select count(distinct (Id))

from `mystic-song-363215.1.daily\_activity`

# 33 unique ids in the daily activity table

;

select count(distinct(Id))

from `mystic-song-363215.1.sleep\_day`

# 24 unique ids in the sleep day table

;

select count(distinct(Id))

from `mystic-song-363215.1.weightlog\_info`

# 8 unique ids in the weightlog table

;

SELECT count(distinct(Id))

FROM `mystic-song-363215.1.daily\_steps`

# 33 unique ids in the daily steps table

;

SELECT LENGTH (CAST(id as string))

FROM `mystic-song-363215.1.daily\_activity`

# the length of each id in the daily activity table is 10

;

SELECT LENGTH (CAST(id as string))

FROM `mystic-song-363215.1.sleepday`

#the length of each id in the sleep day table is 10

;

SELECT LENGTH (CAST(id as string))

FROM `mystic-song-363215.1.weight`

#the length of each id in the weight table is 10

;

SELECT LENGTH (CAST(id as string))

FROM `mystic-song-363215.1.daily\_steps`

#the length of each id in the daily steps table is 10

;

#To check for duplicates in the dataset

SELECT id ,COUNT(id) as num\_of\_records

FROM `mystic-song-363215.1.daily\_activity`

GROUP BY Id

HAVING COUNT(id) > 1

# There were alot of duplicate ids due to the times a user used the device

# flactuation in the duplicated ids shows inconsistency in the use of the device

;

SELECT id, count(\*) as num\_of\_rows

FROM `mystic-song-363215.1.daily\_activity`

GROUP BY id, ActivityDate

HAVING num\_of\_rows > 1

#There are no duplicate rows in the activity table

;

SELECT Id,COUNT(\*) as num\_of\_rows

FROM `mystic-song-363215.1.sleepday`

GROUP BY Id,SleepDay,TotalSleepRecords

HAVING num\_of\_rows > 1

#There are no duplicate rows in the sleepday table removed in excel

;

SELECT Id ,COUNT(\*)as num\_of\_rows

FROM `mystic-song-363215.1.daily\_steps`

GROUP BY Id,ActivityDay,StepTotal

HAVING num\_of\_rows > 1

#there are no duplicate rows in the daily steps table

;

SELECT Id,COUNT(\*) as num\_of\_rows

FROM `mystic-song-363215.1.weight`

GROUP BY Id ,Date,Fat

HAVING num\_of\_rows > 1

#there are no duplicates in the weight table

;

########to split the date and time from the sleepday column

SELECT CAST(sleepday as date),CAST(Sleepday as time)

FROM `mystic-song-363215.1.sleepday`

#########

;

SELECT Id ,COUNT(DISTINCT(ID)) as zerosteps

FROM `mystic-song-363215.1.daily\_steps`

WHERE StepTotal = 0

GROUP BY id

ORDER BY zerosteps desc

#there are 15 id with zero steps

;

#to check number of records with zero steps

SELECT SUM(zerosteps)as dayszero

FROM(SELECT Id ,COUNT(\*) as zerosteps

FROM `mystic-song-363215.1.daily\_steps`

WHERE StepTotal = 0

GROUP BY id

ORDER BY zerosteps desc) as az

#There are 77 records with zero steps

;

SELECT \* ,ROUND((sedentaryminutes/60),2)as sedentaryhours

FROM `mystic-song-363215.1.daily\_activity`

WHERE TotalSteps = 0

#most ids are inactive with a few having very active minutes but no steps counted

;

#analysis

#assigning date as weekends and weekdays

SELECT Activitydate ,EXTRACT(DAYOFWEEK from ActivityDate) as DayOfWeek

FROM `mystic-song-363215.1.daily\_activity`

;

SELECT ActivityDate,

CASE

WHEN DayOfWeek = 2 THEN 'Monday'

WHEN DayOfWeek = 3 THEN 'Tuesday'

WHEN DayOfWeek = 4 THEN 'Wednesday'

WHEN DayOfWeek = 5 THEN 'Thursday'

WHEN DayOfWeek = 6 THEN 'Friday'

WHEN DayOfWeek = 7 THEN 'Saturday'

ELSE 'Sunday'

END AS DayofWeek

FROM(SELECT Activitydate ,EXTRACT(DAYOFWEEK from ActivityDate) as DayOfWeek

FROM `mystic-song-363215.1.daily\_activity`) as temp

;

#to check the average amount of time spent in the bed and asleep

SELECT id,ROUND(AVG(TotalTimeInBed/60),2) as hoursinbed, ROUND(AVG(TotalMinutesAsleep/60),2) hoursasleep

FROM `mystic-song-363215.1.sleepday`

GROUP BY Id

ORDER BY Id ASC

;

#to show the amount of time it takes a person to sleep

SELECT Id ,( ROUND(AVG(TotalTimeInBed) - AVG(TotalMinutesAsleep),2)) as minutesbeforesleeping

