Physical health in corona times: researching daily screen time and steps count

C.A.F. de Pont
Eindhoven,
The Netherlands
c.a.f.d.pont@
student.tue.nl

E.M. van Hilten
Eindhoven,
The Netherlands
e.m.v.hilten@
student.tue.nl

H. van Rixtel
Eindhoven,
The Netherlands
h.v.rixtel@
student.tue.nl

I.W.A. Hikspoors
Eindhoven,
The Netherlands
i.w.a.hikspoors@
student.tue.nl

F.J. Heij
Eindhoven,
The Netherlands
f.j.heij@
student.tue.nl

ABSTRACT
This paper depicts a research that uncovers the possible relation between a person's screen time and steps taken in a day and so, we research how much of the time that we are not working we spend moving. Furthermore, it is researched if an external factor like indoor temperature influences this relation. Five participants have collected temperature, screen time and step count data by a temperature/humidity sensor (DHT11), LDR sensor and a sport band (Xiaomi Mi Band 4, Mi Band 5 or an Huawei band 30 Pro). This data is stored in Excel sheets which are imported, cleaned, analyzed and visualized in Python. These visualizations show that some participant take more steps when they are offline, and some participants when they are online. Some participants take more steps when the temperature was higher. Overall, no conclusions can be drawn due to the divergent lifestyles and the small sample size.

INTRODUCTION
The corona pandemic had a great influence on our daily lives. Due to lockdowns, we started working from home, our social lives were mostly online and gyms were closed. Before the pandemic, we would bike to work, go dancing with friends and train at the gym. But during lockdowns daily schedules changed and most of us had to find different ways to be physically active.

Being physically active is important for both our physical and mental health. Although exercising won't reduce the risk of a corona infection, it can positively influence the immune system [2]. Researchers found increases in mental health problems like depression and anxiety during the pandemic [4]. A study in Austria found that even more for women and young adults being physically active is related to better mental health [3]. However, earlier research from before the pandemic shows that many factors can have an influence. For example, for many people the social aspects of sports are just as important as the sports themselves [7]. Teychenne et al [7] explain that exercising can have a negative influence on mood if someone feels pressured or feels like they have to. Other studies show that a lot of screen time can negatively affect mood, especially the screen time that is not work-related [9].

In order to increase their physical activity, many people started walking. Increase in use of apps like 'Het Ommetje' show that walking was an upcoming sport in corona times [5]. Research shows that it is healthy to take 7000 to 8000 steps per day and do leisure activities[1]. In this paper, we research how much of the time that we are not working we spend moving. Thus, it is interesting to see how we divide our days and if we take more steps when we have more leisure time on a day [1,8]. Therefore, our research question is: what is the relation between screen time and the number of steps taken per day? We hypothesize that we take less steps if we spend more time behind our screens. Temperature might have a great influence on our daily activities as well, so we will check if room temperature has any influence on the number of steps that we take. For example, when it gets really hot in our room, we might decide to go outside for a walk more often .

Since there are so many restrictions in corona times we are tracking our own steps, screen time data and room temperature. We are five students that probably fill their days with different activities. Most of us only use our laptop for study-related activities and almost never for leisure activities. Especially in corona times most work involves a screen, due to online meetings and just being able to collaborate more efficiently at a distance. First we will discuss our sensor selection, then we will present our process and methods and finally we will describe our results.

DESIGN
For our study we collect data of the laptop time, temperature, humidity and the number of taken steps. The number of steps are collected by the use of a Xiaomi Mi Band 4, Mi Band 5 or an Huawei band 30 Pro. The screen time is measured with an LDR sensor. This sensor measures the total amount of light, including the light of the screen. Nevertheless the direct surroundings of the LDR have a big influence on the gathered data, such as the sunlight. Therefore the LDR was placed close to the laptop

with a casing to separate the screen from outside jammers (Figure 1). Because of this the data was mostly influenced by the light of the screen. If the laptop is turned on, the screen emits more light. The LDR measures the light intensity, which makes it possible to determine whether the screen is on or off.

By connecting the circuit to the laptop, the temperature/humidity sensor (DHT11) measures data on the location of the laptop. This data and the data of the LDR will be saved on an SD card placed in the data logger shield (RobotDyn). This shield with an SD card is placed on top of the Arduino UNO.

In Figure 2 the circuit is shown with the DHT11, LDR and a resistor of 10K.

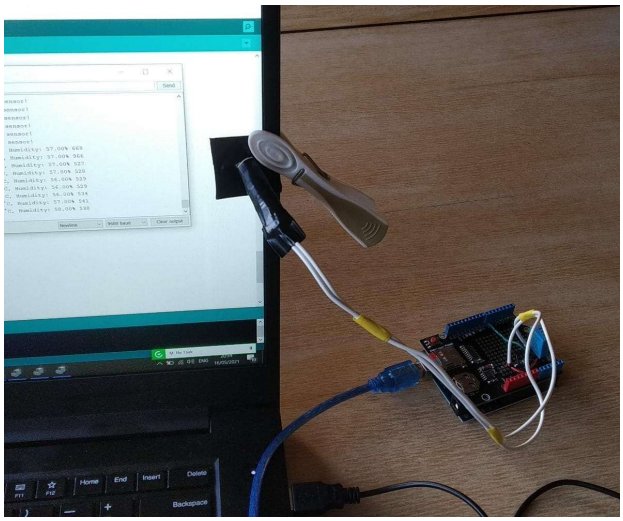


Figure 1. The study setup: the LDR is covered and attached to the screen with a clip, and the Arduino Uno and DHT11-sensor on the right.

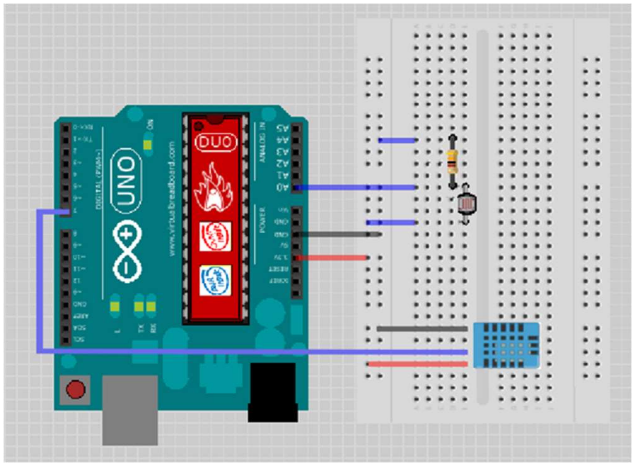


Figure 2. Schematic of the electronic circuit [6].

PROCESS

In this chapter, the different steps of the process are discussed. These consist of the acquisition of stakeholders and collecting data, importing the data, cleaning the data, and analyzing and visualizing the data.

Acquisition of stakeholders and collecting data

The data that is collected for this study was obtained from the group members. This data was only shared within the group, and with the tutors and teachers of the course for which the study was carried out. All group members are Industrial Design students. They travel regularly from home to the university and sometimes travel longer to visit their parents or friends. Two of the participants have set times for sports training. The other three members sport when they feel like it or they go for a walk every now and then. These differences might be visible in the data and therefore the group members can recognize datasets and link these to other members. It is important to note that this information stays within the group so that outside of this course, the data is anonymous. Before starting this research, a study proposal was approved (Appendix A).

Importing the data

Both the data from the DHT sensor and the Mi Band/Huawei band 30 Pro had to be imported and put into an Excel sheet to be able to use it in Python. As explained in the previous chapter, the DHT data was stored on an SD card in an Excel sheet. This made it easy to import the data in Python. On the other hand, the Mi Band led to more troubles. The Mi Fit app that comes with this smartwatch can export Excel sheets with the collected data, but this often resulted in empty sheets for the group members. Therefore the app Notify for the Mi Band was used. Although this was a better option it led to two other problems. Firstly, this app restricted us in using data that was collected before the moment of downloading the app. Therefore the start of collecting the data that was used for the study was later than planned. Secondly, the Notify for Mi Band app could not be used by the IOS user in the group. The app QS access was used to download the data. One participant gathered data by the use of a Huawei band 30 Pro. She was not able to export her data to Excel instantly. She used the Huawei health app to put her data manually into an Excel sheet.

Cleaning the data

The data sheet that was obtained from the participant that used the QS access app, showed the number of steps per hour. Therefore, every group member had to set the measuring points to hours. This was done by taking the average temperature per hour and the sum of taken steps at the end of every hour. To further clean the data, the average value of the LDR sensor was calculated per hour. Every member created conditions for their own LDR to define whether the laptop was on, off or if it was uncertain. Lastly,

columns were made that contained the month, day and hour. The entire Python code can be found in Appendix B.

Visualizing and analyzing

After cleaning the data, making visualizations was the next step to be able to draw conclusions from the data. For this, we tried out many types of visuals in order to find the ones that best showed the correlation between the screen time and the number of steps. The code that was used is in Appendix C. Our final visuals are a boxplot that shows the relation between the number of steps and the screen status of all the participants combined, and a bar chart is created to verify any relation between the number of steps and the temperature. For this bar chart we created five separate graphs to make the comparison a bit easier.

