**Phase 5; project submission**

**Measure energy consumption**

**Abstract: The task of reducing the energy footprint of IT devices and software has been a challenge for Green IT research. Monitoring approaches have primarily focused on measuring the energy consumption of the hardware components of computing devices. The use of applications or software on our**

**computer systems consumes energy and it also affects how various hardware components and system resources consume energy. Consequently, running web browsers applications will utilise considerable energy and battery consumption. In this research, we have run different types of experiments which involve the use of several measuring tools. Firsly, a joulemeter is used to monitor**

**(and measure) the power consumed by the hardware and software while running web-based and stand-alone applications on several devices. Additionally, the tablet in-built battery status checker is used to measure the battery consumption when web-based applications are run on the device.**

**Introduction:**

Green :computing technology focuses on the efficient use of computing resources. In computing devices such as laptops, smartphones, tablets, orother mobile devices,energy consumption is the top priority because they are run on battery, with limitedlifespan, as their source of power (Banerjee et al. 2007).With the increasing complexity of IT equipment, the energy consumption rate of these devicessystem also increases (Silven and Jyrkka,2007). Most portable mobile device users are conscious of the energy usage by these devices and consequently, they look for ways through which the lifespan of the battery can be extended to serve them longer (Rahmati et al. 2007).

**Aim:**

The aim of this paper is to discuss the results for several investigations conducted on the energy (and battery) consumption for running web-based and standalone applications on Windowsand IOS portable computing devices. The following objectives will help to achieve this aim:

•Research Objective 1:To conduct experiments on the measurement of energy consumed for running youtube videos in different web browsers (e.g. Google Chrome, Mozilla Firefox, etc…) on Windows (i.e. laptops), and IOS machines (i.e. tablet);

• Research Objective 2: To conduct experiments on the measurement of energy consumed for playing audio and video files on several media players for windows (on a laptop); 

•Research Objective 3: To conduct analyses on data collected in Research Objectives 1 and 2

**Literature Review:**

The Smart2020 report (The Climate Group and GeSI, 2008) predicts an increasing trend of BAU CO2 emissions for the ICT industry. The emissions growth rate for three ICT categories (end-user devices, telecommunication and networks, and data centers) is expected to decrease from 6.1% 3.8%. By 2020, the ICT industry’s footprint is expected to rise to 1.3 GtCO2e (equivalent to 2.3% of global emissions by 2020). The PC (e.g. desktops, laptops, etc.) footprint (due to its embodied and usage emissions) is the highest (60%) followed by printers (18%), peripherals (13%), smartphones (10%), and tablets (1%). It is estimated that the footprint of end-user devices will grow at 2.3 percent per year to reach 0.67 GtCO2e in 2020 and thus, energy efficiency improvements in these devices and their proper usage are essential for reducing their overall footprint.

**Energy Consumption of software:**

Green and sustainable software is a software product that has the smallest possible economic, societal, ecological impact as well as impact on human beings (Ahmed, et al., 2014). This has led to the introduction of various programmes and initiatives that encourages energy efficient software such as green software engineering and Eco-design software.

According to the Greenhouse Gas Protocol (2012), applications are executed with an OS. They affect the power consumption of a device due to data requests and processing. Managing energy requires accurate measurement of the energy available and consumed by a system.

This involves monitoring or estimating the resource and energy consumption of hardware and software (Noureddine, et al., 2013). However, a device’s power consumption is subjected to the type of application and the task being performed which is evident in our experimental results presented in Section 4 of this paper.

In order to reduce the overall power consumption for a web-based or standalone task, it will be necessary to provide users with an insight of the power consumption of the different web-based browser applications (e.g. Google Chrome, Internet Explorer, Mozilla Firefox, Safari, etc…) and also the resource hungry nature of many applications such as movie player and games

**EEnergy Consumption of Media Player:**

* Modern technologies incorporate a number of power management features to reduce power waste. Dynamic Voltage and Frequency Scaling (DVFS) can enable the CPU speed to be dynamically varied based on the workload which leads to a reduced power consumption during periods of low utilization (Liu, et al., 2008). The energy-aware dynamic voltage scaling technique has been used to reduce energy consumption in portable media players (Yang & Song, 2009).
* This scheme showed a relationship between frame size and decoding time. These two cited work merely discuss how energy consumption can be reduced using various techniques, but have not measured the actual amount of energy being consumed by the application.
* However, the energy consumption of Windows Media Player has been measured using the EEcoMark v2 tool (EecoMark, 2011) but the empirical details of the measurement have not been explicitly discussed. Media playback application power consumption has been analysed by Sabharwal (2011) using windows event tracing.
* Event tracing does not seem to be an appropriate method for measuring energy consumption because the process itself may have impact on the results. A comparative analysis of energy consumption of media players has been conducted by Techradar (2010).
* The energy consumption is monitored by playing a DVD on Windows Media Player (WMP) and VLC Media Player. Their research results show that the VLC Media Player is more energy efficient than Windows Media Player. However, the cited work has not mentioned which tool has been used for measurement and additionally, the experiment procedures have not been explitcitly discussed

**Metrics, Measure and Tools for Energy Consumption:**

**Metrics**

* Generally, software does not directly consume energy. However, running the software involves the hardware which consumes energy.
* Therefore, the resource usage metric such as the CPU usage, memory and disk usage are used as the measuring criteria
* It is important to analyse the energy efficiency of software by observing the amount of resource utilized versus the useful work performed. To measure the power consumption of a system, the power consumed by individual PC components must be measured.
* Therefore a system wide resource utilization monitoring technique at the user level seems to be more appropriate.

**Measure and Tools**

* There is a wide range of methods for measuring the energy consumption of a computer system. Generally, the measurement of energy consumption is grouped into three categories hardware, software and power models .
* Measuring energy consumption of hardware using devices such as wattsup1 and the method described by McIntire and colleagues (2007) yields a precise value. PowerScope is a tool that uses a multi-meter to measure the energy consumption of applications .
* This method is more precise because it can determine the energy consumption of a specific process and even procedures within the process. However, these methods have some limitations.
* It can only monitor hardware devices, not flexible, requires additional hardware and the value may fluctuate due to electro-mechanical issues.
* It is also difficult to upgrade to a more newer and precise monitoring without replacing the entire hardware.
* Power models are used to calculate the energy consumption of hardware and software. Kansal and Zhao (2008) use a generic automated tool to profile the energy usage of various resources components used by an application.
* This method is either too generic or coarse-grained and it is platform dependent (Seo, et al., 2007). The model proposed by Lewis and colleagues (2012) is an integrated model for the calculation of a system’s energy consumption.
* More promising approaches are software energy measurement using energy application profiler . In their contribution, Varrol and Heiser (2010) use Openmoko Neo Freerunner to decompose the energy consumption of each resource of a system.
* PowerAPI is an Application Programming Interface (API) used for monitoring the real time energy consumption of applications at the granularity of system process . PowerAPI can also be used to estimate the energy consumption of a running process for hardware resources e.g. CPU or for hard disk or for both and many more other resources
* Energy consumption estimation in PowerAPI distinguishes the energy consumption for hardware resources and software blocks of codes. pTop is a process-level power profiling tool which provides information on the power consumption of the running processes in joules
* It gives the power consumption values for the CPU, computer memory, hard disk and the network interface for each process. The energy consumed by an application is the sum of energy consumed by individual resources in addition to energy consumed by the interaction of these processor
* The windows version also uses the windows API to perform the same task. Intel energy checker is a software development kit SDK with the capability to provide the Application Programming Interface (API) required to define, measure, and share energy efficiency data (Intel, 2010). The SDK is developed with the intention to facilitate

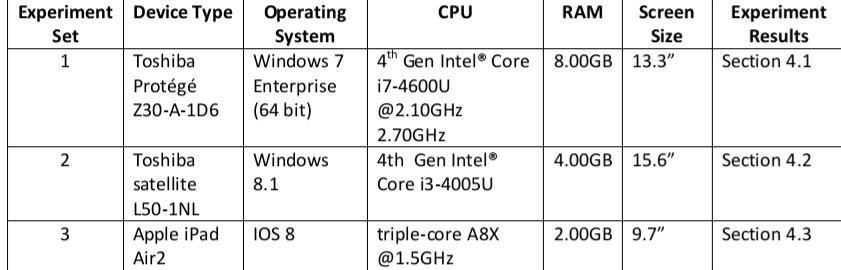
**Methodolog**

There are three sets of physical experiments conducted to investigate the relationship between Various applications and their energy (and battery) consumption. Their respective results are Presented in: (i) Section 4.1 – web browser applications and their energy consumption (for a laptop); (ii) Section 4.2 – web browser applications and their battery consumption; (iii) Section 4.3 – media Players for Windows and their energy consumption (for a laptop).

**Experiment Setup**

**Hardware devices**

The hardware devices used for the 3 sets of experiments have been tabulated in Table 1



**Measuring Tool (Software);**

Joulemeter 1.22

Is used for experiment sets 1 and 2 while an inbuilt battery status checker is

Employed for experiment set 3. There are some specific guidelines which are adhered to when

Calibrating the Joulemeter. They are: (i) ensure that the laptop battery is fully charged (or more than

50%); (ii) automatic or manual calibration of the energy model. It is necessary to ensure that other

Applications or programs are not running while the automatic calibration is in progress. Detailed

Instructions on using the Joulemeter are found in this user manual .