FROM `mystic-song-363215.1.sleepday`

GROUP BY id

ORDER BY Id ASC

#shows that some ids took alot of timebefore sleeping

;

#to see the average distance and active minutes an time takes a user to sleep

SELECT Id, ROUND(AVG(TotalDistance),2) as distance, ROUND(AVG((VeryActiveMinutes+FairlyActiveMinutes+LightlyActiveMinutes)/60),2) as totaltime

FROM `mystic-song-363215.1.daily\_activity`

GROUP BY Id

ORDER BY Id ASC

;

#comparison between activetime,distance,hour slept,and time taken before sleeping

SELECT S.Id,

ROUND(AVG(TotalDistance),2) as distance ,

ROUND(AVG((VeryActiveMinutes+FairlyActiveMinutes+LightlyActiveMinutes)/60),2) as activetime ,

ROUND(AVG(TotalTimeInBed) - AVG(TotalMinutesAsleep),2) as minutesbeforesleeping,

ROUND(AVG(TotalMinutesAsleep/60),2) hoursasleep

FROM `mystic-song-363215.1.sleepday` as S

JOIN `mystic-song-363215.1.daily\_activity` as DA

ON S.Id = DA.Id

GROUP BY Id

ORDER BY id DESC

;

# calories burnt ,steps taken and mins taken to sleep

SELECT DA.Id,

AVG(Calories) as caloriesburnt,

ROUND(AVG(TotalTimeInBed) - AVG(TotalMinutesAsleep),2) as minutesbeforesleeping,

ROUND(AVG(TotalMinutesAsleep/60),2) timeasleep

FROM `mystic-song-363215.1.daily\_activity`as DA

JOIN `mystic-song-363215.1.sleepday` as S

ON DA.Id = S.Id

GROUP BY Id,Calories

ORDER BY Calories DESC

;

#to show the weight ,active minutes and sleep time

SELECT W.Id,

ROUND(AVG(Weightkg),2) as weight,

ROUND(AVG((VeryActiveMinutes+FairlyActiveMinutes+LightlyActiveMinutes)/60),2) as activetime ,

ROUND(AVG(TotalMinutesAsleep/60),2) hoursasleep

FROM `mystic-song-363215.1.weight` as W

INNER JOIN `mystic-song-363215.1.sleepday` as S

ON W.Id = S.Id

INNER JOIN `mystic-song-363215.1.daily\_activity` as DA

ON S.Id = DA.Id

GROUP BY id

ORDER BY weight DESC

#only 6 records are shown

;

#to find ids that dont have records in either the sleep or weight table

SELECT COUNT(DISTINCT(Id))

FROM `mystic-song-363215.1.daily\_activity`

WHERE Id NOT IN (SELECT S.Id

FROM `mystic-song-363215.1.sleepday` as S

JOIN `mystic-song-363215.1.weight` as W

ON S.Id = W.Id

GROUP BY Id )

#Out of the 33 unique ids in daily activity,27 dont have records in the sleep and weight tables

;

#this query pulls days of the week with no stepcount

SELECT COUNT(StepTotal) as steps,

CASE

WHEN DayOfWeek = 2 THEN 'Monday'

WHEN DayOfWeek = 3 THEN 'Tuesday'

WHEN DayOfWeek = 4 THEN 'Wednesday'

WHEN DayOfWeek = 5 THEN 'Thursday'

WHEN DayOfWeek = 6 THEN 'Friday'

WHEN DayOfWeek = 7 THEN 'Saturday'

ELSE 'Sunday'

END AS DayofWeek

FROM(SELECT \*,EXTRACT(DAYOFWEEK from ActivityDay) as DayOfWeek

FROM `mystic-song-363215.1.daily\_steps`) as temp

WHERE StepTotal =0

GROUP BY DAYOFWEEK

ORDER BY DAYOFWEEK DESC

;

#average amount of sleep per day

SELECT

ROUND(AVG(TotalMinutesAsleep/60),2) timeasleep,

CASE

WHEN DayOfWeek = 2 THEN 'Monday'

WHEN DayOfWeek = 3 THEN 'Tuesday'

WHEN DayOfWeek = 4 THEN 'Wednesday'

WHEN DayOfWeek = 5 THEN 'Thursday'

WHEN DayOfWeek = 6 THEN 'Friday'

WHEN DayOfWeek = 7 THEN 'Saturday'

ELSE 'Sunday'

END AS DayofWeek

FROM(SELECT \*,EXTRACT(DAYOFWEEK from ActivityDay) as DayOfWeek

FROM `mystic-song-363215.1.daily\_steps`) as temp

JOIN `mystic-song-363215.1.sleepday` as S

ON temp.Id = S.Id

GROUP BY DAYOFWEEK

ORDER BY DayOfWeek DESC

;

# to find the dates the data was collected in the daily intensities table

SELECT MIN (ActivityDate) AS startdate, MAX(ActivityDate) AS enddate

FROM `mystic-song-363215.1.daily\_activity`

# startdate 2016-04-12 enddate 2016-05-12

;