ETHICS AND FAIR-PRINCIPLES

FAIR-principles

Findable: The data is made findable by storing the collected data in the Canvas page of our course group (team 9). The team members, the faculty, the Making Sense of Sensors course, the teachers and tutors know this and can find the data here.

Accessible: The data is made accessible by using standardized and consistent communication. Information about numerical values is provided and these values have consistency in the stored location and unit. The information is provided in English and punctuation is used consistently. When the data is no longer available, there will be provided an appendix with metadata.

Interoperable: The information and data is made interoperable by using a language that everyone understands. English and universal vocabulary are consistently used. Next to this, detailed descriptions of definitions are provided where needed.

Reusable: The data is reusable as the data is provided with an extensive and clear description of what kind of data and information there is provided (the metadata). There is explained what kind of data is collected, in what unit, in what time span, how it is collected (used software, hardware and electronic circuit) and what external factors influence the data (for example, sunlight on the LDR). Next to this, it is explained what attributes the columns show. When the data is no longer available, it will be clear how to re-use it. The metadata can be found in Appendix D.

Ethics

Ethics was taken into account in multiple ways. First of all, privacy of any group members is not violated. Group members remained anonymous and no personal data was stored. Of course, the group is small, so if one knows a bit more about a group members' habits (for example one person trains every Thursday at 9 am), they can easily match the data with the right person. This required strict agreements within the group. Data is only shared within the group and with teachers and tutors of the course 'Making Sense of Sensors'. Data is shared on a safe shared platform, Canvas, so the chances of a data leak are as small as possible.

Secondly, group members should not feel unsafe, pressured or uncomfortable in any way. The study influenced their behavior as little as possible, so the group allowed them to go through their days as they like. The temperature and humidity sensor did track data in the background and since the group only looked at each other's data after two weeks, it is not likely that this data influenced one's behavior.

However, everyone is wearing a Mi Band or a Huawei Band 30 Pro and they can constantly see their number of steps per day and their heart rate. There is no guarantee that this won't influence one's behavior. For example, a group member might be motivated to walk more if they see that their average number of steps is low. So, the group members did first get used to the Mi Band before it collected data. This might decrease the possibility of influencing.

This may have influenced outcomes of the study, but should not be an ethical problem as long as this was their own decision. The group did avoid pressuring or influencing each other. They did not share data within the group during the two weeks of the experiment, so no one would feel obligated to move more or less to be better than or have similar data to the rest of the group.

RESULTS

The boxplots in Figure 3 shows that participants 1 and 4 often take more steps per hour than the other three participants. Participant 4 clearly takes more steps when their screen is off. Participant 5 takes more steps when their screen is on. Appendix E holds the boxplots per participant, they show the results for participant 2, 3 and 5 a bit more in detail.

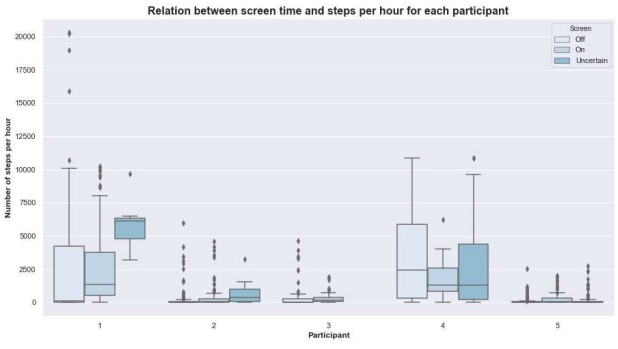


Figure 3. Boxplots of participants 1-5.

We also checked for all participants if there was a relation between temperature and the number of steps per hour. The

graph for participant 1 shows an ascending line; when the temperature goes up, they often take more steps as well. The distribution for participant 2 looks similar to a normal distribution. They took more steps on average temperatures of 24 degrees. Participants 3 and 5 do not show a clear temperature preference. Participant 4 took more steps per hour when the temperature was about 21 degrees Celsius. It is important to note that the average step count per hour varies a lot between participants. Thus the y-axis also has other values for each participant.

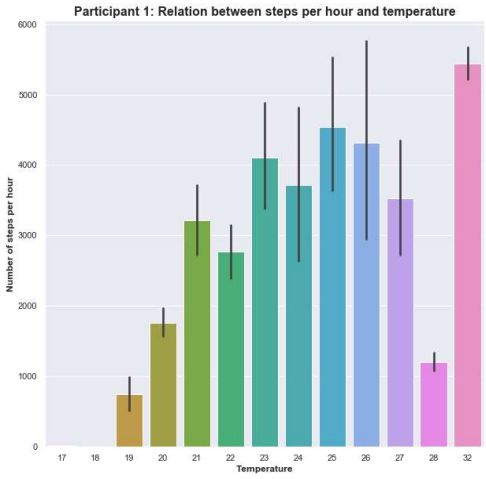


Figure 4. Barplot participant 1.

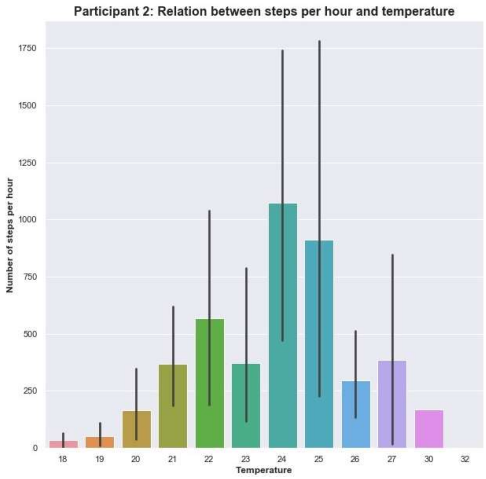


Figure 5. Barplot participant 2.

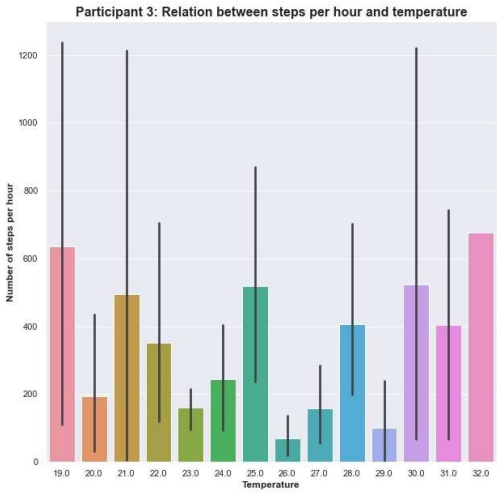


Figure 6. Barplot participant 3.

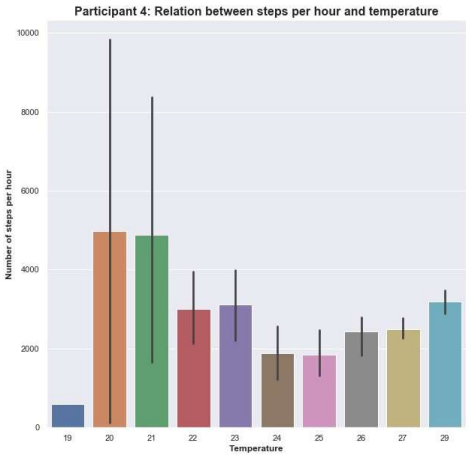


Figure 7. Barplot participant 4.

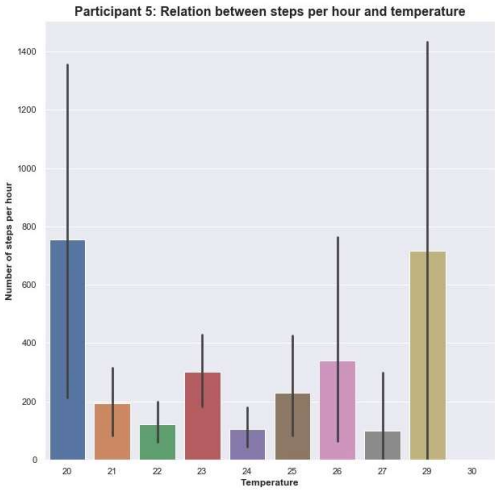


Figure 8. Barplot participant 5.

DISCUSSION

During this research, we learned things because of our own choices and new experiences during the project. We will discuss and reflect how these learning points would change a similar research project like this in the future.

We researched the relation between laptop screen time and the number of steps taken per day. Furthermore we checked if room temperature has any influence on the number of steps that we take. Therefore, we used a setup consisting of a LDR and a temperature/humidity sensor. The LDR had to stay connected to the laptop screen and so, the temperature sensor was close to the laptop which might influence the temperature because of the laptop heat. It would be better to create two separate setups for the two sensors.

The screen time and temperature had to be measured during the entire day and therefore, we had to bring it everywhere. When traveling there is no data collected, because of missing power and sometimes participants forgot to connect it again. The setup was a little bit fragile which let the wires move sometimes (when traveling) and so, the data was not collected properly. When forgetting to check this, we get again more gaps in the data. Next time, all the wires should be soldered to make it more stable.