**Experimental Procedures**

**Experiment Set 1: To investigate the energy consumption of several web browser applications in**

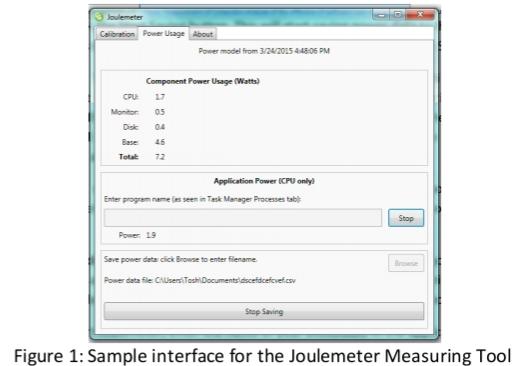
**Windows (the sample interface is shown in Figure 1)**

**Experimental Steps**

1. **Create a csv file for saving the real time power consumption data via Joulemeter (by clicking**

**On the browse button);**

1. **Click on the start saving button;**
2. **Click on the start button to run the application in Google Chrome**
3. **Click on the stop saving button to end the application;**
4. **Repeat the above steps for 9 times;**
5. **Repeat all the above steps for each of the following web browser**

****

The constants of this experiment are:

1. The wifi network used is eudroam;
2. Constant environment (experiment is carried out within the same office throughout the

Entire experiment);

1. The time for all the experiments is from 1200 -1600 for Day 1 and Day 2.

The limitations of the experiments are:

1. The Joulemeter only monitors the energy consumption of the client machine;
2. Human inconsistency involved when clicking on the essential buttons
3. Technical inconsistency which rendered several of the experiments as errors

**Results and Discussion**

This section will discuss the results of the data analysis for the three sets of experiments discussed

Above. The Joulemeter monitored raw data is for the time stamp (in ms), power consumption (in

Watts) for each component: CPU, monitor, disk, base and the application. The formula used to

Calculate the energy consumption by each component is: Energy (J) = Power (W) x Time (s). The

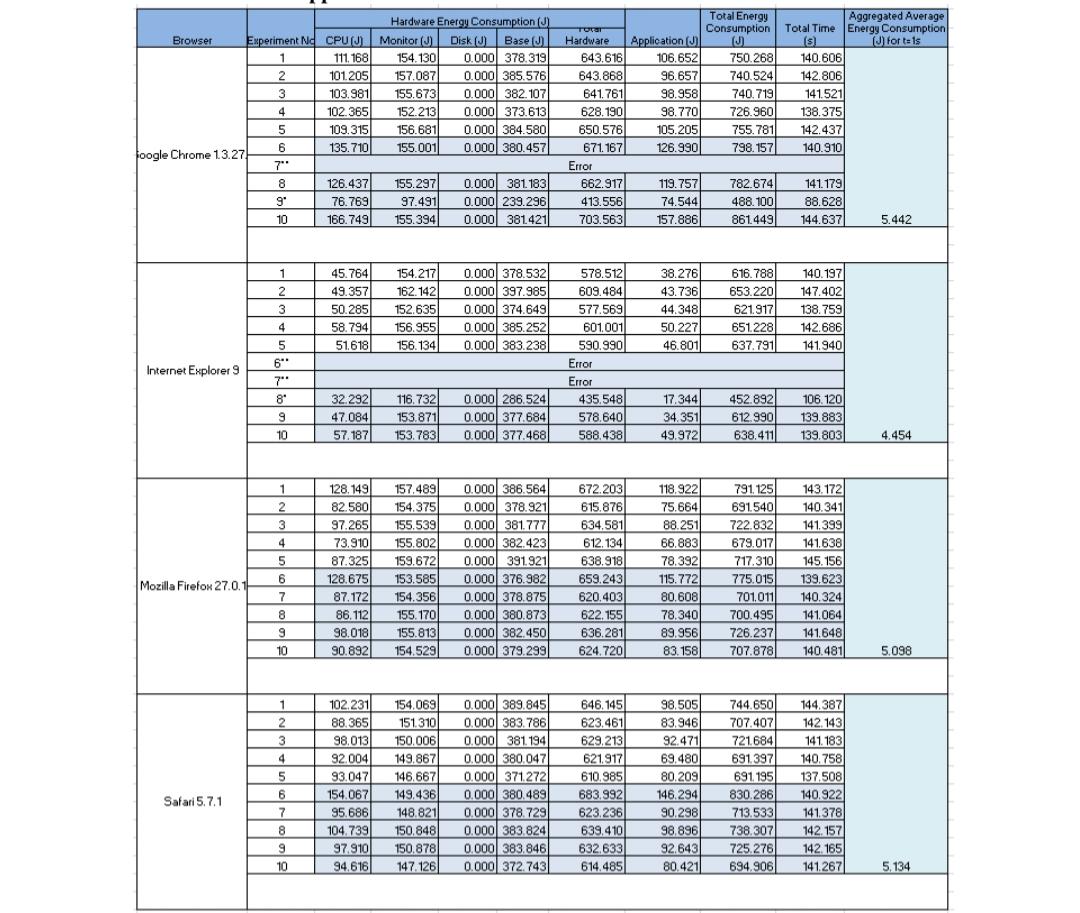
Results of the calculation for all the experiments runs are shown in Table 2. Note that the data is

Cleansed so as to omit records with application power consumption = 0W. If the number of

Remaining records > 50% of the raw data then the cleansed readings of the csv file will be included in

The data analysis. However, if it is otherwise, then the experiment is considered an error.

**Web Browser Applications for Windows**

****

*The hardware and application energy consumption for running several web browser Applications*

*Table 3 depicts the aggregated data for all the experiments conducted for each web browser: Google*

*Chrome, Internet Explorer, Mozilla Firefox, and Safari. However, in order to provide a fair*

*comparison among the web browsers, the time for running the application will have to be set to 1s*

*(i.e. t = 1s) Consequently, the corresponding energy consumption for each component will have to*

*be normalised for t = 1s (see Table 4). Based on the results shown in Table 4, it seems that Internet*

*Explorer 9 consumes the least energy on laptops, followed by Mozilla Firefox and Safari while Google*

*Chrome seems to be the highest energy consumer. These results are consistent with experiments*

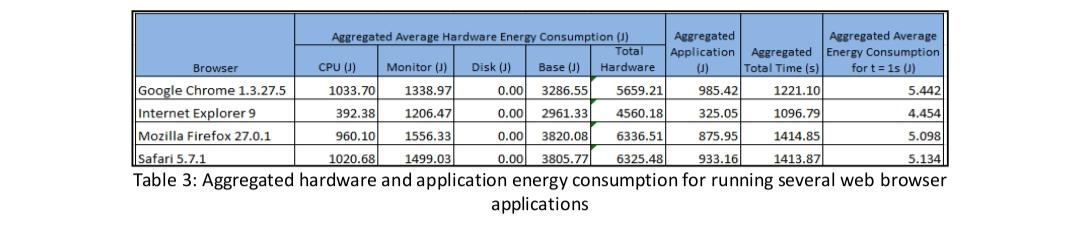
*Conducted by the Center for Sustainable Energy Systems at Fraunhofer USA4, which compare the*

*Energy consumption of Internet Explorer 10, Mozilla Firefox and Google Chrome on laptops and*

*Desktops. Their results reveal that Google Chrome consumes the highest amount of energy followed*

*By Mozilla Firefox. The conclusion drawn by them is that Internet Explorer seems to be the most*

*Energy efficient web browser.*

**

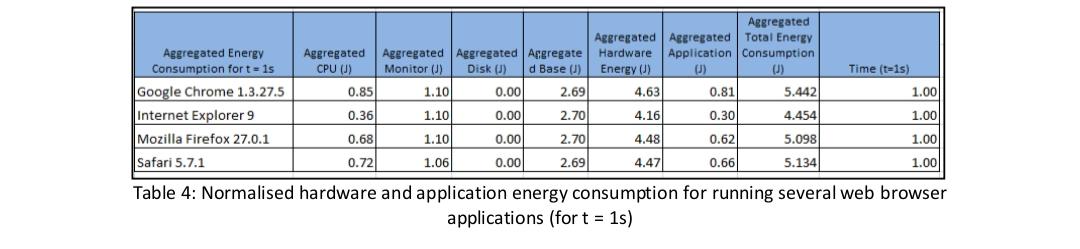
**

Figure 2 is drawn based on the normalised values in Table 4. The normalised aggregated monitor and

Base energy consumption values seem to be similar for all the web browsers. The CPU energy

Consumed by Google Chrome seems to be the highest while Internet Explorer seems to be the

Lowest. On the other hand, the CPU energy consumption for Mozilla Firefox and Safari seems to be

Similar

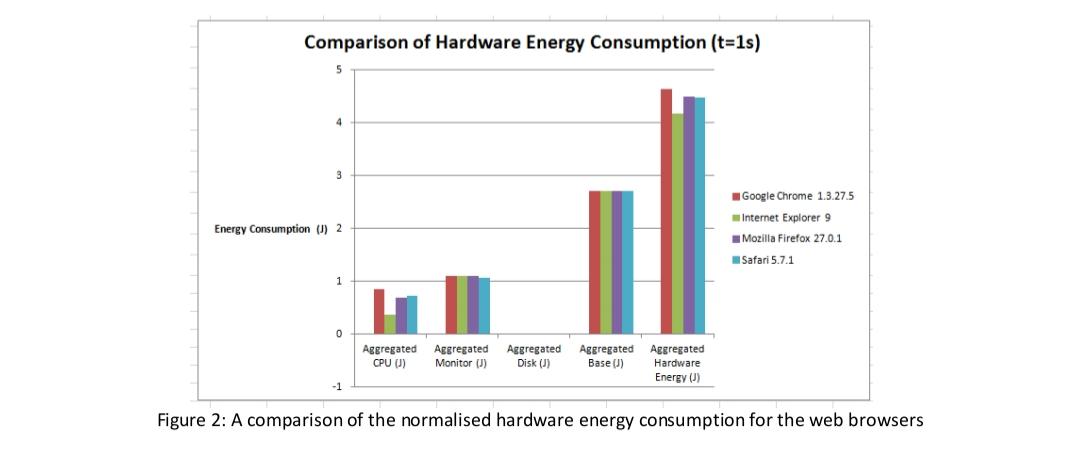


Figure 3 depicts the energy consumption by the hardware and application for each web browser. It

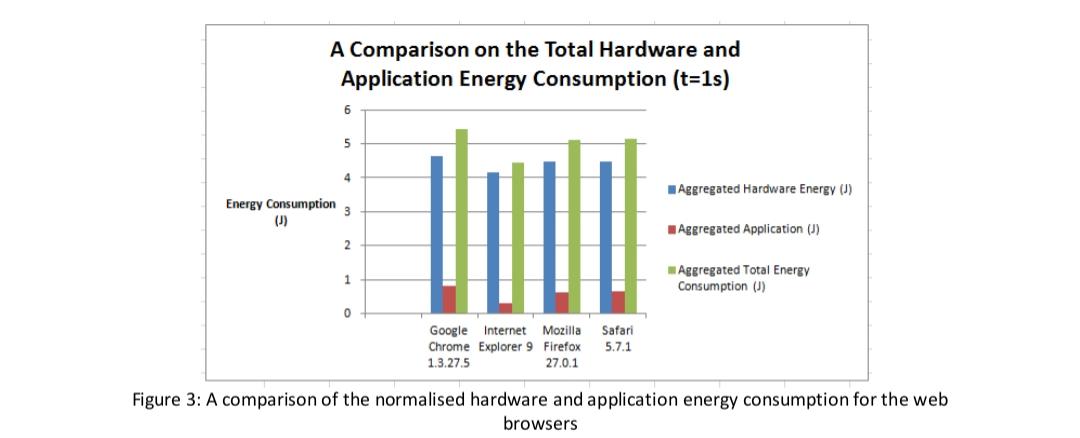
Can be seen that the energy consumed by the application is very much less compared to the energy

Consumed by the hardware that runs the application. In order to reduce the overall energy

Consumption, a further investigation on the interface as well as processes that occur between the

Software and hardware will have to be conducted and optimized. The energy consumption patterns

For the various web browsers are consistent with that for the hardware.



