Analysis

Work with dataset to find the insights, by:

- Data
- Aggregation
- Summary
- Gaining insights

I. Hourly steps throughout a day

```
>>>
SELECT
  TIME(ActivityHour) AS Hour,
  Avg(TotalSteps) AS Steps
FROM
  `capstone-case-studies.bellabeat_smart_devices.hourly_activity`
GROUP BY
  Hour
ORDER BY
  Hour;
>>>
```

II. Average steps throughout a week

```
>>>
SELECT
  -- convert strings to date then to weekday
  FORMAT_DATE('%A', PARSE_DATE('%d.%m.%y', ActivityDate)) AS Weekday,
  AVG(TotalSteps) AS avg_steps
FROM
  `capstone-case-studies.bellabeat_smart_devices.daily_activity`
GROUP BY
  Weekday
>>>
```

2. User want to get better sleeping rather than wasting time on bed

I. Find out TotalMinutesAsleep:TotalTimeInBed distribution:

```
>>>
SELECT
  AVG(TotalTimeInBed/TotalMinutesAsleep) AS Mean,
  STDDEV(TotalTimeInBed/TotalMinutesAsleep) AS STD,
  MAX(TotalTimeInBed/TotalMinutesAsleep) AS Max,
  Min(TotalTimeInBed/TotalMinutesAsleep) AS Min
FROM
```

```
 `capstone-case-studies.bellabeat_smart_devices.daily_sleep`
>>>
```

II. Analyze steps for those who not sleep well

```
>>>
-- Organize and format data

WITH
  Lazy_people_data AS
  (SELECT
    A.Id,
    A.TotalSteps,
    S.TotalMinutesAsleep,
    S.SleepInBedRatio
  FROM
    `capstone-case-studies.bellabeat_smart_devices.daily_activity` A

  -- Merge lazy people sleep data with Activity
  RIGHT JOIN
    (SELECT
      Id,
      SleepDay,
      TotalMinutesAsleep,
      -- using ratio for better relationship
      (TotalTimeInBed/TotalMinutesAsleep) AS SleepInBedRatio
    FROM
      `capstone-case-studies.bellabeat_smart_devices.daily_sleep`

    -- Filtering lazy people by meanRatio
    WHERE
      (TotalTimeInBed/TotalMinutesAsleep) > 1.104) S
  ON
    A.Id = S.Id AND
    A.ActivityDate = S.SleepDay)

-- now transform data to find out trends
-- analyze the trends for Active minutes
SELECT
  Id,
  ROUND(AVG(TotalSteps), 2) AS avg_steps,
  AVG(SleepInBedRatio) AS avg_sleep_ratio
```

```
FROM
  Lazy_people_data
GROUP BY
  Id
ORDER BY
  avg_sleep_ratio DESC;
>>>
```

III. Analyze the steps for those who sleep well

```
>>>
-- Organize and format data

WITH
  Active_people_data AS
  (SELECT
    A.Id,
    A.TotalSteps,
    S.TotalMinutesAsleep,
    S.SleepInBedRatio
  FROM
    `capstone-case-studies.bellabeat_smart_devices.daily_activity` A
  -- Merge active people sleep data with Activity
  RIGHT JOIN
    (SELECT
      Id,
      SleepDay,
      TotalMinutesAsleep,
      (TotalTimeInBed/TotalMinutesAsleep) AS SleepInBedRatio
    FROM
      `capstone-case-studies.bellabeat_smart_devices.daily_sleep`

    -- Filtering data using meanRatio
    WHERE
      (TotalTimeInBed/TotalMinutesAsleep) < 1.104) S
  ON
    A.Id = S.Id AND
    A.ActivityDate = S.SleepDay)

-- now transform data to find out trends
-- analyze the trends for Active minutes
```

```
SELECT
  Id,
  ROUND(AVG(TotalSteps), 2) AS avg_steps,
  AVG(SleepInBedRatio) AS avg_sleep_ratio
FROM
  Active_people_data
GROUP BY
  Id
ORDER BY
  avg_sleep_ratio DESC;

>>>
```

