINCT(Id)) as users
FROM `tribal-marker-389314.minute_METs_narrow.minute_METs_narrow`;

SELECT
  DATE(PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p', ActivityMinute)) AS MET_date,
  COUNT(DISTINCT(Id)) as users
FROM `tribal-marker-389314.minute_METs_narrow.minute_METs_narrow`
GROUP BY MET_date
ORDER BY MET_date;
```



MET data were provided by 33 participants on 12 April 2016. The number of users decreased throughout the research period, 19 participants provided data on 12 May 2016.

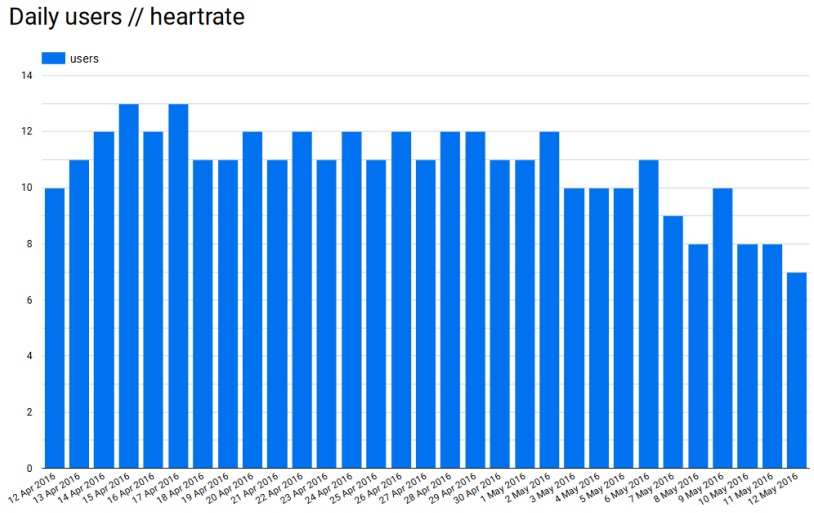
```sql
SELECT
  COUNT(DISTINCT(Id)) as users
FROM `tribal-marker-389314.sleep_day.sleep_day`;

SELECT
  DATE(PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p', SleepDay)) AS date,
  COUNT(DISTINCT(Id)) as users
FROM `tribal-marker-389314.sleep_day.sleep_day`
GROUP BY date
ORDER BY date;
```



Sleep recording was only important for 24 participants (72,7%). The number of daily users varied between 8 and 17.

```sql
SELECT
  COUNT(DISTINCT(Id)) AS users
FROM `tribal-marker-389314.heartrate_seconds.heartrate_seconds`;
```

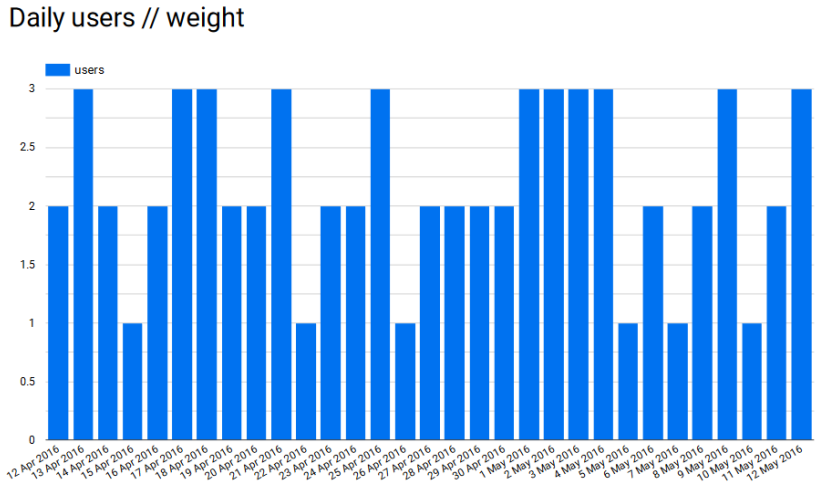
```sql
SELECT
  DATE(PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p', Time)) as heartrate_date,
  COUNT(DISTINCT TRIM(Id)) as users
FROM `tribal-marker-389314.heartrate_seconds.heartrate_seconds`
GROUP BY heartrate_date
ORDER BY heartrate_date;
```



14 participants (42.42%) used Fitbit for heartrate tracking. The daily user number varied between 7 and 13.