**Energy Consumption ForecastingImport**

## **Import relevant python packages**



## **Next let's load all the packages we will need for analysis**

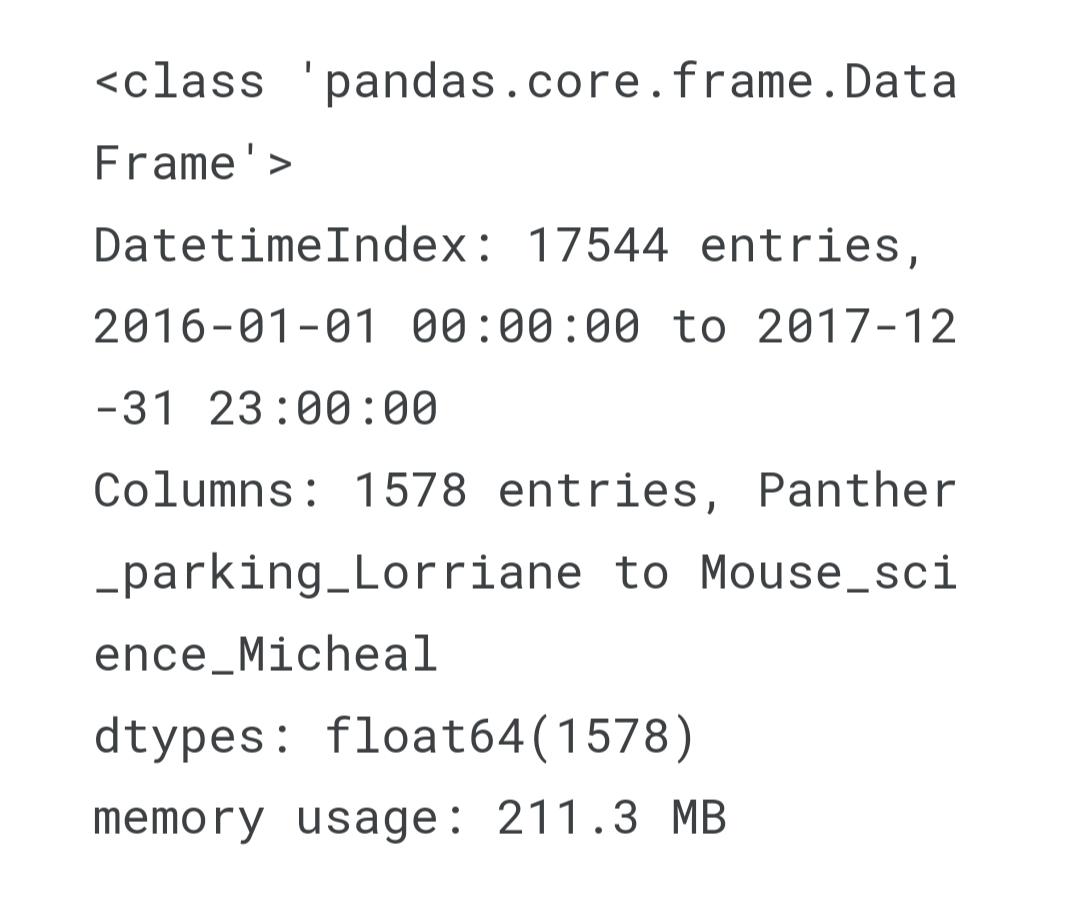
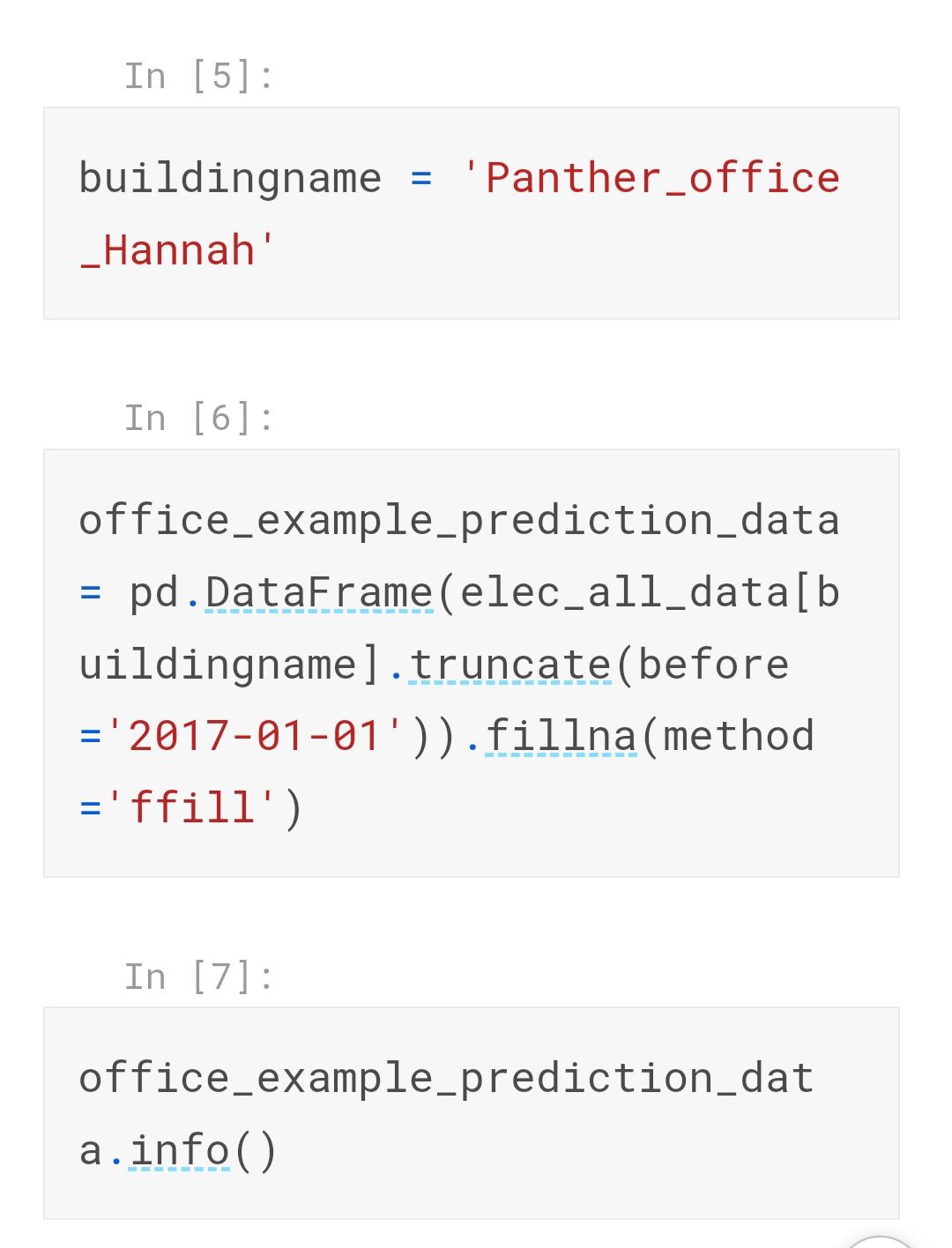
## 

