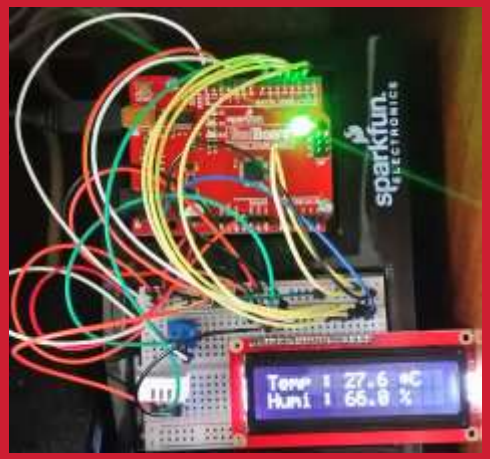
Arduino-IOT

[wk14]



Arduino + Node Data Mining

Visualization of Signals using Arduino, Node.js & Storing Signals in MongoDB & Mining Data using Python

Comsi, INJE University

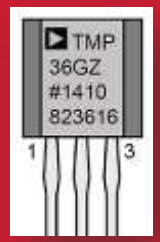
2nd semester, 2019

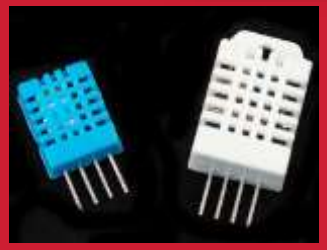
Email : chaos21c@gmail.com

My ID

ID	성명
AA01	김관용
AA02	백동진
AA03	김도훈
AA04	김희찬
AA05	류재현
AA06	문민규
AA07	박진석
AA08	이승협
AA09	표혜성
AA10	김다영
AA11	성소진
AA12	김해인
AA13	신송주
AA14	윤지훈

[Review]





- [wk13]
  - RT Data management with MongoDB
  - Multi-sensor circuits(cds-dht22)
  - Complete your project
  - Upload folder: AAnn_Rpt11

wk13 : Practice : AAnn_Rpt11

- [Target of this week]
  - Complete your works
  - Save your outcomes and upload outputs in github

제출폴더명 : **AAnn_Rpt11**

- 압축할 파일들

① AAnn_iot_json.png

② AAnn_iot_client.png

③ AAnn_s1000.csv (mongoexport file)

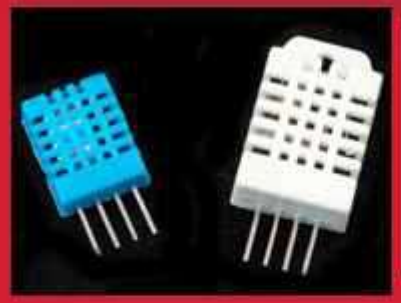
④ AAnn_s1000.png

⑤ client_IoT.html

⑥ All *.ino

⑦ All *.js

⑧ All *.html



[Goal]