The LDR did not give consistent output: the range between the different participants differs too much and so the individual LDR sensors/values differ from each other. Besides this, for some participants the output values for screen on or off were really close or even the same sometimes which causes uncertainties. All the participants made sure that there was almost no influence of environmental light. We think some small differences in the values can be caused by differences in activities on the laptop, difference in brightness or settings, but the inconsistency of the LDRs creates the most differences in values. Using another sensor, or a sensor of higher quality, might be better.

In the first weeks of the research, we started collecting data, but ran into problems a few times and so, the collected data was not relevant anymore. For example, after a few days we wanted to export the sport band data, but we did not get what we wanted. We had to use the Notify app, but this one did not export data retroactively and was not available for IOS users. Therefore, there only could be exported data from that day on. To be more efficient in the end, it would be better to have a testing day where we would have made sure everything was working in the right way. It would also be good to make sure that all the participants check if the data is collected each day. Some of the participants did, but others did not and so, they have in the end collected data less days because something went wrong.

In the analyzing phase, the data cleaning steps were very different for each participant because of differences in sport band and exporting app as mentioned before. This resulted in small differences in the final cleaned data set of each participant. It took some time to make everything consistent, which could be prevented by setting guidelines that everyone should follow. All these things together would make a research project like this more efficient and effective.

CONCLUSION

The research question was: what is the relation between screen time and the number of steps taken per day? We found that the lifestyles of the participants differ much, thus it is hard to draw conclusions with the present sample size. More participants would be needed to validate relations that are currently uncertain. We conclude that there are no significant correlations between the participants' screen time and the number of steps taken per day. Next to that, no proof was found that the temperature of the participants' environment influences the steps taken per day.

REFERENCES

[1] Yukitoshi Aoyagi and Roy. J. Shephard. 2009. Steps Per Day. Sports Medicine. 39, 6 (2009), 423-438.

[2] Wilhelm Bloch, Martin Halle and Jürgen M. Steinacker. 2020. Sport in Corona Times. Dtsch Z Sportmed. 71 (2020), 83-84. DOI: 10.5960/dzsm.2020.432.

[3] Christoph Pieh, Sanja Budimir and Thomas Probst. 2020. The effect of age, gender, income, work, and physical activity on mental health during coronavirus disease (COVID-19) lockdown in Austria. Journal of Psychosomatic Research.136 (September, 2020), 110278. DOI: https://doi.org/10.1016/j.jpsychores.2020.110186

[4] Nicola Meda, Susanna Pardini, Irene Slongo, Luca Bodini, Mauro Agostino Zordan, Paolo Rigobello, Francesco Visioli and Caterina Novara. 2021. Journal of Psychiatric Research. 134 (February 2021), 69-77. DOI: https://doi.org/10.1016/j.jpsychires.2020.12.045

[5] RTLNieuws. 2021. Micha bedacht de Ommetje-app: 'Dit succes was niet te voorspellen'. https://www.rtlnieuws.nl/economie/business/artikel/5213878/ommetje-makers-bedenker-erik-scherder-hersenstichting-bedrijf

[6] Tarantula3. 2018. Assembly [Illustratie]. Create.Arduino.cc (6-15-2021) https://create.arduino.cc/projecthub/tarantula3/using-an-ldr-sensor-with-arduino-807b1c

[7] Megan Teychenne, Rhiannon L. White, Justin Richards, Felipe B. Schuch, Simon Rosenbaum and Jason A. Bennie. 2020. Do we need physical activity guidelines for mental health: What does the evidence tell us? Mental Health and Physical Activity. 18

(March 2020), 100315. DOI: https://doi.org/10.1016/j.mhpa.2019.100315

[8] Mary West. 2021. 8 tips to try to maintain health while working from home. 2021. https://www.medicalnewstoday.com/articles/working-from-home-health-tips

[9] Suhua Xiao, Zi Yan and Li Zhao. 2020. Physical Activity, Screen Time and Mood Disturbance Among Chinese Adolescents During COVID-19. Journal of Psychosocial Nursing and Mental Health Services. 59, 4 (2020). DOI: https://doi.org/10.3928/02793695-20201104-04

APPENDIX

Appendix A. The study proposal.

STUDY PROPOSAL

MAKING SENSE OF SENSORS

INTRODUCTION

In the corona pandemic working from home has become the norm. People are sitting more behind their laptops. Research shows that taking 7000-8000 steps per day is healthy, next to separating working and leisure time [1,2]. We want to investigate whether people take time off from their laptop time when the weather is nice and during a free day, as it is good for their physical vitality and mental well-being.

WHAT IS THE RELATION BETWEEN LAPTOP SCREEN TIME AND THE NUMBER OF STEPS TAKEN PER DAY?

The experiment researches the following variables: laptop screen time per day and the number of steps taken per day. Five students will measure the variables every 30 minutes for 14 days. The screen time will be measured by an LDR circuit on an Arduino that will track the light coming from the laptop. The application ProcrastiTracker will be used to do a one-time test to check the accuracy of the LDR since there will be light in the surrounding. The steps will be recorded with the Mi Band 4 or 5. The data will be analyzed with Python. We are curious whether the relation between screen time and steps taken is influenced by the room temperature and the type of day (free or workday) they are measured on. The room temperature will be measured using a DHT11 Temperature/ Humidity Sensor and stored on an SD card using a RobotDyn Data Logging Shield.

Bottlenecks that can occur during the execution of the experiment can be the sensors breaking down and thus, fail to collect the data continuously. To minimize possible data gaps we will check the datasheet twice a week and check the blinking Arduino light every day. We will start on time, so we will have some extra time to deal with the things that go wrong.

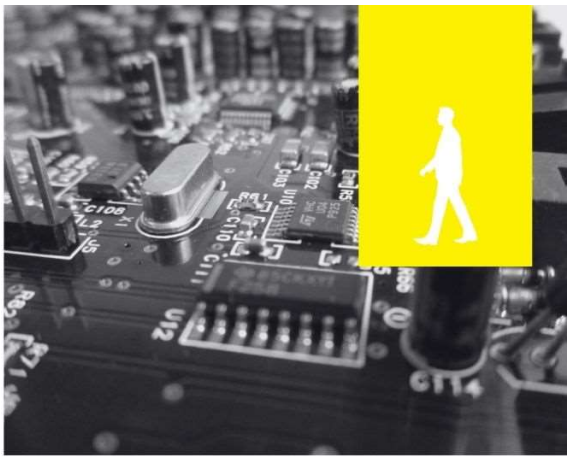
FAIR-PRINCIPLES

FINDABLE. The data will be findable by storing the collected data on the Canvas file page of our course group (team 9). The team members, the faculty, the Making Sense of Sensors teachers and tutors know this and can find the data here.

ACCESSIBLE. The data will be accessible by using standardized and consistent communication. Information about numerical values will be provided and these values will have consistency in the stored location and unit. The information will be provided in English and punctuation will be used consistently.

INTEROPERABLE. The information and data will be interoperable by using a language that everyone understands. We will use English consistently and use universal vocabulary. Next to this, we will provide detailed descriptions of definitions where needed or reference an external source for more information.

REUSABLE. The data will be provided with extensive and clear descriptions. There will be explained what kind of data is collected, in what unit, in what time span and how it is collected. Next to this, the attributes of the columns will be explained. When the data is no longer available, it will be clear how to reuse it.



ETHICS

Ethics will be taken into account in multiple ways. First of all, privacy of any group members should not be violated. Therefore, group members should remain anonymous and no personal data should be stored. Of course, the group is small, so if one knows a bit more about a group members' habits (for example one person trains every Thursday at 7 pm), they can easily match the data with the right person. This requires strict agreements within the group. Data will only be shared within the group and with teachers and tutors of the course 'Making Sense of Sensors'. Data will be shared on a safe shared platform, Canvas, so the chances of a data leak are as small as possible.

Secondly, group members should not feel unsafe, pressured nor uncomfortable in any way. The study should influence their behavior as little as possible, so the group will allow them to go through their days as they like. The temperature and humidity sensor will track data in the background and since the group will only look at this data after two weeks, it is not likely that this data is going to influence one's behavior in any negative way.

However, everyone is wearing a MiBand and they can constantly see their number of steps per day and their heart rate. There is no guarantee that this won't influence one's behavior. For example, a group member might be motivated to walk more if they see that their average number of steps is low.

This may influence the outcomes of the study, but should not be an ethical problem as long as this is their own decision. The group needs to avoid pressuring or influencing each other.

The best way to do this would be not to share data within the group during the two weeks of the experiment, so no one will feel obligated to move more or less to be better than or have similar data to the rest of the group.