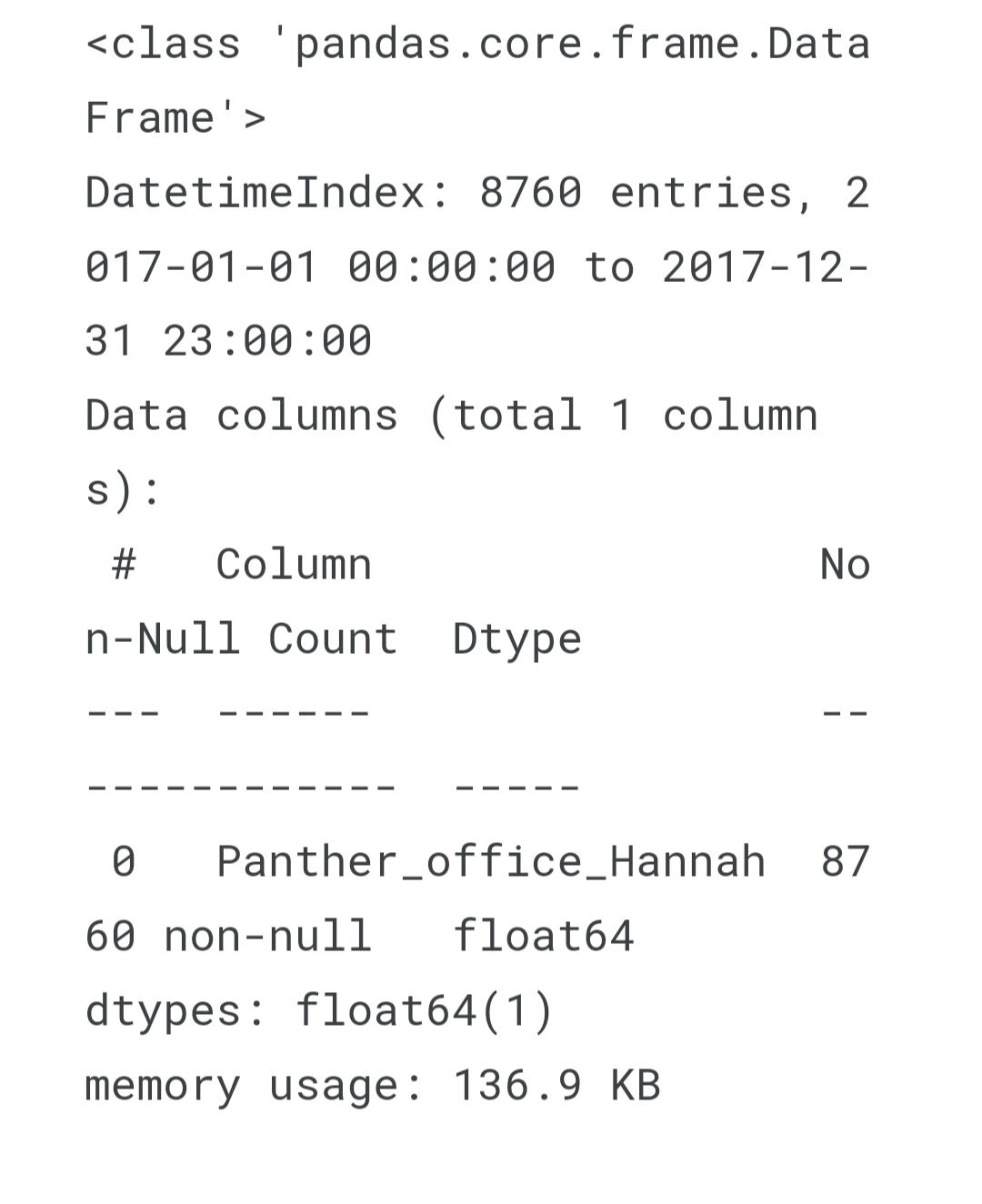
## 

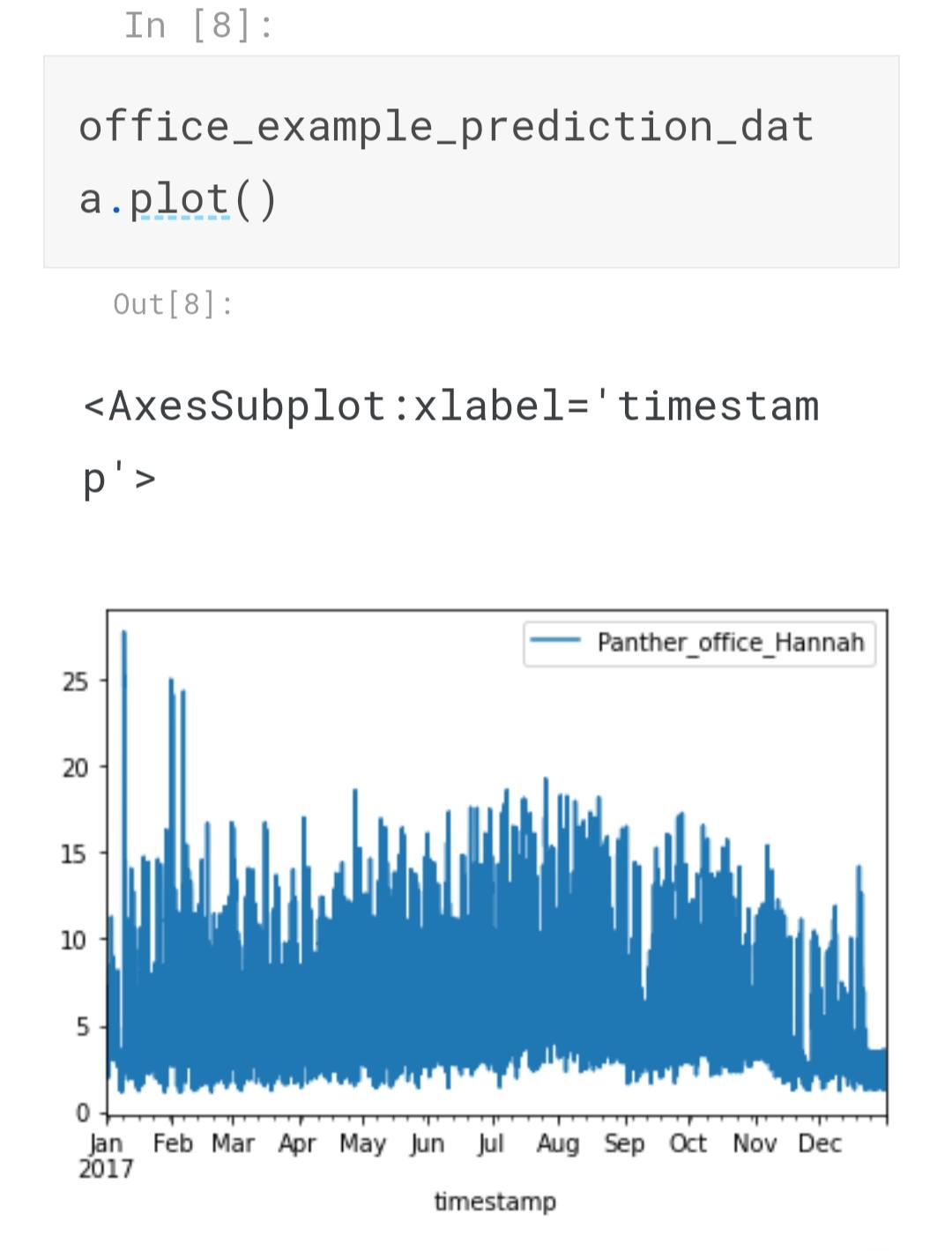
Load electricity data and weather data

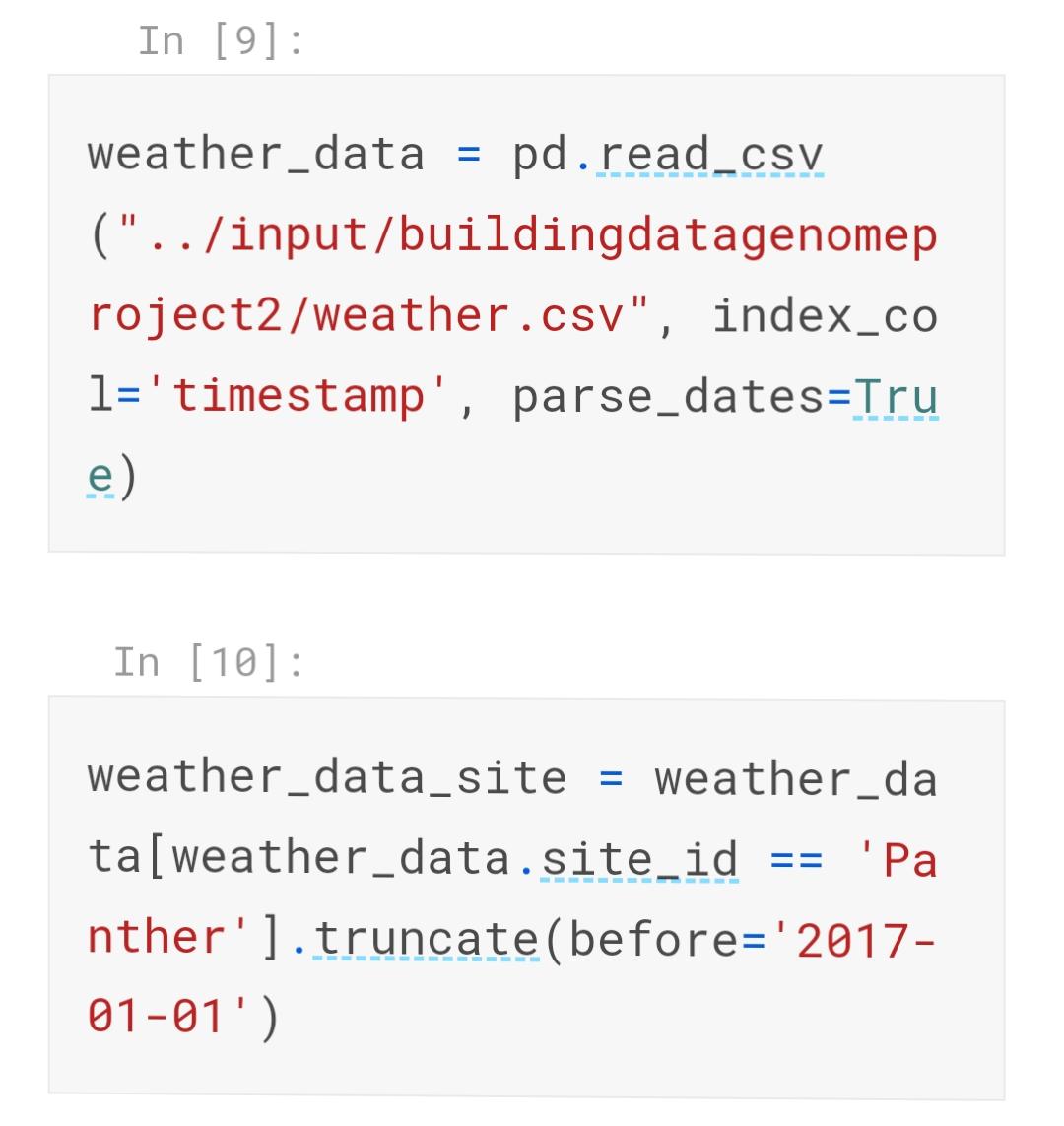
First we can load the data from the BDG in the same as our previous weather analysis influence notebook from the Construction Phase videos