Arduino + Node.js

+ plotly.js

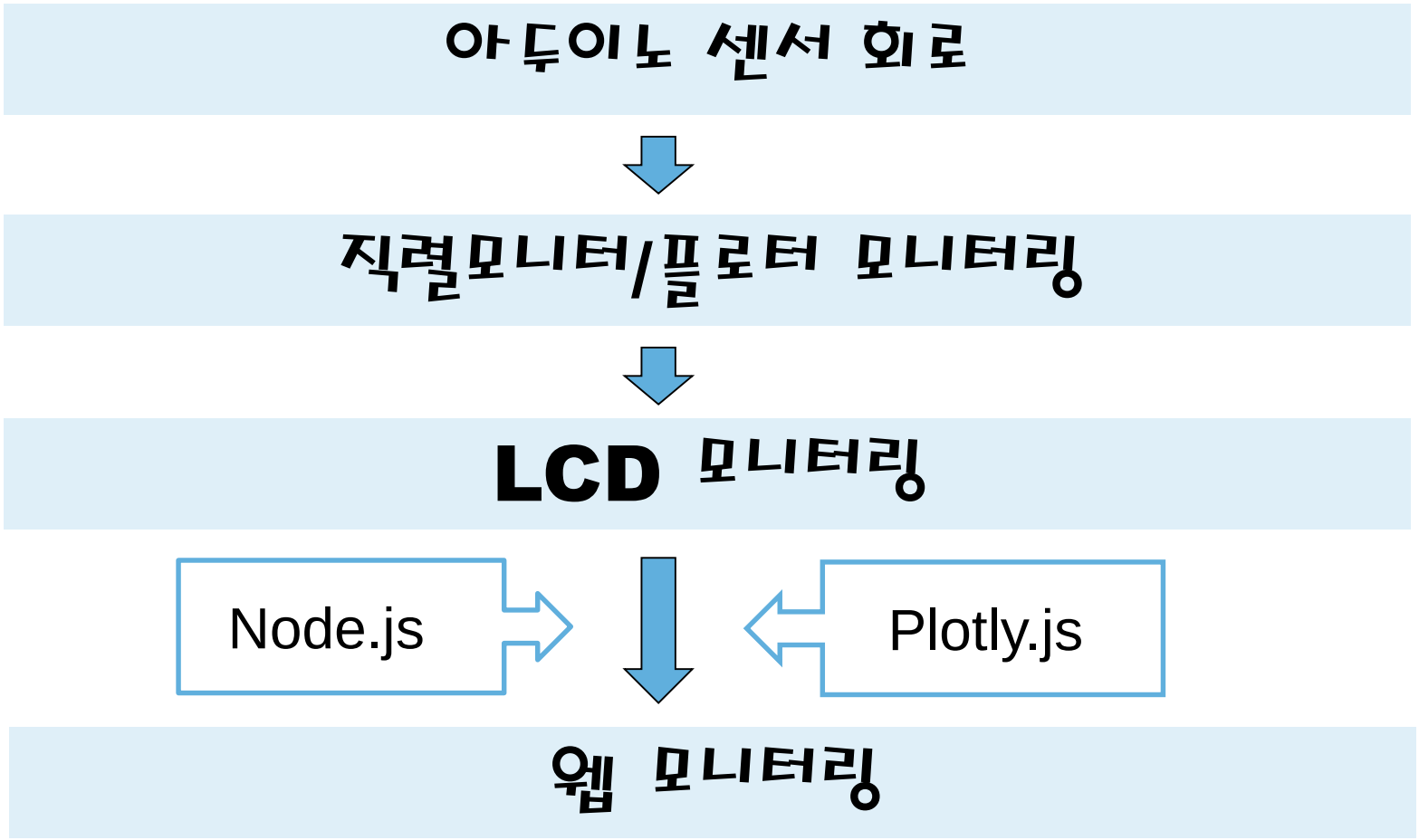
+ MongoDB

→ Data storaging

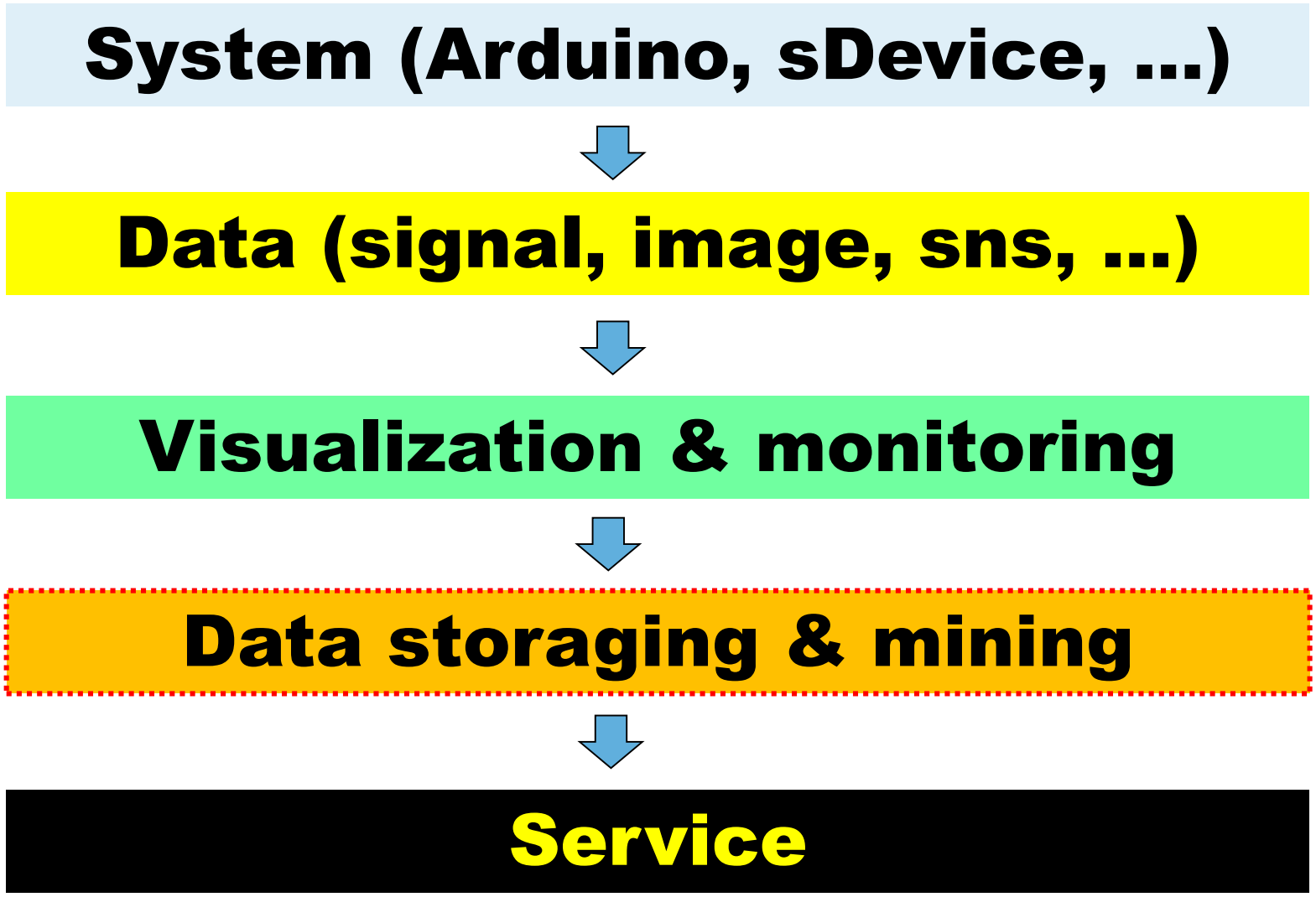