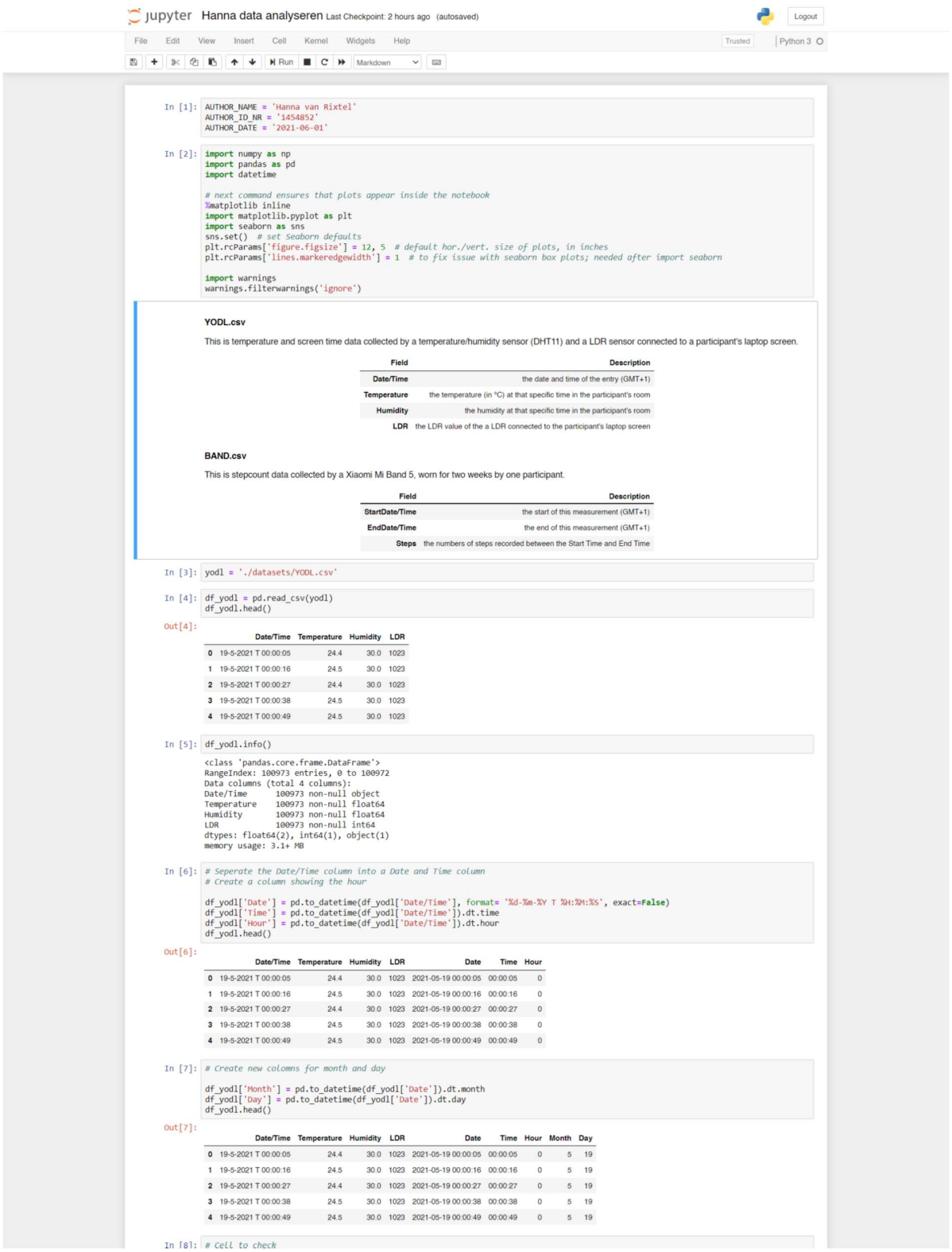
REFERENCES

[1] 8 tips to try to maintain health while working from home. 2021. Medicalnewstoday.com. https://www.medicalnewstoday.com/articles/working-from-home-health-tips.
[2] Aoyagi, Y. and Shephard, R. 2009. Steps Per Day. Sports Medicine 39, 6, 423-438.

PROPOSED BY STUDENT TEAM 9

COACHED BY RICK VAN SCHIE

Appendix B. Python code: analysis



jupyter Hanna data analyseren Last Checkpoint: 2 hours ago (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Markdown

In [1]:
```
AUTHOR_NAME = 'Hanna van Rixtel'
AUTHOR_ID_NR = '1454852'
AUTHOR_DATE = '2021-06-01'
```

In [2]:
```
import numpy as np
import pandas as pd
import datetime

# next command ensures that plots appear inside the notebook
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()  # set Seaborn defaults
plt.rcParams['figure.figsize'] = 12, 5  # default hor./vert. size of plots, in inches
plt.rcParams['lines.markeredgewidth'] = 1  # to fix issue with seaborn box plots; needed after import seaborn

import warnings
warnings.filterwarnings('ignore')
```

YODL.csv

This is temperature and screen time data collected by a temperature/humidity sensor (DHT11) and a LDR sensor connected to a participant's laptop screen.

Field	Description
Date/Time	the date and time of the entry (GMT+1)
Temperature	the temperature (in °C) at that specific time in the participant's room
Humidity	the humidity at that specific time in the participant's room
LDR	the LDR value of the a LDR connected to the participant's laptop screen

BAND.csv

This is stepcount data collected by a Xiaomi Mi Band 5, worn for two weeks by one participant.

Field	Description
StartDate/Time	the start of this measurement (GMT+1)
EndDate/Time	the end of this measurement (GMT+1)
Steps	the numbers of steps recorded between the Start Time and End Time

In [3]:
```
yodl = './datasets/YODL.csv'
```

In [4]:
```
df_yodl = pd.read_csv(yodl)
df_yodl.head()
```

Out[4]:

	Date/Time	Temperature	Humidity	LDR
0	19-5-2021 T 00:00:05	24.4	30.0	1023
1	19-5-2021 T 00:00:16	24.5	30.0	1023
2	19-5-2021 T 00:00:27	24.4	30.0	1023
3	19-5-2021 T 00:00:38	24.5	30.0	1023
4	19-5-2021 T 00:00:49	24.5	30.0	1023

In [5]:
```
df_yodl.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 100973 entries, 0 to 100972
Data columns (total 4 columns):
Date/Time      100973 non-null object
Temperature    100973 non-null float64
Humidity       100973 non-null float64
LDR            100973 non-null int64
dtypes: float64(2), int64(1), object(1)
memory usage: 3.1+ MB
```

In [6]:
```
# Seperate the Date/Time column into a Date and Time column
# Create a column showing the hour

df_yodl['Date'] = pd.to_datetime(df_yodl['Date/Time'], format= '%d-%m-%Y T %H:%M:%S', exact=False)
df_yodl['Time'] = pd.to_datetime(df_yodl['Date/Time']).dt.time
df_yodl['Hour'] = pd.to_datetime(df_yodl['Date/Time']).dt.hour
df_yodl.head()
```

Out[6]:

	Date/Time	Temperature	Humidity	LDR	Date	Time	Hour
0	19-5-2021 T 00:00:05	24.4	30.0	1023	2021-05-19 00:00:05	00:00:05	0
1	19-5-2021 T 00:00:16	24.5	30.0	1023	2021-05-19 00:00:16	00:00:16	0
2	19-5-2021 T 00:00:27	24.4	30.0	1023	2021-05-19 00:00:27	00:00:27	0
3	19-5-2021 T 00:00:38	24.5	30.0	1023	2021-05-19 00:00:38	00:00:38	0
4	19-5-2021 T 00:00:49	24.5	30.0	1023	2021-05-19 00:00:49	00:00:49	0

In [7]:
```
# Create new colomns for month and day

df_yodl['Month'] = pd.to_datetime(df_yodl['Date']).dt.month
df_yodl['Day'] = pd.to_datetime(df_yodl['Date']).dt.day
df_yodl.head()
```

Out[7]:

	Date/Time	Temperature	Humidity	LDR	Date	Time	Hour	Month	Day
0	19-5-2021 T 00:00:05	24.4	30.0	1023	2021-05-19 00:00:05	00:00:05	0	5	19
1	19-5-2021 T 00:00:16	24.5	30.0	1023	2021-05-19 00:00:16	00:00:16	0	5	19
2	19-5-2021 T 00:00:27	24.4	30.0	1023	2021-05-19 00:00:27	00:00:27	0	5	19
3	19-5-2021 T 00:00:38	24.5	30.0	1023	2021-05-19 00:00:38	00:00:38	0	5	19
4	19-5-2021 T 00:00:49	24.5	30.0	1023	2021-05-19 00:00:49	00:00:49	0	5	19

In [8]:
```
# Cell to check
```

```
df_yod1.count()
```

Out[8]:
```
Date/Time      100973
Temperature    100973
Humidity       100973
LDR            100973
Date           100973
Time           100973
Hour           100973
Month          100973
Day            100973
dtype: int64
```

In [9]:
```
# Create a new dataframe without the LDR information

df_temperature = df_yod1[["Temperature", "Humidity", "Month", "Day", "Time", "Hour"]].copy()
df_temperature.head()
```

Out[9]:

	Temperature	Humidity	Month	Day	Time	Hour
0	24.4	30.0	5	19	00:00:05	0
1	24.5	30.0	5	19	00:00:16	0
2	24.4	30.0	5	19	00:00:27	0
3	24.5	30.0	5	19	00:00:38	0
4	24.5	30.0	5	19	00:00:49	0