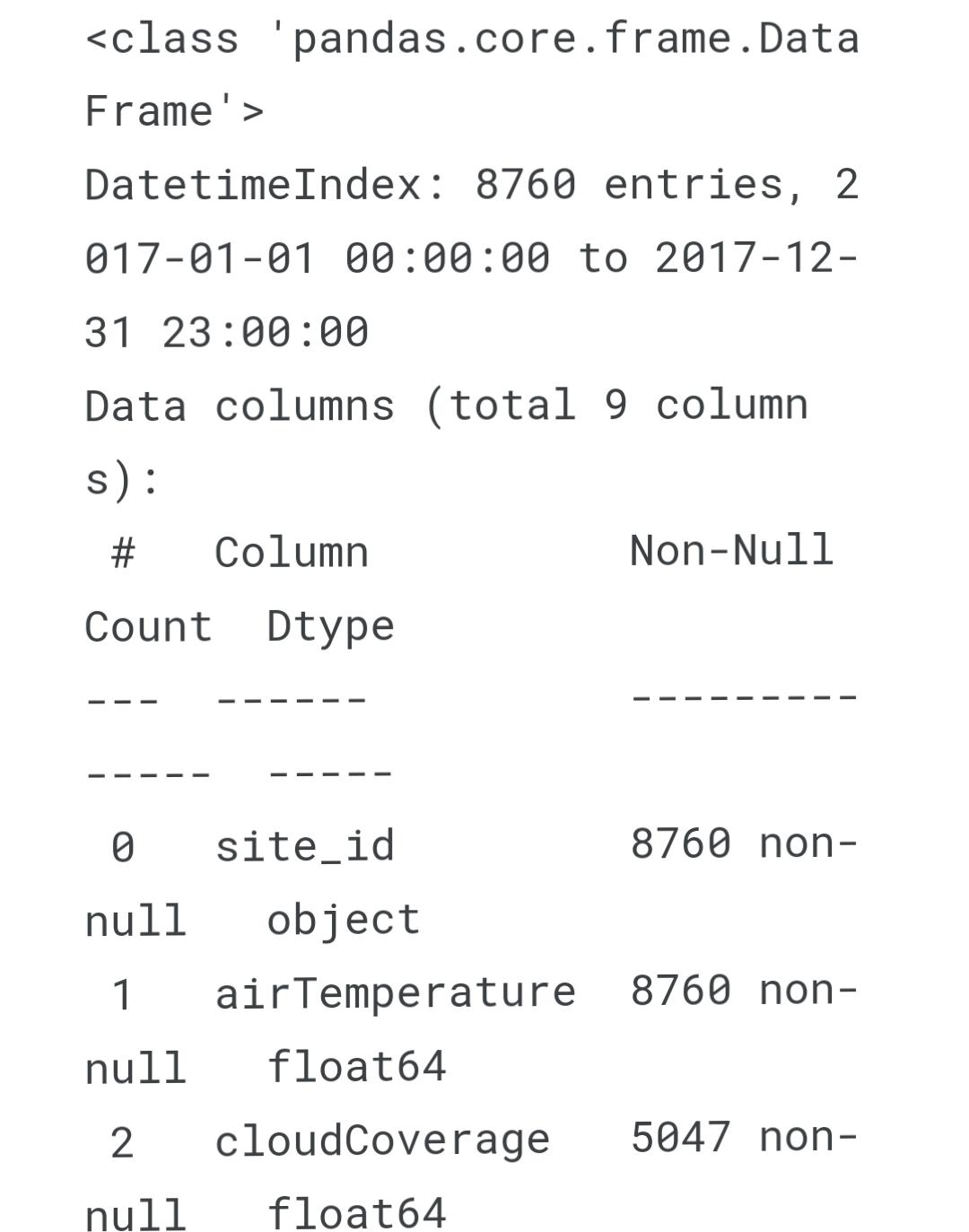


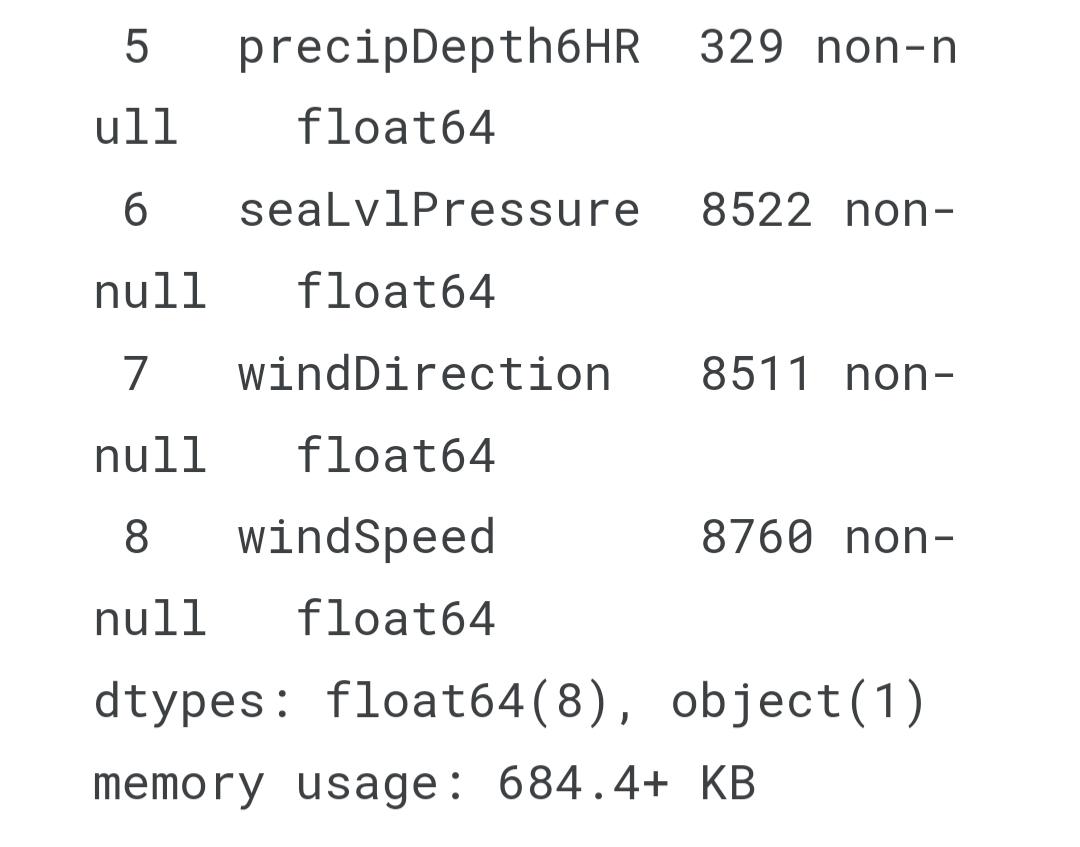


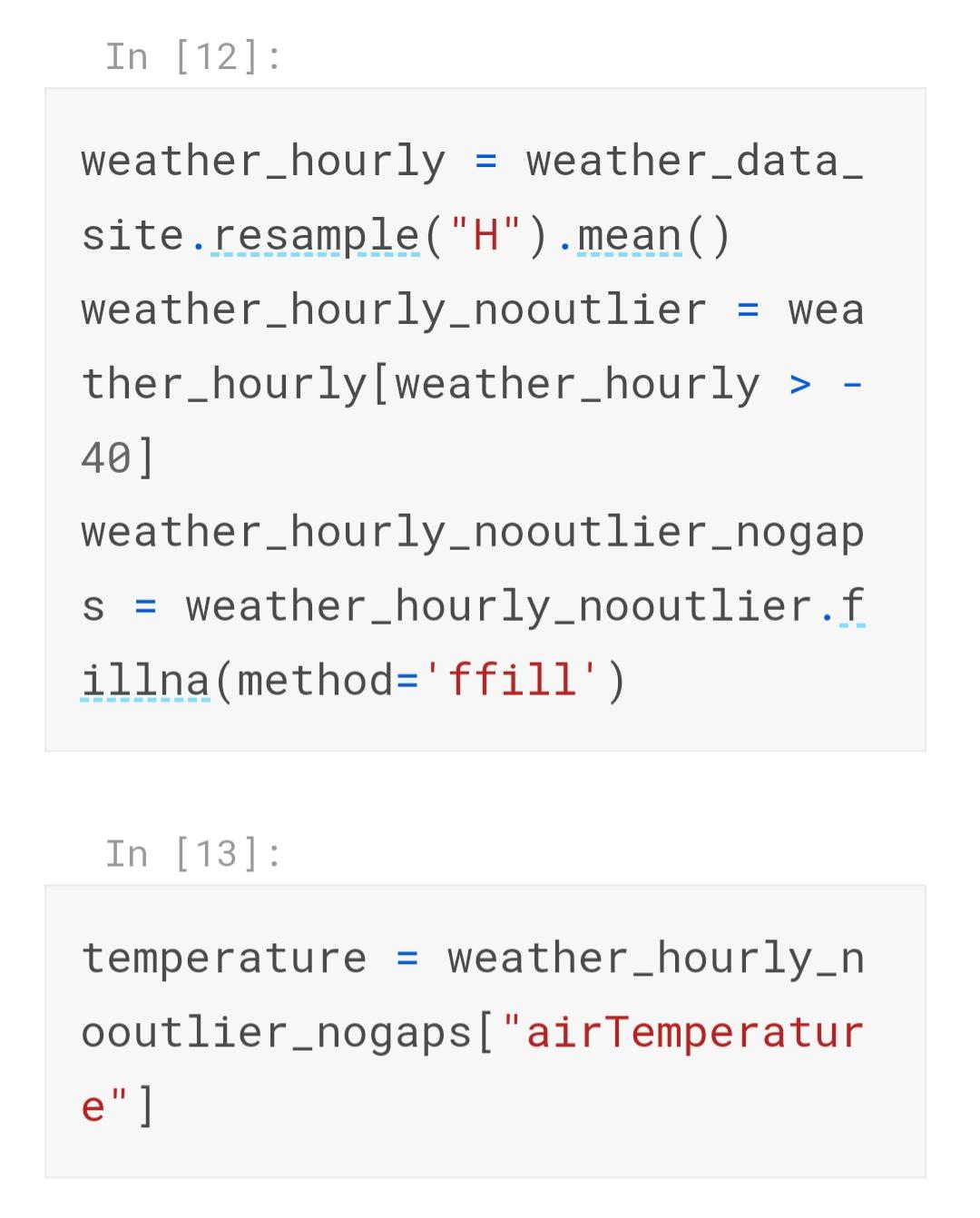