```sql
SELECT
  COUNT(DISTINCT(Id)) as users
FROM `tribal-marker-389314.weight_logInfo.weight_logInfo`;
```

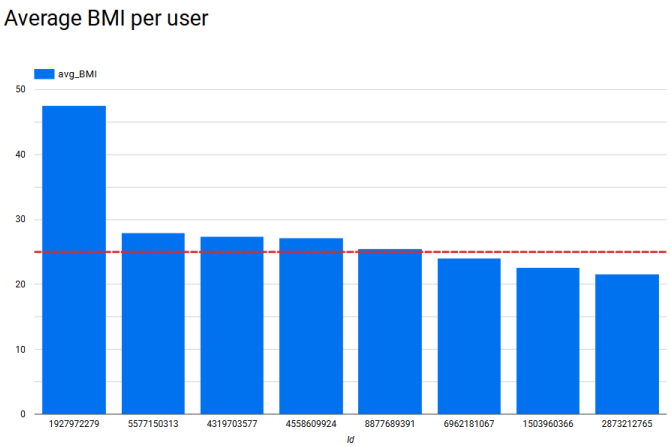
```sql
SELECT
  DATE(PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p', Date)) as weight_date,
  COUNT(DISTINCT TRIM(Id)) as users
FROM `tribal-marker-389314.weight_logInfo.weight_logInfo`
GROUP BY weight_date
ORDER BY weight_date;
```



Weight recording was the least popular function of the smart device. Only 8 participants (24,24%) used Fitbit for weight tracking at some point of the research period. The number of daily users varied between 1 and 3.

```sql
SELECT
  Id,
  AVG(BMI) AS avg_BMI
FROM weight_logInfo.weight_logInfo
```

```
GROUP BY Id;
```



5 out of 8 users of the weight recording function are above the normal BMI range (the red reference line marks 25, the upper limit of the normal BMI range).

```
SELECT AVG(Calories) as avg_cal
FROM `tribal-marker-389314.daily_activity.daily_activity`;
-- avg_cal = 2303.60957446808

SELECT
Id, AVG(Calories) as avg_cal
FROM `tribal-marker-389314.daily_activity.daily_activity`
WHERE Id = 1927972279 OR Id = 4319703577 OR Id = 4558609924 OR Id = 5577150313
OR Id = 8877689391
GROUP BY Id;
```

User ID	Average Calories Burnt
4319703577	2037.677419
5577150313	3359.633333
1927972279	2172.806452
4558609924	2033.258065
8877689391	3420.258065

By further analyzing their behaviour, we can see that the average calories burnt exceeds the overall average calories in case of only 2 users.

```
SELECT
  COUNT(DISTINCT(Id)) as users
FROM `tribal-marker-389314.weight_logInfo.weight_logInfo`
WHERE IsManualReport = false;
```

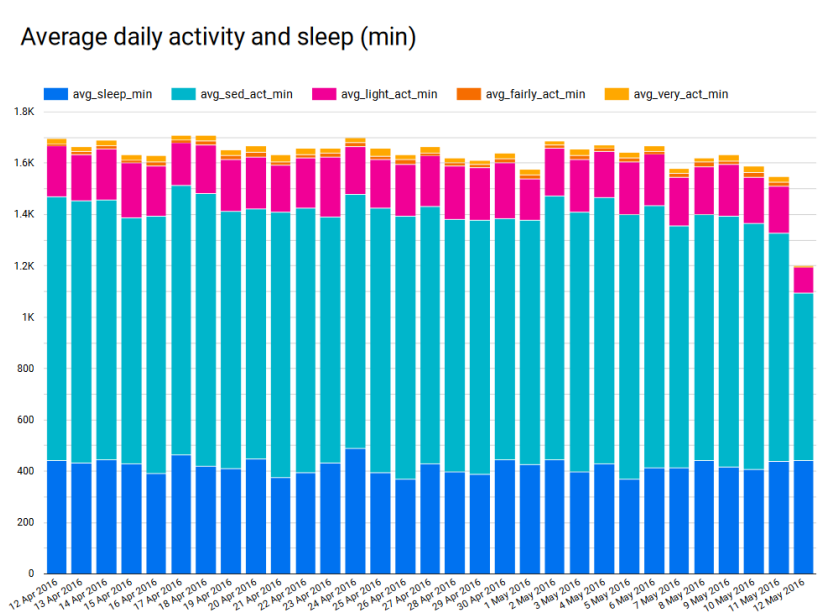
3 users use smart scale and do not track their weight manually.
Based on the above, we can conclude that weight loss is not the primary goal of Fitbit users.

2. How active are Fitbit users?