In [10]:
```
# Calculating the mean of the temperature and humidity of a hour

grouped_temperature = df_temperature.groupby(['Month','Day','Hour'])
df_temperature_mean = grouped_temperature['Temperature', 'Humidity'].mean().reset_index()
df_temperature_mean.head()
```

Out[10]:

	Month	Day	Hour	Temperature	Humidity
0	5	19	0	24.392308	30.744615
1	5	19	1	24.345399	31.398773
2	5	19	2	24.332615	32.116923
3	5	19	3	24.323926	32.877301
4	5	19	4	24.316615	33.000000

In [11]:
```
# Create a new dataframe without the information about the temperature and humidity

df_ldr = df_yod1[["LDR", "Month", "Day", "Time", "Hour"]].copy()
df_ldr.head()
```

Out[11]:

	LDR	Month	Day	Time	Hour
0	1023	5	19	00:00:05	0
1	1023	5	19	00:00:16	0
2	1023	5	19	00:00:27	0
3	1023	5	19	00:00:38	0
4	1023	5	19	00:00:49	0

In [12]:
```
# Calculating the mean of the LDR of a hour

grouped_ldr = df_ldr.groupby(['Month','Day','Hour'])
df_ldr_mean = grouped_ldr['LDR'].mean().reset_index()
df_ldr_mean.head()
```

Out[12]:

	Month	Day	Hour	LDR
0	5	19	0	1022.996923
1	5	19	1	1022.969325
2	5	19	2	1022.990769
3	5	19	3	1022.987730
4	5	19	4	1022.990769

In [13]:
```
# Setting the conditions that determine if the laptop screen is on, off or uncertain

conditions = [
    (df_ldr_mean['LDR'] >= 200) & (df_ldr_mean['LDR'] <= 920) ,
    (df_ldr_mean['LDR'] > 920) & (df_ldr_mean['LDR'] < 980) ,
    (df_ldr_mean['LDR'] < 200),
    (df_ldr_mean['LDR'] >= 980)
    ]

values = ['On', 'Uncertain', 'Uncertain', 'Off']
df_ldr_mean['Screen'] = np.select(conditions, values)
df_ldr_mean.head()
```

Out[13]:

	Month	Day	Hour	LDR	Screen
0	5	19	0	1022.996923	Off
1	5	19	1	1022.969325	Off
2	5	19	2	1022.990769	Off
3	5	19	3	1022.987730	Off
4	5	19	4	1022.990769	Off

In [14]:
```
# Check how many Uncertains there are

df_ldr_mean[df_ldr_mean['Screen'] == 'Uncertain'].count()
```

Out[14]:
```
Month     151
Day       151
Hour      151
LDR       151
Screen    151
dtype: int64
```

In [15]:
```
# Adding the band cells

# Importing the data from the Mi Band
miband = './datasets/BAND.csv'
```

```
df_miband = pd.read_csv(miband)
df_miband.head()
```

Out[15]:

	StartDate/Time	EndDate/Time	Steps
0	19/05/2021 00:00	19/05/2021 01:00	27.0
1	19/05/2021 01:00	19/05/2021 02:00	0.0
2	19/05/2021 02:00	19/05/2021 03:00	0.0
3	19/05/2021 03:00	19/05/2021 04:00	0.0
4	19/05/2021 04:00	19/05/2021 05:00	0.0

In [16]:
```
# Drop the EndDate/Time column as there is only one time value needed

df_bandstart = df_miband.drop('EndDate/Time', axis=1)
df_bandstart.head()
```

Out[16]:

	StartDate/Time	Steps
0	19/05/2021 00:00	27.0
1	19/05/2021 01:00	0.0
2	19/05/2021 02:00	0.0
3	19/05/2021 03:00	0.0
4	19/05/2021 04:00	0.0

In [17]:
```
# Changing the date and time to a generic writing style to match the YODL-data.
# And changing the time ('Hour') to the hour-number.

df_bandstart['Date'] = pd.to_datetime(df_bandstart['StartDate/Time'], format= '%d/%m/%Y %H:%M' , exact=False)
df_bandstart['Time'] = pd.to_datetime(df_bandstart['StartDate/Time']).dt.time
df_bandstart['Hour'] = pd.to_datetime(df_bandstart['StartDate/Time']).dt.hour
df_bandstart.head()
```

Out[17]:

	StartDate/Time	Steps	Date	Time	Hour
0	19/05/2021 00:00	27.0	2021-05-19 00:00:00	00:00:00	0
1	19/05/2021 01:00	0.0	2021-05-19 01:00:00	01:00:00	1
2	19/05/2021 02:00	0.0	2021-05-19 02:00:00	02:00:00	2
3	19/05/2021 03:00	0.0	2021-05-19 03:00:00	03:00:00	3
4	19/05/2021 04:00	0.0	2021-05-19 04:00:00	04:00:00	4

In [18]:
```
# Drop the StartDate/Time as there is a seperate column for the hour

df_band = df_bandstart.drop('StartDate/Time', axis=1)
df_band.head()
```

Out[18]:

	Steps	Date	Time	Hour
0	27.0	2021-05-19 00:00:00	00:00:00	0
1	0.0	2021-05-19 01:00:00	01:00:00	1
2	0.0	2021-05-19 02:00:00	02:00:00	2
3	0.0	2021-05-19 03:00:00	03:00:00	3
4	0.0	2021-05-19 04:00:00	04:00:00	4

In [19]:
```
# Create new colomns for month and day

df_band['Month'] = pd.to_datetime(df_band['Date']).dt.month
df_band['Day'] = pd.to_datetime(df_band['Date']).dt.day
df_band.head()
```

Out[19]:

	Steps	Date	Time	Hour	Month	Day
0	27.0	2021-05-19 00:00:00	00:00:00	0	5	19
1	0.0	2021-05-19 01:00:00	01:00:00	1	5	19
2	0.0	2021-05-19 02:00:00	02:00:00	2	5	19
3	0.0	2021-05-19 03:00:00	03:00:00	3	5	19
4	0.0	2021-05-19 04:00:00	04:00:00	4	5	19

In [20]:
```
# Only keep the relevant columns

df_band = df_band[['Month', 'Day', 'Hour', 'Steps']]
df_band.head()
```

Out[20]:

	Month	Day	Hour	Steps
0	5	19	0	27.0
1	5	19	1	0.0
2	5	19	2	0.0
3	5	19	3	0.0
4	5	19	4	0.0

In [21]:
```
# Create one dataset with all the relevant information of the LDR and Temperature

df_yodl_result = pd.merge(df_ldr_mean, df_temperature_mean, on=['Month', 'Day', 'Hour'], how = 'inner')
df_yodl_result.head()
```

Out[21]:

	Month	Day	Hour	LDR	Screen	Temperature	Humidity
0	5	19	0	1022.996923	Off	24.392308	30.744615
1	5	19	1	1022.969325	Off	24.345399	31.398773
2	5	19	2	1022.990769	Off	24.332615	32.116923
3	5	19	3	1022.987730	Off	24.323926	32.877301
4	5	19	4	1022.990769	Off	24.316615	33.000000

In [22]:
```
# Create one complete dataset with all the relevant information of one participant

df_participant5 = pd.merge(df_yodl_result, df_band, on=['Month', 'Day', 'Hour'], how = 'inner')
df_participant5.head()
```

Out[22]:

	Month	Day	Hour	LDR	Screen	Temperature	Humidity	Steps
0	5	19	0	1022.996923	Off	24.392308	30.744615	27.0
1	5	19	1	1022.969325	Off	24.345399	31.398773	0.0
2	5	19	2	1022.990769	Off	24.332615	32.116923	0.0
3	5	19	3	1022.987730	Off	24.323926	32.877301	0.0
4	5	19	4	1022.990769	Off	24.316615	33.000000	0.0

In [23]:
```
df_participant5 = df_participant5[['Month', 'Day', 'Hour', 'LDR', 'Screen', 'Temperature', 'Humidity', 'Steps']]
df_participant5.head()
```

Out[23]:

	Month	Day	Hour	LDR	Screen	Temperature	Humidity	Steps
0	5	19	0	1022.996923	Off	24.392308	30.744615	27.0
1	5	19	1	1022.969325	Off	24.345399	31.398773	0.0
2	5	19	2	1022.990769	Off	24.332615	32.116923	0.0
3	5	19	3	1022.987730	Off	24.323926	32.877301	0.0
4	5	19	4	1022.990769	Off	24.316615	33.000000	0.0

In [24]:
```
# Export the dataframe of participant 5 to an excel sheet.

df_participant5.to_excel("participant5.xlsx")
```

Appendix C. Python code: visualization.

jupyter visualisations Last Checkpoint: an hour ago (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Run Markdown

In [1]:
```
AUTHOR_NAME = 'Hanna van Rixtel'
AUTHOR_ID_NR = '1454862'
AUTHOR_DATE = '2021-06-17'

AUTHOR_NAME, AUTHOR_ID_NR, AUTHOR_DATE
```

Out[1]:
```
('Hanna van Rixtel', '1454862', '2021-06-17')
```

In [2]:
```
import numpy as np
import pandas as pd
import datetime

# next command ensures that plots appear inside the notebook
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns
sns.set()  # set Seaborn defaults
plt.rcParams['figure.figsize'] = 12, 5  # default hor./vert. size of plots, in inches
plt.rcParams['lines.markeredgewidth'] = 1  # to fix issue with seaborn box plots; needed after import seaborn

import warnings
warnings.filterwarnings('ignore')
```

participant5.csv

This is temperature, screen time and stepcount data collected by temperature/humidity sensor (DHT11), LDR sensor and a Xiaomi Mi Band 5 of participant 5 of two weeks. This dataset is already cleaned.