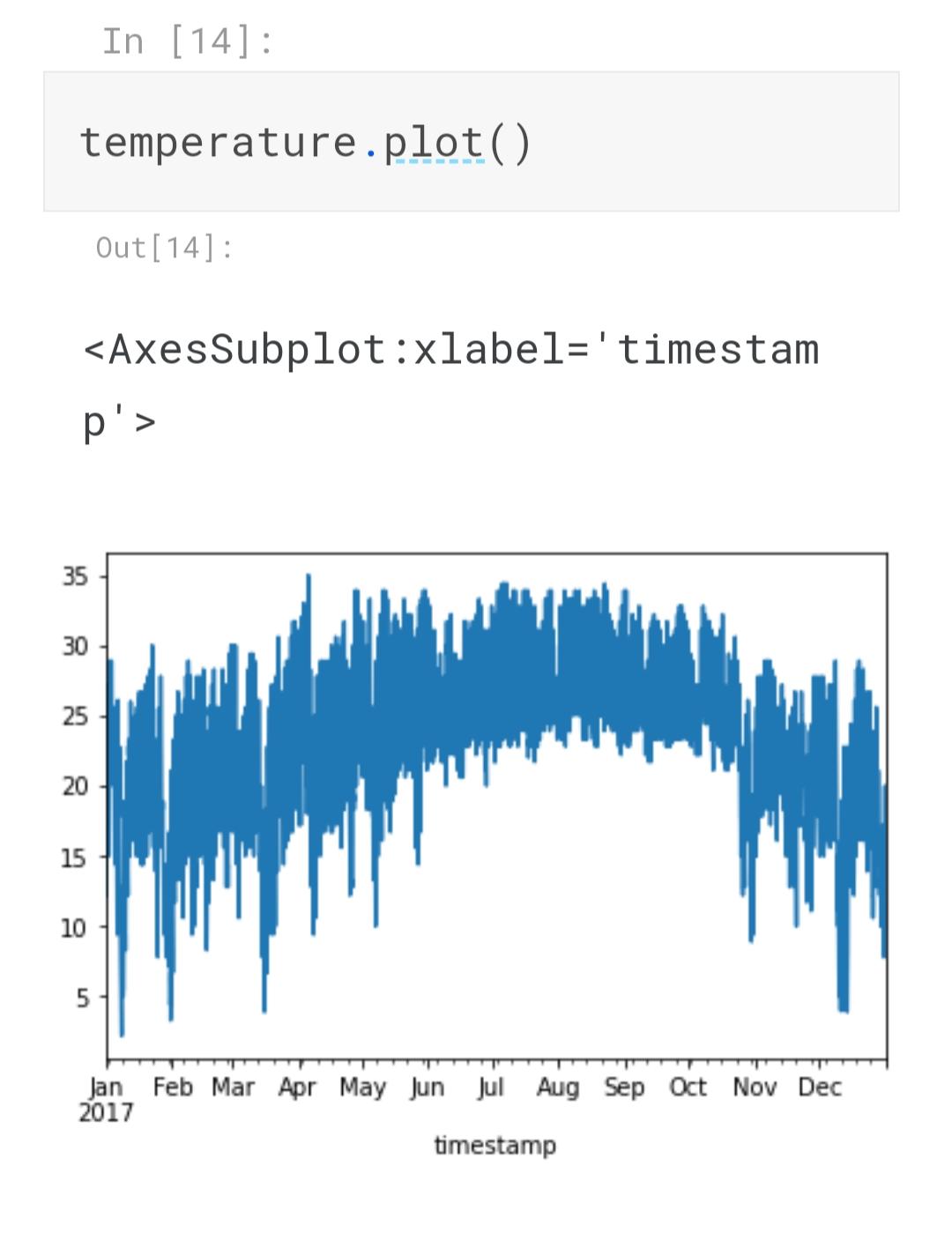


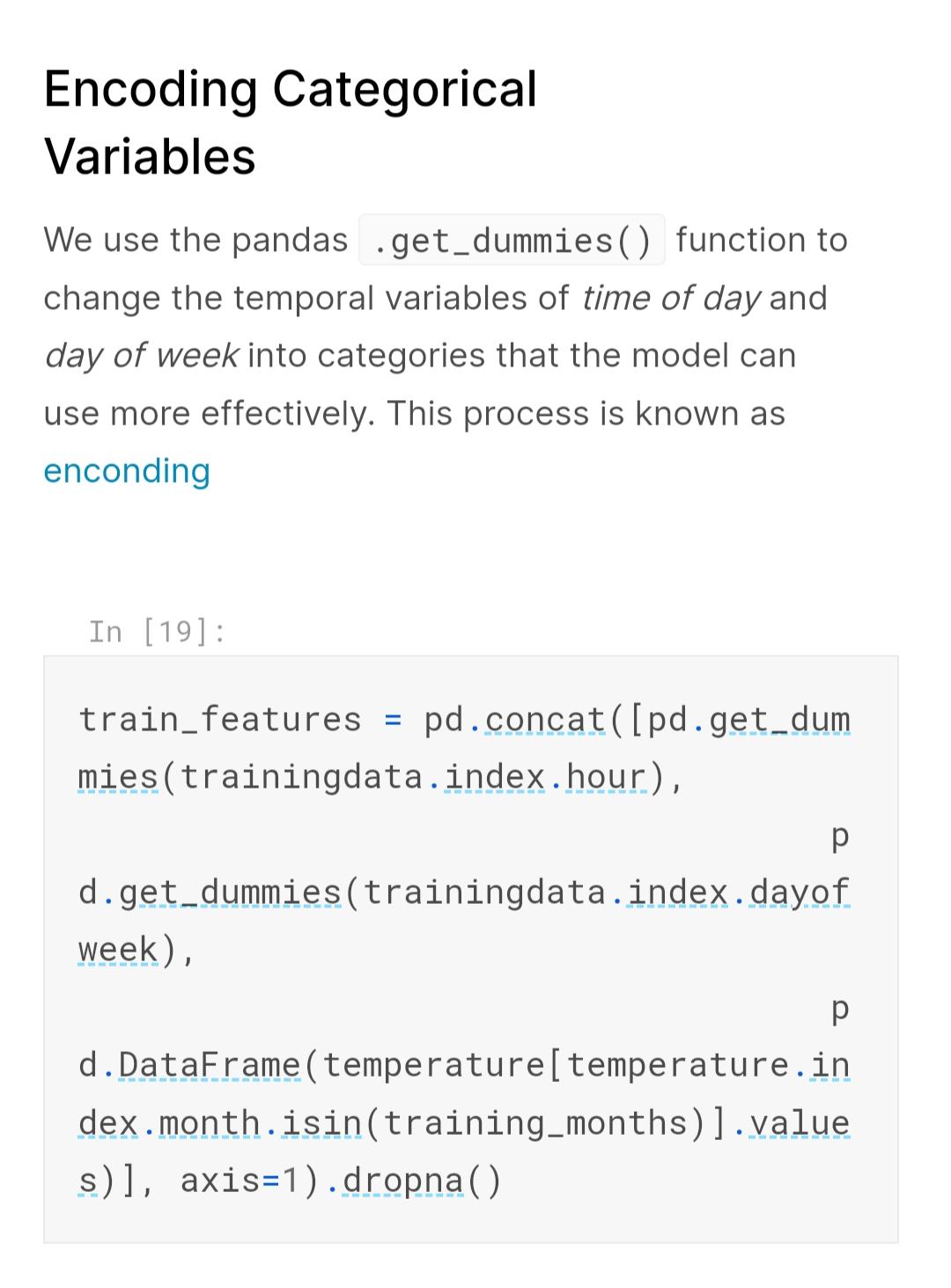


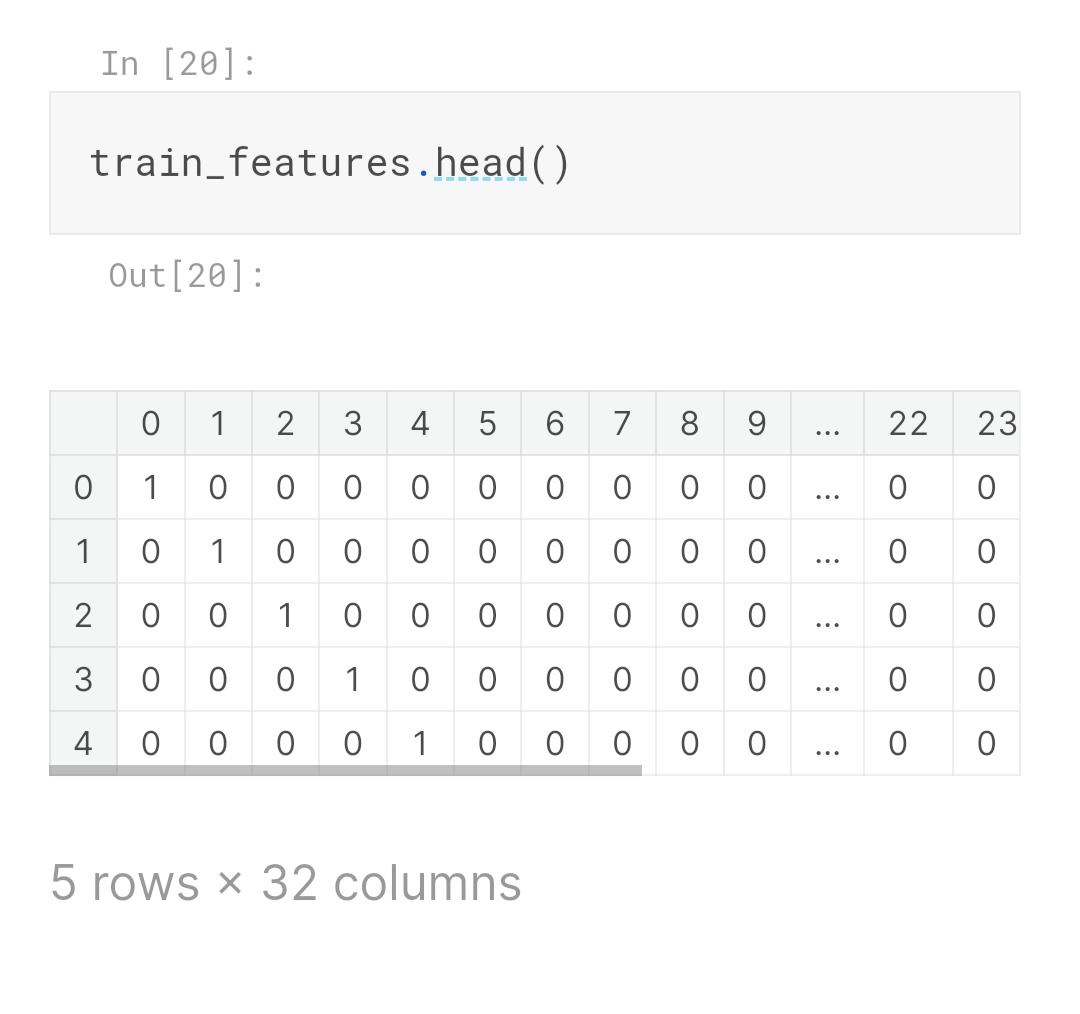


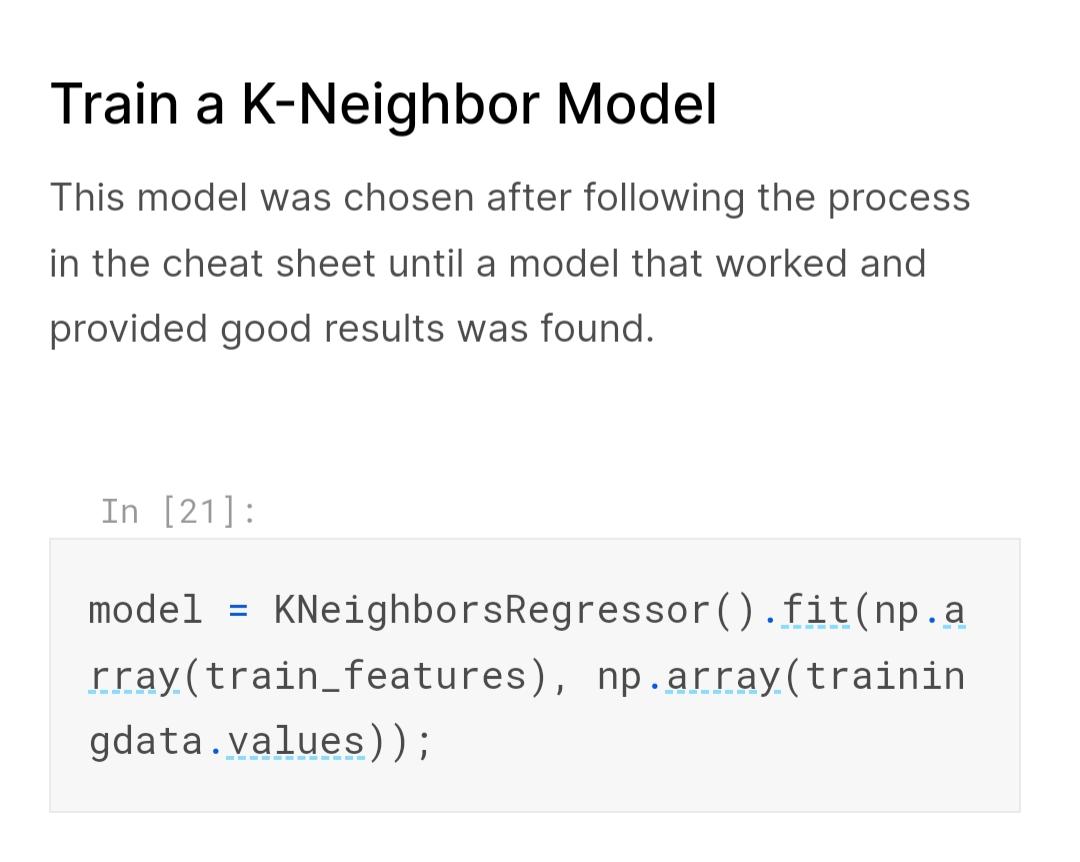