```
SELECT
  ActivityDate,
  AVG(daily_activity.SedentaryMinutes) AS avg_sed_act_min,
```

```
  AVG(daily_activity.LightlyActiveMinutes) AS  avg_light_act_min,
  AVG(daily_activity.FairlyActiveMinutes) AS avg_fairly_act_min,
  AVG(daily_activity.VeryActiveMinutes) AS avg_very_act_min,
  AVG(sleep_day.TotalMinutesAsleep) AS avg_sleep_min
FROM daily_activity.daily_activity
FULL JOIN sleep_day.sleep_day
ON CAST(daily_activity.Id AS STRING) = sleep_day.Id AND
daily_activity.ActivityDate = DATE(PARSE_DATETIME('%m/%d/%Y %H:%M:%S %p',
SleepDay))
GROUP BY ActivityDate
ORDER BY ActivityDate;

SELECT
  AVG(TotalMinutesAsleep) as avg_sleep_min
FROM sleep_day.sleep_day

SELECT
  AVG(SedentaryMinutes) AS avg_sed_act_min,
  AVG(LightlyActiveMinutes) AS  avg_light_act_min,
  AVG(FairlyActiveMinutes) AS avg_fairly_act_min,
  AVG(VeryActiveMinutes) AS avg_very_act_min
FROM daily_activity.daily_activity;
```

avg_sed_act_min	avg_light_act_min	avg_fairly_act_min	avg_very_act_min	avg_sleep_min
991.2106383	192.812766	13.56489362	21.16489362	419.4673123



On average, Fitbit users spend most of their day by sedentary activity (16,5h) or asleep (7h) (possibly office workers). They spend 21 minutes by very active activity and 14 minutes by fairly active activity.

In certain cases, the sum of the daily averages exceeds 1440 min (24h).

```
SELECT COUNT(*) FROM daily_activity.daily_activity
GROUP BY Id, ActivityDate
HAVING COUNT(*) != 1;
```

```
SELECT COUNT(*) AS sleep_count, SUM(TotalMinutesAsleep) AS sleep_min,
DATE(PARSE_DATETIME('%m/%d/%Y %I:%M:%S %p', SleepDay)) as sleep_date, Id FROM
sleep_day.sleep_day
GROUP BY Id, sleep_date
HAVING COUNT(*) != 1;
```

sleep_count	sleep_min	sleep_date	Id
2	942	2016-05-05	4388161847
2	1040	2016-05-07	4702921684
2	776	2016-04-25	8378563200

```
SELECT (SedentaryMinutes + LightlyActiveMinutes + FairlyActiveMinutes +
VeryActiveMinutes) + 1040
FROM daily_activity.daily_activity
WHERE Id = 4702921684 AND ActivityDate = '2016-05-07';

SELECT (SedentaryMinutes + LightlyActiveMinutes + FairlyActiveMinutes +
VeryActiveMinutes) + 942
FROM daily_activity.daily_activity
WHERE Id = 4388161847 AND ActivityDate = '2016-05-05';