Field	Description
Participant	indicating to which participant this dataset belongs
Month	the month of the entry
Day	the day of the entry
Hour	the hour of the entry
Screen	indicating if the screen is on, off or uncertain in that hour
Temperature	the temperature (in °C) recorded in that hour in the participant's room
Steps	the numbers of steps recorded in that hour

Next datasets consist of the same kind of data but for a different participant: participant1, participant2, participant3 and participant4.

participant5-numbers.csv

This is temperature, screen time and stepcount data collected by temperature/humidity sensor (DHT11), LDR sensor and a Xiaomi Mi Band 5 of participant 5 of two weeks. This dataset is already cleaned. This time, information about the screen is provided in a different way.

Field	Description
Participant	indicating to which participant this dataset belongs
Month	the month of the entry
Day	the day of the entry
Hour	the hour of the entry
Screen	indicating if the screen is on, off or uncertain in that hour by showing 1(on), 2(off), 3(uncertain)
Temperature	the temperature (in °C) recorded in that hour in the participant's room
Steps	the numbers of steps recorded in that hour

In [3]:
```
participant5 = './alldatasets/participant5.csv'
```

In [4]:
```
df_participant5 = pd.read_csv(participant5, sep = ';')
df_participant5.head()
```

Out[4]:

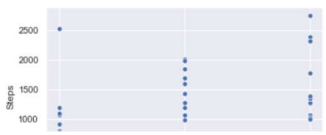
	Participant	Month	Day	Hour	Screen	Temperature	Steps
0	5	5	19	0	Off	24	27
1	5	5	19	1	Off	24	0
2	5	5	19	2	Off	24	0
3	5	5	19	3	Off	24	0
4	5	5	19	4	Off	24	0

In [5]:
```
df_participant5.info()
```

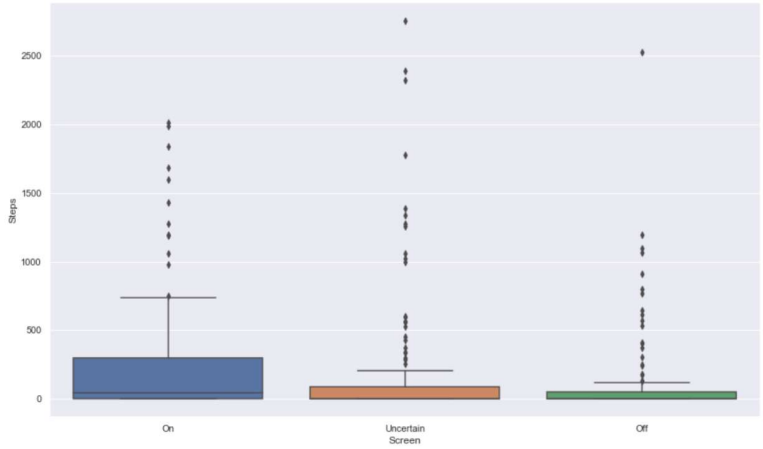
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 333 entries, 0 to 332
Data columns (total 7 columns):
Participant    333 non-null int64
Month          333 non-null int64
Day            333 non-null int64
Hour           333 non-null int64
Screen         333 non-null object
Temperature    333 non-null int64
Steps          333 non-null int64
dtypes: int64(6), object(1)
memory usage: 18.3+ KB
```

Exploration visualisations

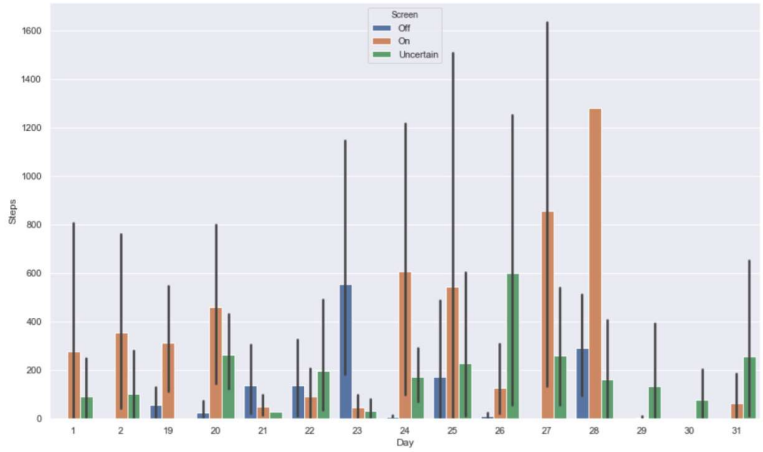
In [6]:
```
sns.scatterplot(x='Screen', y='Steps', data= df_participant5);
```



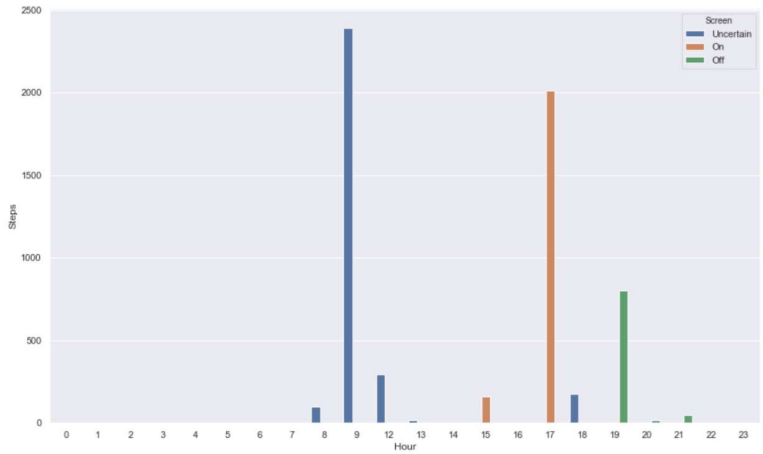
In [7]:
```
plt.figure(figsize=(15,9))
sns.boxplot(x='Screen', y='Steps', data=df_participant5, order=['On', 'Uncertain', 'Off']);
```



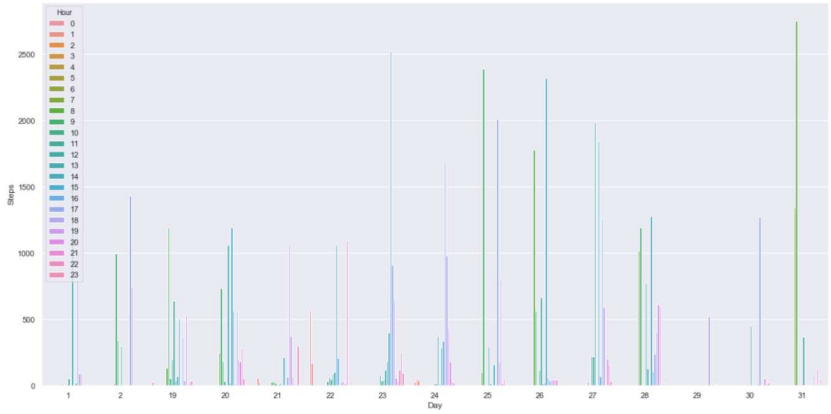
In [8]:
```
plt.figure(figsize=(15,9))
sns.barplot(x='Day', y='Steps', hue='Screen', data=df_participant5);
```



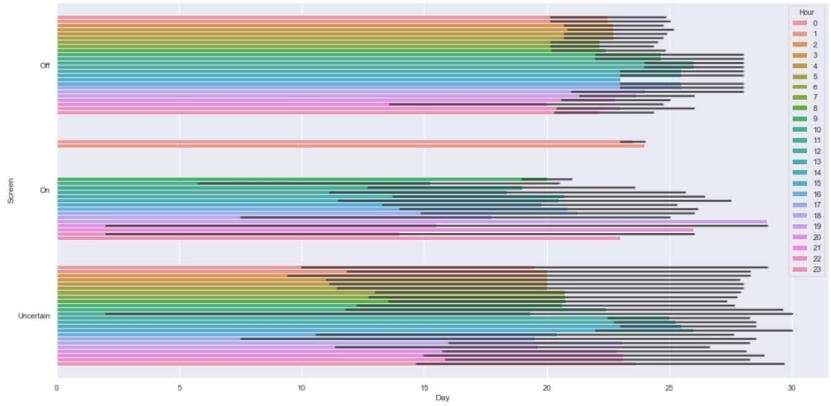
In [9]:
```
plt.figure(figsize=(15,9))
sns.barplot(x='Hour', y='Steps', hue='Screen', data=df_participant5[df_participant5['Day']==25]);
```



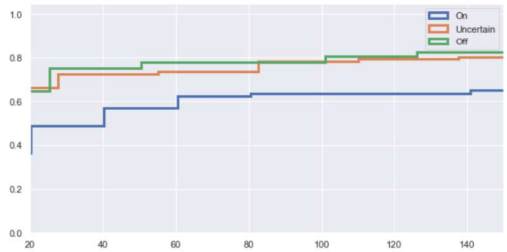
In [10]:
```
plt.figure(figsize=(20,10))
sns.barplot(x='Day', y='Steps', hue='Hour', data=df_participant5);
```



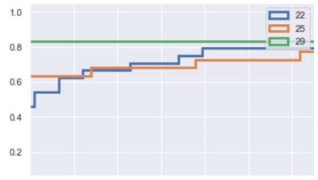
In [11]:
```
plt.figure(figsize=(20,10))
sns.barplot(x='Day', y='Screen', hue='Hour', data=df_participant5);
```

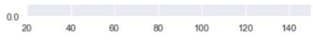


In [12]:
```
plt.figure(figsize=(10,5))