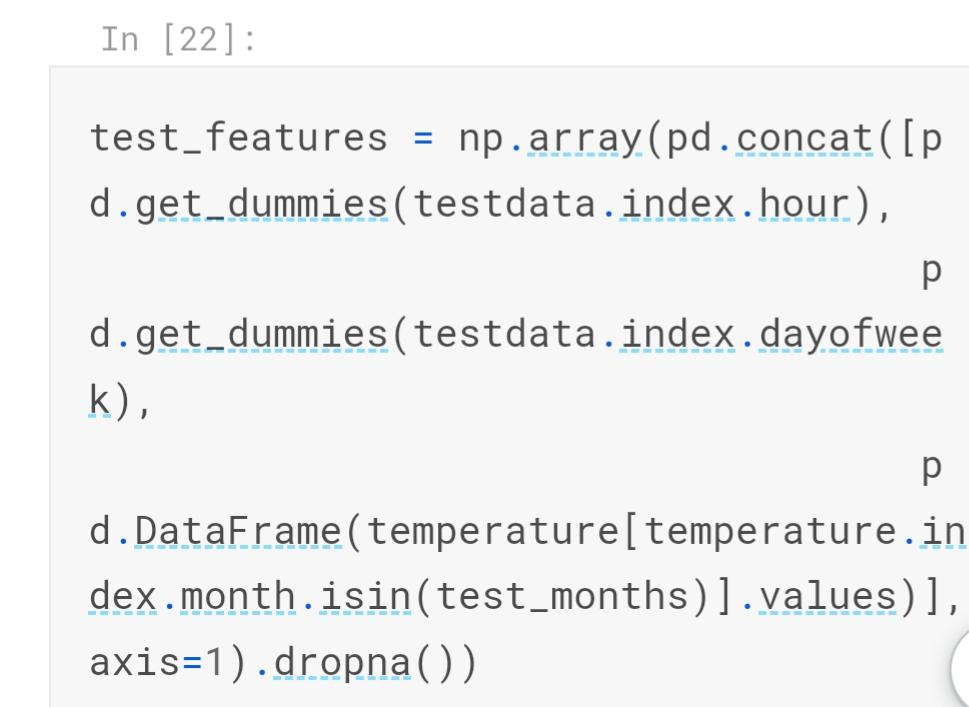


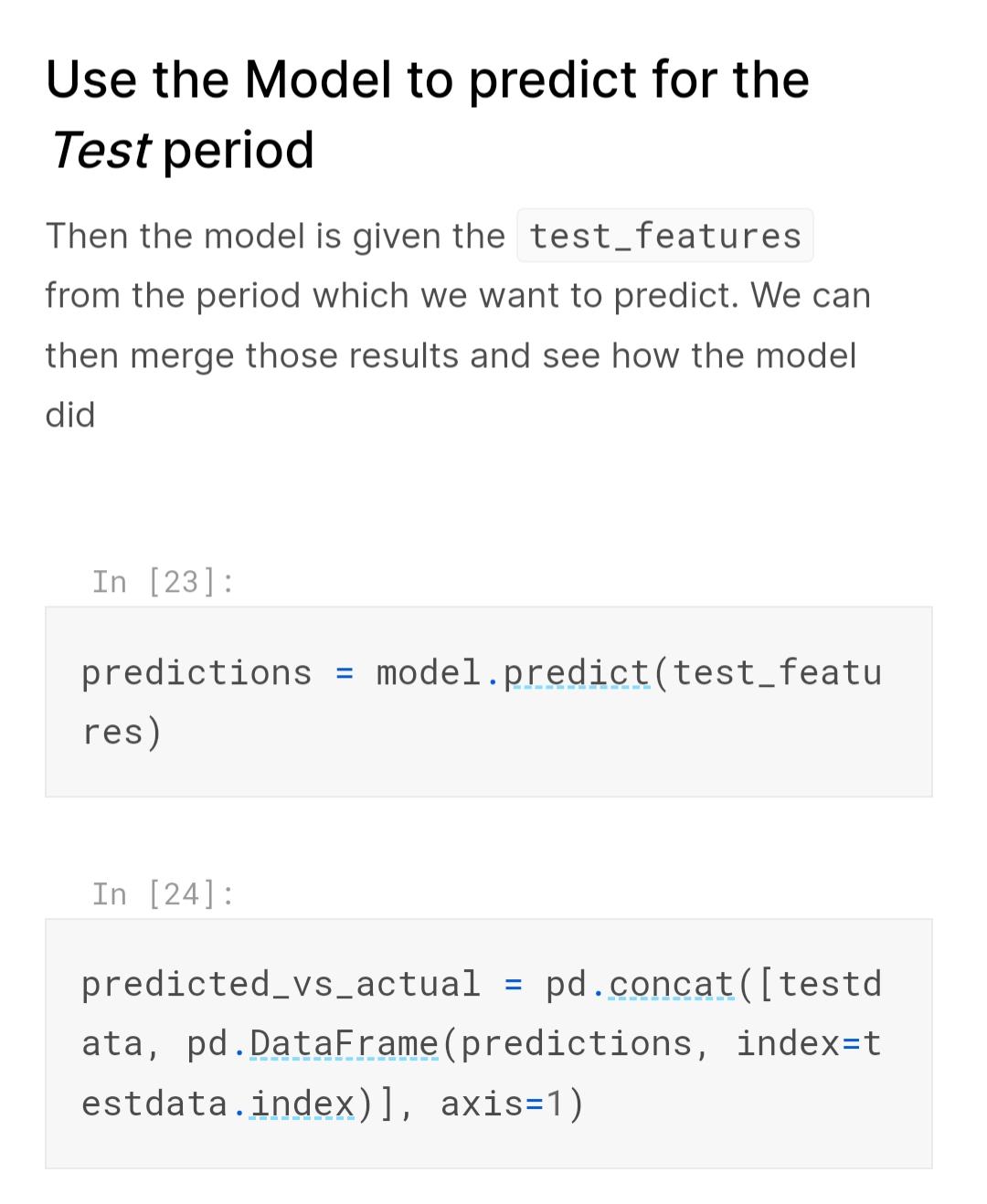


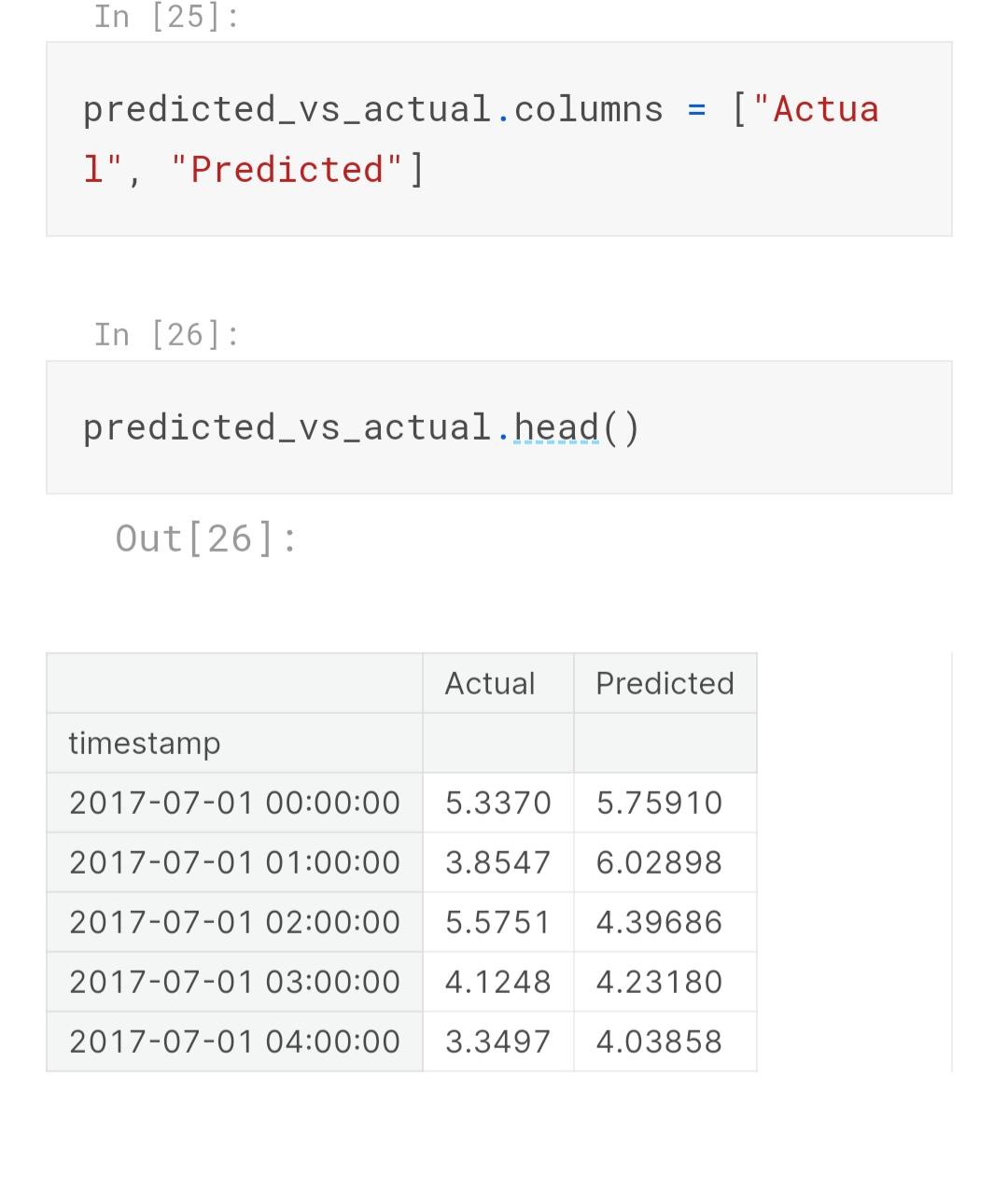