SELECT (SedentaryMinutes + LightlyActiveMinutes + FairlyActiveMinutes +
VeryActiveMinutes) + 776
FROM daily_activity.daily_activity
WHERE Id = 8378563200 AND ActivityDate = '2016-04-25';
```

The query below collects time averages excluding users with the anomaly above:

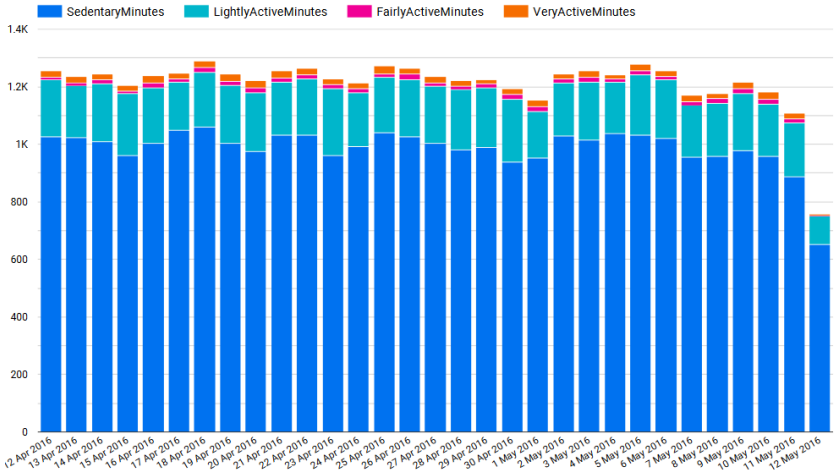
```
SELECT
  ActivityDate,
  AVG(daily_activity.SedentaryMinutes) AS avg_sed_act_min,
  AVG(daily_activity.LightlyActiveMinutes) AS  avg_light_act_min,
  AVG(daily_activity.FairlyActiveMinutes) AS avg_fairly_act_min,
  AVG(daily_activity.VeryActiveMinutes) AS avg_very_act_min,
  AVG(sleep_day.TotalMinutesAsleep) AS avg_sleep_min
FROM daily_activity.daily_activity
FULL JOIN sleep_day.sleep_day
ON CAST(daily_activity.Id AS STRING) = sleep_day.Id AND
daily_activity.ActivityDate = DATE(PARSE_DATETIME('%m/%d/%Y %H:%M:%S %p',
SleepDay))
WHERE sleep_day.Id NOT IN ('4702921684','4388161847', '8378563200')
GROUP BY ActivityDate
ORDER BY ActivityDate;
```

By eliminating those users from the analysis, the sum of the daily average sleep + activity remains below 1440 min.

```
SELECT
  ActivityDate,
  AVG(SedentaryMinutes) AS avg_sed_act_min,
  AVG(LightlyActiveMinutes) AS  avg_light_act_min,
  AVG(FairlyActiveMinutes) AS avg_fairly_act_min,
  AVG(VeryActiveMinutes) AS avg_very_act_min
FROM daily_activity.daily_activity
```

```
GROUP BY ActivityDate
ORDER BY ActivityDate;
```

Average daily activity (min)

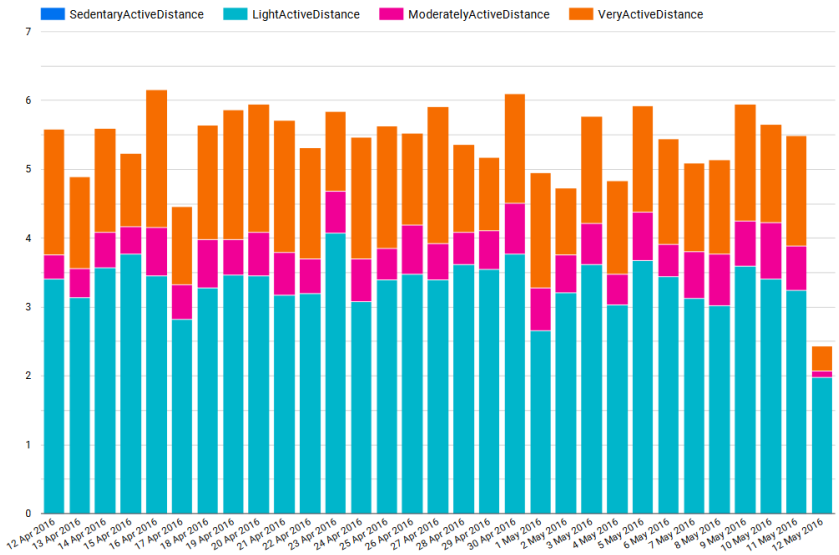


The importance of sedentary activity is even more obvious if we look at the daily activity (number of minutes awake).