r_on= df_participant5[df_participant5['Screen']=='On']['Steps']
r_uncertain= df_participant5[df_participant5['Screen']=='Uncertain']['Steps']
r_off = df_participant5[df_participant5['Screen']=='Off']['Steps']

ax=r_on.hist(histtype='step', bins=100,linewidth=3.0,cumulative=True,normed=True);
r_uncertain.hist(histtype='step', bins=100,linewidth=3.0,cumulative=True,normed=True);
r_off.hist(histtype='step',  bins=100,linewidth=3.0,cumulative=True,normed=True);

plt.legend(['On','Uncertain','Off']);
ax.set_xlim(20,150);
```

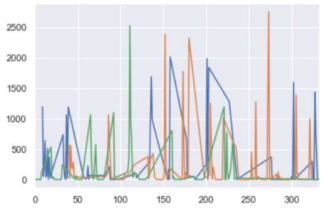


In [13]:
```
r_22= df_participant5[df_participant5['Day']==22]['Steps']
r_25= df_participant5[df_participant5['Day']==25]['Steps']
r_29 = df_participant5[df_participant5['Day']==29]['Steps']

ax=r_22.hist(histtype='step', bins=100,linewidth=3.0,cumulative=True,normed=True);
r_25.hist(histtype='step', bins=100,linewidth=3.0,cumulative=True,normed=True);
r_29.hist(histtype='step',  bins=100,linewidth=3.0,cumulative=True,normed=True);

plt.legend(['22','25','29']);
ax.set_xlim(20,150);
```





In [14]:
```
#fig, ax = plt.subplots(1, 1,figsize=(15,10))

on = df_participant5['Screen'] =='On'
uncertain = df_participant5['Screen']=='Uncertain'
off = df_participant5['Screen']== 'Off'

df_participant5[on]['Steps'].plot(color="#4C72B0")
df_participant5[uncertain]['Steps'].plot(color="#DD8452")
df_participant5[off]['Steps'].plot(color="#55A868");
```



In [15]:
```
numbers = './datasets/participant5-numbers.csv'
```

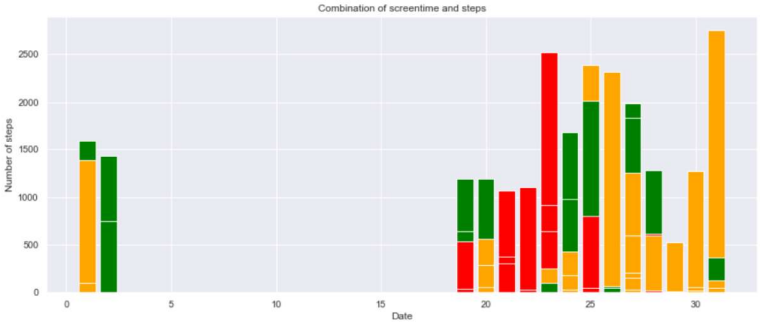
In [16]:
```
df_participant5num = pd.read_csv(numbers, sep = ';')
df_participant5num .head()
```

Out[16]:

	Month	Day	Hour	LDR	Screen	Temperature	Humidity	Steps
0	5	19	0	1022,996923	2	24,39230769	30,74461538	27
1	5	19	1	1022,969325	2	24,34539877	31,39877301	0
2	5	19	2	1022,990769	2	24,33261538	32,11692308	0
3	5	19	3	1022,98773	2	24,32392638	32,87730061	0
4	5	19	4	1022,990769	2	24,31661538	33	0

In [17]:
```
y =df_participant5num['Screen']

col = []
for val in y:
    if val == 3: #uncertain
        col.append('orange')
    if val == 1: #ON
        col.append('green')
    elif val == 2: #off
        col.append("red")
```

In [18]:
```
height = df_participant5num['Steps']
bars = df_participant5num['Day']
plt.figure(figsize=(15,6))

plt.bar(bars, height, color= col)
plt.title('Combination of screentime and steps')
plt.xlabel('Date')
plt.ylabel('Number of steps')

# Show graphic
plt.show()
```



In [19]:
```
# Participant 5 was already imported

participant1 = './alldatasets/participant1.csv'
participant2 = './alldatasets/participant2.csv'
participant3 = './alldatasets/participant3.csv'
participant4 = './alldatasets/participant4.csv'
```

In [20]:
```
df_participant1 = pd.read_csv(participant1, sep = ';')
```

In [21]:
```
df_participant2 = pd.read_csv(participant2, sep = ';')
```

In [22]:
```
df_participant3 = pd.read_csv(participant3, sep = ';')
```

In [23]:
```
df_participant4 = pd.read_csv(participant4, sep = ';')
```

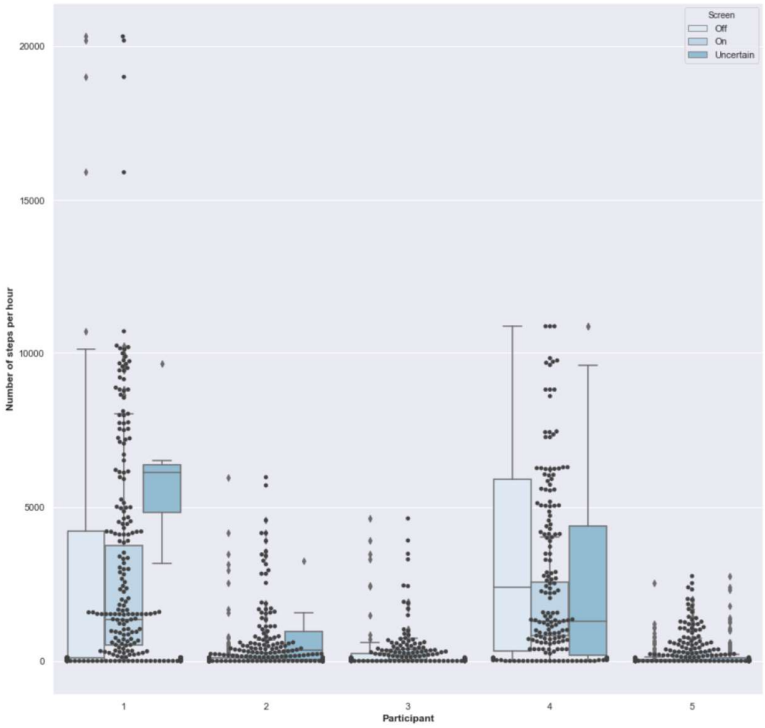
In [24]:
```
df_combined = pd.concat([df_participant1, df_participant2, df_participant3, df_participant4, df_participant5])
df_combined.head()
```

Out[24]:

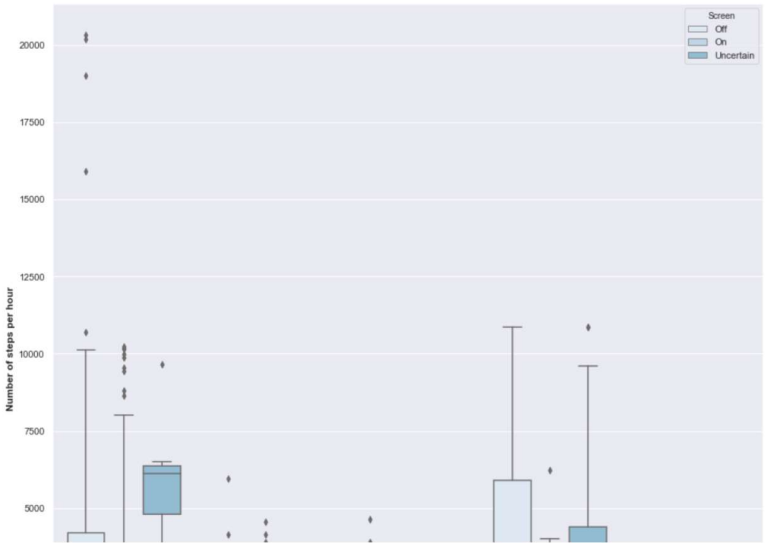
	Participant	Month	Day	Hour	Screen	Temperature	Steps
0	1	5	19	0	Off	21,85	0.0
1	1	5	19	1	Off	21,45784615	0.0

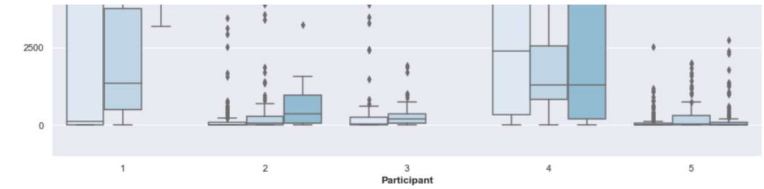
2	1	5	19	2	Off	21,17822086	0.0
3	1	5	19	3	Off	20,92	0.0
4	1	5	19	4	Off	20,67116564	0.0

In [25]:
```
plt.figure(figsize=(15,15))

sns.set_palette("Blues")
sns.boxplot(x='Participant', y='Steps', hue='Screen', data=df_combined);
sns.swarmplot(x='Participant', y='Steps', data=df_combined, color=".25")