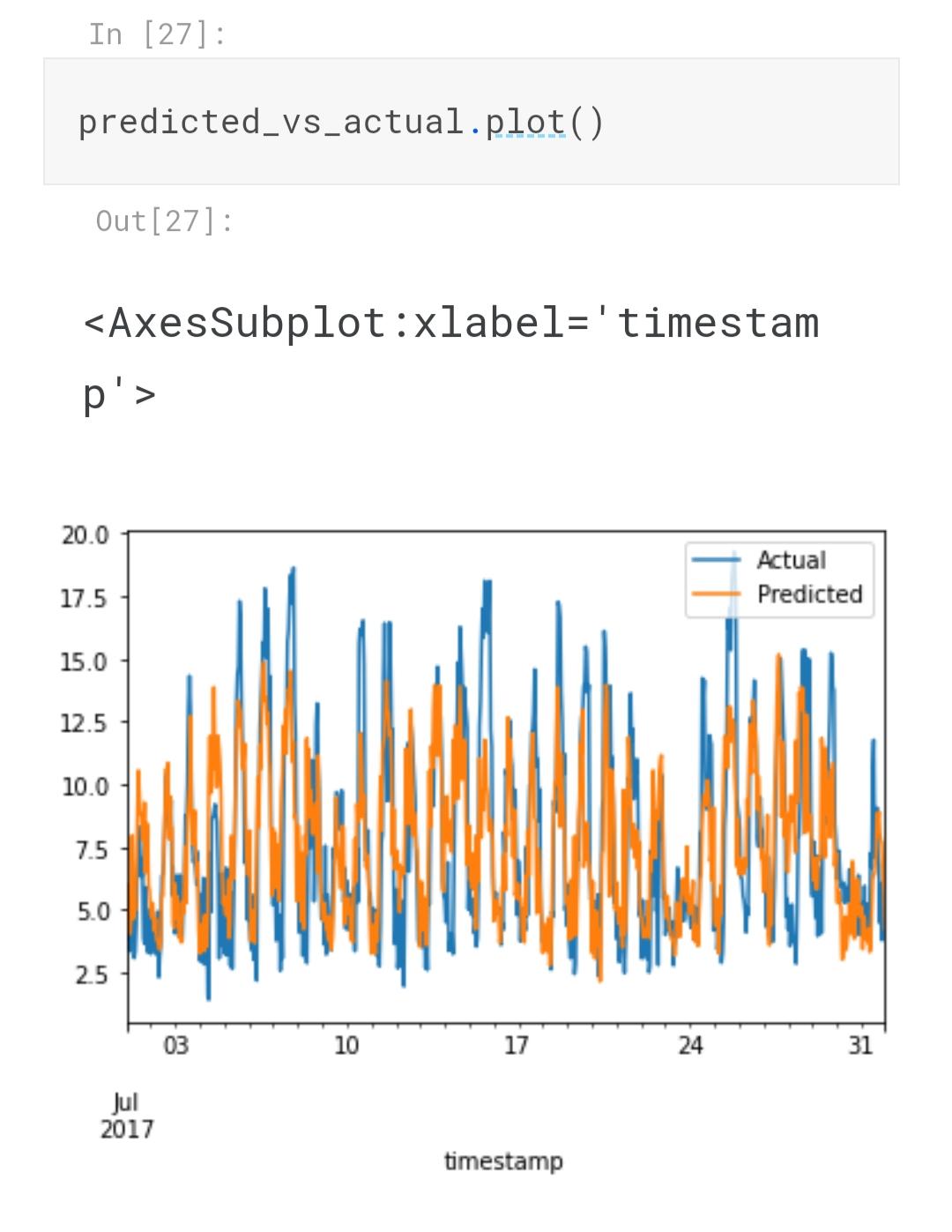




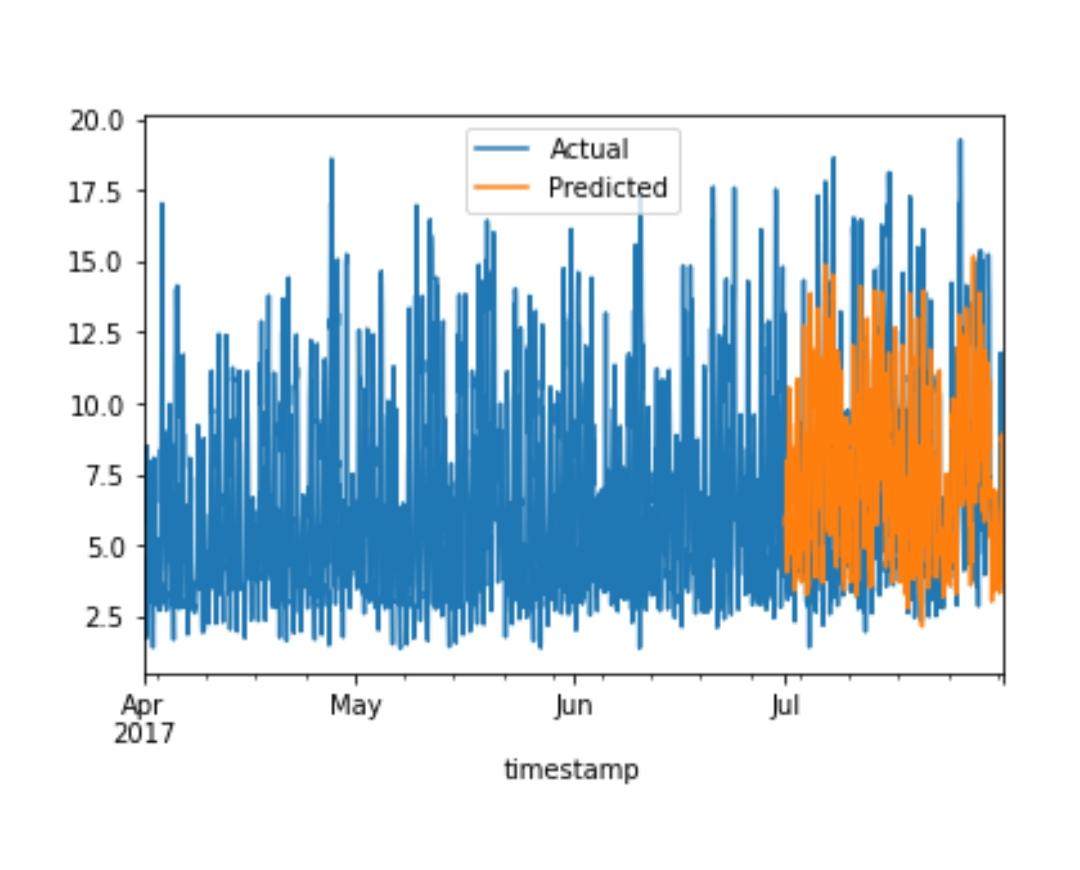












Measure energy consumption

project submission phase 5

**Thanks you**