Find out the relationship between the different proportions of Active Distances and the total steps.

```
>>>
SELECT
  Id,
  ROUND(AVG(LoggedActivitiesPercent), 2) AS LoggedActivitiesPercent,
  ROUND(AVG(VeryActivePercent), 2) AS VeryActivePercent,
  ROUND(AVG(ModeratelyActivePercent), 2) AS ModeratelyActivePercent,
  ROUND(AVG(LightActivePercent), 2) AS LightActivePercent,
  ROUND(AVG(SedentaryActivePercent), 2) AS SedentaryActivePercent,
  ROUND(AVG(steps), 2) AS avg_steps
From
  -- Convert the distance into the prop
  (SELECT
    Id,
    ActivityDate,
    (LoggedActivitiesDistance/TrackerDistance) As LoggedActivitiesPercent,
    (VeryActiveDistance/TrackerDistance) As VeryActivePercent,
    (ModeratelyActiveDistance/TrackerDistance) As ModeratelyActivePercent,
    (LightActiveDistance/TrackerDistance) As LightActivePercent,
    (SedentaryActiveDistance/TrackerDistance) As SedentaryActivePercent,
    TotalSteps as steps
  FROM
    `capstone-case-studies.bellabeat_smart_devices.daily_activity`
  WHERE
    TrackerDistance <> 0 AND
    (LoggedActivitiesDistance +
    VeryActiveDistance +
```

```
    ModeratelyActiveDistance +
    LightActiveDistance +
    SedentaryActiveDistance)/TrackerDistance > 0.99)
GROUP BY
  1
ORDER BY
  avg_steps DESC
>>>
```

3. Find out Fitbit usage by different types of users

I. Types of user we have in fitbit

```
>>>
SELECT
  UserType,
  ROUND(COUNT(Id)/936, 3) AS Distribution
FROM
  (SELECT
    Id,
    (CASE
      WHEN TotalSteps < 5000 THEN "sedentary"
      WHEN TotalSteps >= 5000 AND TotalSteps < 7499 THEN "lightly active"
      WHEN TotalSteps >= 7500 AND TotalSteps < 9999 THEN "fairly active"
      ELSE "very active"
    END) AS UserType
  FROM `capstone-case-studies.bellabeat_smart_devices.daily_activity`)
GROUP BY
  UserType
>>>
```

II. How much each customer use their smart device monthly

```
>>>
SELECT
  MonthlyUse,
  COUNT(Id)/24 AS Distribution
FROM
  (SELECT
    Id,
    (CASE
      WHEN COUNT(SleepDay) >= 1 AND COUNT(SleepDay) <= 10 THEN "low use"
      WHEN COUNT(SleepDay) >= 11 AND COUNT(SleepDay) <= 20 THEN "moderate use"
```

```
      ELSE "high use"
    END) AS MonthlyUse
  FROM `capstone-case-studies.bellabeat_smart_devices.daily_sleep`
  GROUP BY
    Id)
GROUP BY
  MonthlyUse
>>>
```

III. Daily usage of smart device by each customer

```
>>>
SELECT
  DailyUse,
  ROUND(COUNT(Id)/936, 3) AS Distribution
FROM
  (SELECT
    Id,
    (CASE
      WHEN percentMinuteWorn = 1 THEN "All day"
      WHEN percentMinuteWorn < 1 AND percentMinuteWorn >= 0.5 THEN "More than
half day"
      WHEN percentMinuteWorn < 0.5 AND percentMinuteWorn > 0.0 THEN "Less than
half day"
    END) AS DailyUse
  FROM  (SELECT
      Id,
       (VeryActiveMinutes+fairlyactiveminutes+lightlyactiveminutes+sedentaryminu
tes)/1440 AS percentMinuteWorn
    FROM `capstone-case-studies.bellabeat_smart_devices.daily_activity`))
GROUP BY
  DailyUse
>>>
```