# back to the default colours
sns.set_palette("deep")

plt.xlabel('Participant', fontweight="bold")
plt.ylabel('Number of steps per hour', fontweight="bold");
```

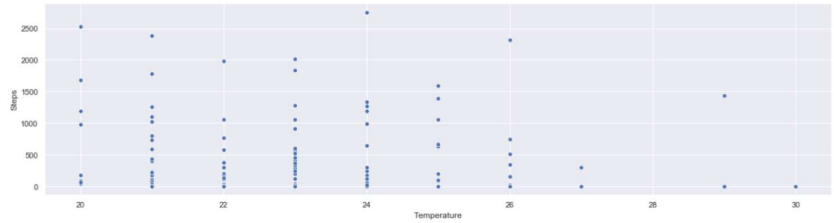


In [26]:
```
plt.figure(figsize=(15,15))

sns.set_palette("Blues")
sns.boxplot(x='Participant', y='Steps', hue='Screen', data=df_combined);

# back to the default colours
sns.set_palette("deep")

plt.xlabel('Participant', fontweight="bold")
plt.ylabel('Number of steps per hour', fontweight="bold");
```

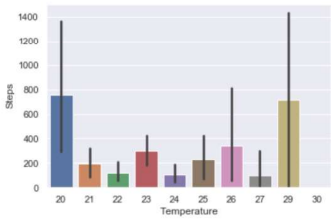




In [27]:
```
plt.figure(figsize=(20,5))
sns.scatterplot(x ='Temperature', y='Steps', data= df_participant5);
```

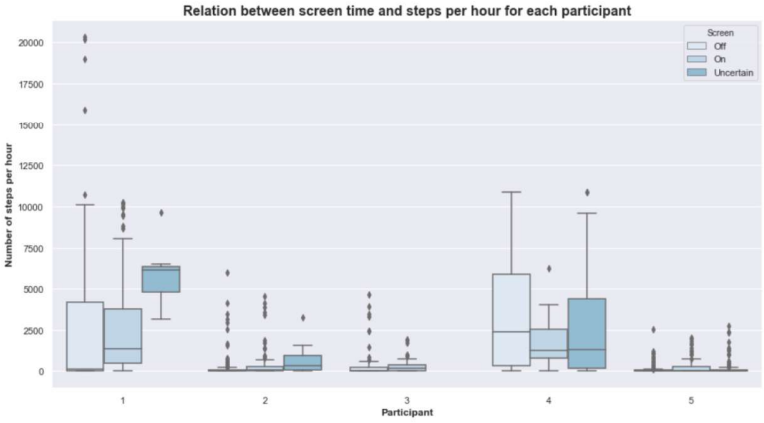


In [28]:
```
sns.barplot(x='Temperature', y='Steps', data=df_participant5);
```

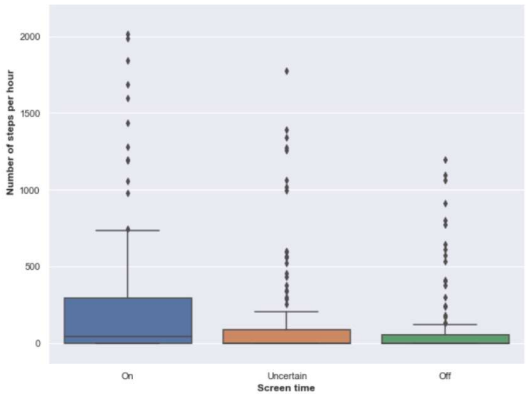


Final visualisation

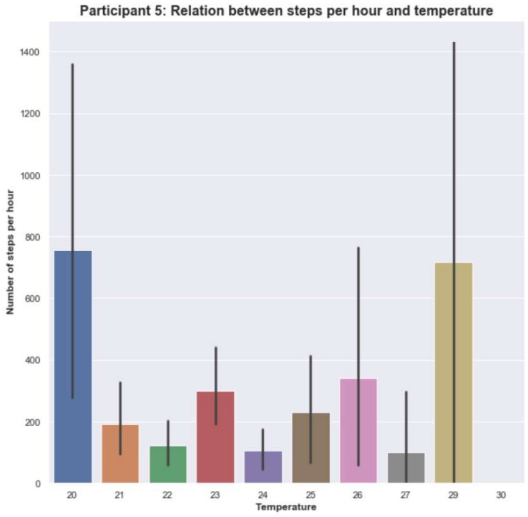
In [29]:
```
plt.figure(figsize=(15,8))

sns.set_palette("Blues")
sns.boxplot(x='Participant', y='Steps', hue='Screen', data=df_combined);

# back to the default colours
sns.set_palette("deep")

plt.title('Relation between screen time and steps per hour for each participant', fontsize = 16, fontweight="bold")
plt.xlabel('Participant', fontweight="bold")
plt.ylabel('Number of steps per hour', fontweight="bold");

plt.savefig('AllParticipants.jpg')
```



In [30]:
```
plt.figure(figsize=(10,10))
sns.boxplot(x='Screen', y='Steps', data=df_participant5, order=['On', 'Uncertain', 'Off'])

plt.title('Participant 5: Relation between steps per hour and screen time', fontsize = 16, fontweight="bold")
plt.xlabel('Screen time', fontweight="bold")
plt.ylabel('Number of steps per hour', fontweight="bold");

plt.savefig('Participant5-02.jpg')
```



In [31]:
```
plt.figure(figsize=(10,10))

sns.barplot(x='Temperature', y='Steps', data=df_participant5)
plt.title('Participant 5: Relation between steps per hour and temperature', fontsize = 16, fontweight="bold")
plt.xlabel('Temperature', fontweight="bold")
plt.ylabel('Number of steps per hour', fontweight="bold");

plt.savefig('Participant5-01.jpg')
```



Appendix D. Metadata.

YODL.csv

This is temperature and screen time data collected by a temperature/humidity sensor (DHT11) and a LDR sensor connected to a participant's laptop screen.

Field	Description
Date/Time	the date and time of the entry (GMT+1)
Temperature	the temperature (in °C) at that specific time in the participant's room
Humidity	the humidity at that specific time in the participant's room
LDR	the LDR value of the a LDR connected to the participant's laptop screen

BAND.csv

This is stepcount data collected by a Xiaomi Mi Band 5, worn for two weeks by one participant.

Field	Description
StartDate/Time	the start of this measurement (GMT+1)
EndDate/Time	the end of this measurement (GMT+1)
Steps	the numbers of steps recorded between the Start Time and End Time

participant5.csv

This is temperature, screen time and stepcount data collected by temperature/humidity sensor (DHT11), LDR sensor and a Xiaomi Mi Band 5 of participant 5 of two weeks. This dataset is already cleaned.

Field	Description
Participant	indicating to which participant this dataset belongs
Month	the month of the entry
Day	the day of the entry
Hour	the hour of the entry
Screen	indicating if the screen is on, off or uncertain in that hour
Temperature	the temperature (in °C) recorded in that hour in the participant's room
Steps	the numbers of steps recorded in that hour

Next datasets consist of the same kind of data but for a different participant: participant1, participant2, participant3 and participant4.

participant5-numbers.csv

This is temperature, screen time and stepcount data collected by temperature/humidity sensor (DHT11), LDR sensor and a Xiaomi Mi Band 5 of participant 5 of two weeks. This dataset is already cleaned. This time, information about the screen is provided in a different way.

Field	Description
Participant	indicating to which participant this dataset belongs
Month	the month of the entry
Day	the day of the entry
Hour	the hour of the entry
Screen	indicating if the screen is on, off or uncertain in that hour by showing 1(on), 2(off), 3(uncertain)
Temperature	the temperature (in °C) recorded in that hour in the participant's room
Steps	the numbers of steps recorded in that hour

Appendix E. Boxplots per participant.