```
SELECT
  ActivityDate,
  AVG(SedentaryActiveDistance) AS avg_sed_act_dis,
  AVG(LightActiveDistance) AS  avg_light_act_dis,
  AVG(ModeratelyActiveDistance) AS avg_fairly_act_dis,
  AVG(VeryActiveDistance) AS avg_very_act_dis
FROM daily_activity.daily_activity
GROUP BY ActivityDate
ORDER BY ActivityDate;

SELECT
  AVG(LightActiveDistance) AS  avg_light_act_dis,
  AVG(VeryActiveDistance) AS avg_very_act_dis,
  AVG(ModeratelyActiveDistance) AS avg_fairly_act_dis,
FROM daily_activity.daily_activity;
```

avg_light_act_dis	avg_very_act_dis	avg_fairly_act_dis
3.340819149	1.502680851	0.5675425514

Average daily activity (km)

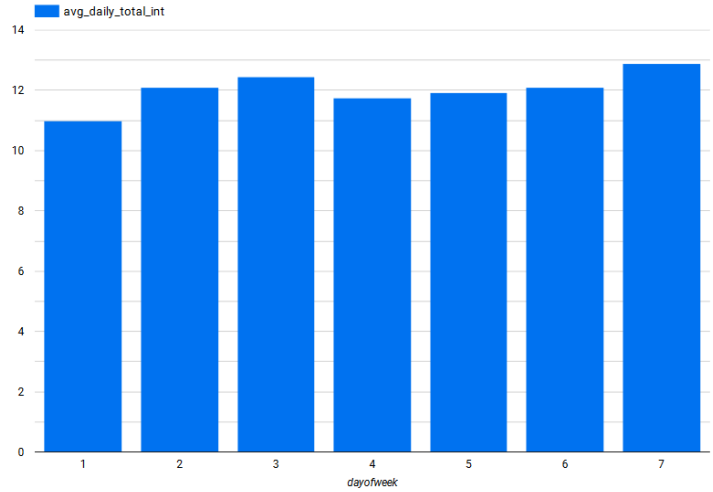


The kms travelled during sedentary activity are close to 0. Users travel on average 3,34 km during light activity, 1,5 km during very active activity and 0,57 km during moderate activity.

3. When is the product used for sport (part of day, part of week)?

```
SELECT
  EXTRACT(DAYOFWEEK FROM Activity_hour) AS dayofweek,
  AVG(TotalIntensity) AS avg_daily_total_int
FROM `tribal-marker-389314.hourly_activity.hourly_activity`
GROUP BY EXTRACT(DAYOFWEEK FROM Activity_hour);
```

Average total intensity per day

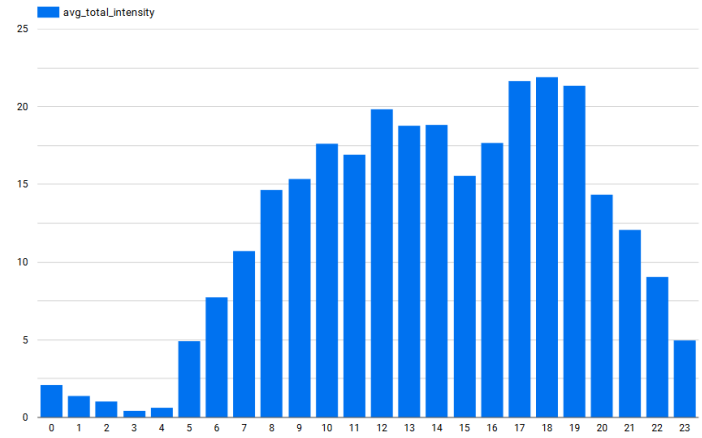


The first day of the week is Sunday.

The most active days are Saturday and Tuesday, the least active one is Sunday.

```
SELECT
  EXTRACT(HOUR FROM Activity_hour) as hour,
  AVG(TotalIntensity) as avg_total_int
FROM `tribal-marker-389314.hourly_activity.hourly_activity`
GROUP BY EXTRACT(HOUR FROM Activity_hour);
```

Average total intensity per hour



The active hours of a day are between 6:00 and 23:00. The peak is between 17:00 and 20:00, so users most probably tend to do sports in the evening.

phase 6 // act

Bellabeat`s possible target group is office workers who intend to include sport in their daily routine. They are not necessarily overweight, but spend considerable time of their day sitting or with light activity. They do sport mainly between 17:00 and 20:00 and their most active day is Saturday and Tuesday.

Based on these findings, Bellabeat should consider:

1. Assess the comfortability of the product (Leaf or Time); design a product that can be comfortable even for sleeping.
2. Align product design with the evening activities by focusing on visibility and accessibility (Leaf or Time).
3. Setting up an alert system in order to stand up and move a few minutes in every hour.
4. Setting up a gamified function in order to increase fairly/moderately and very active minutes/distances (e. g. championships between users, weekly challenges).
5. Gathering richer activity data (activity types; type of sports).