& visualization

& mining

A5.1 Introduction to data visualization

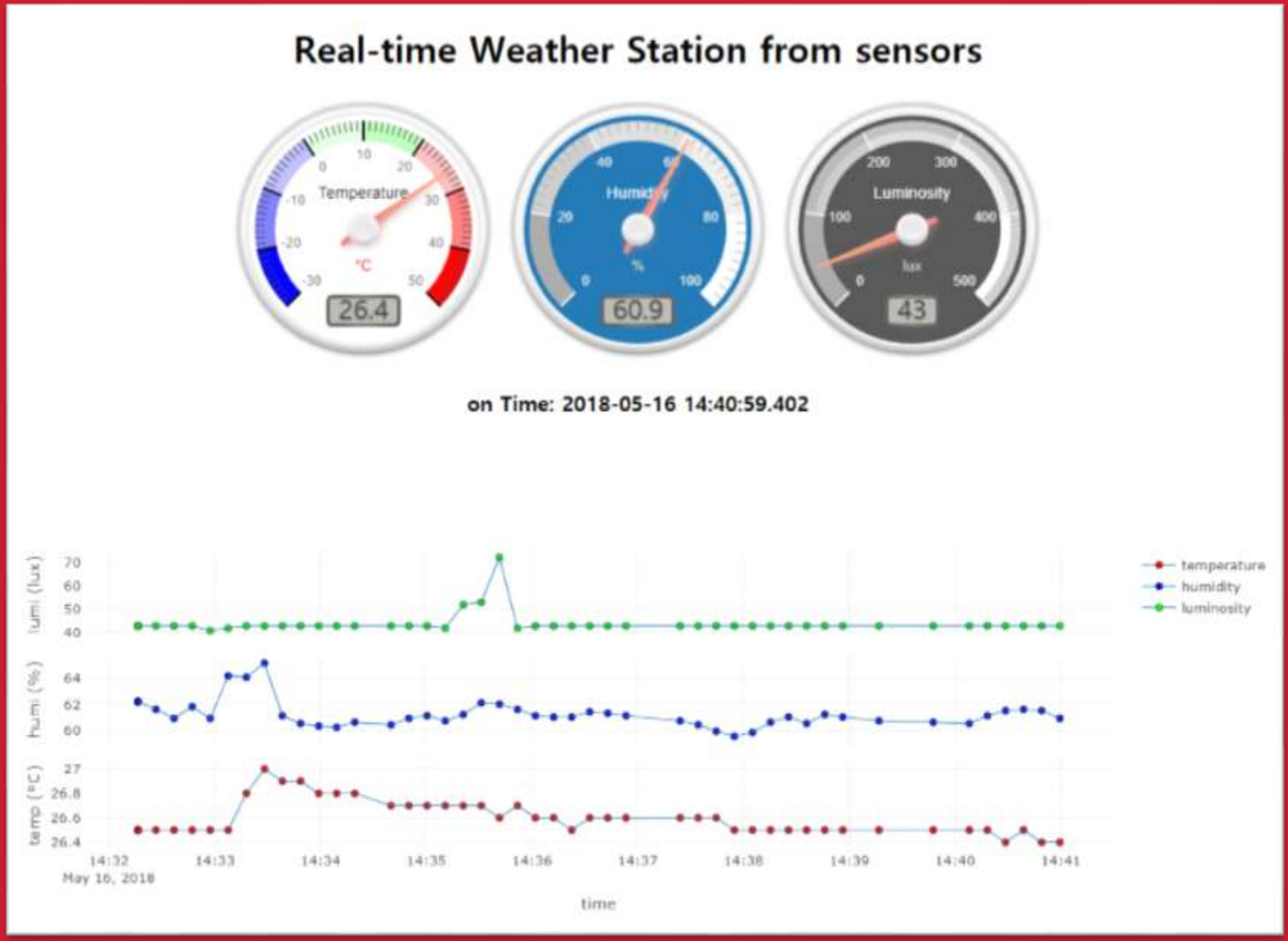


A5. Introduction to IoT service

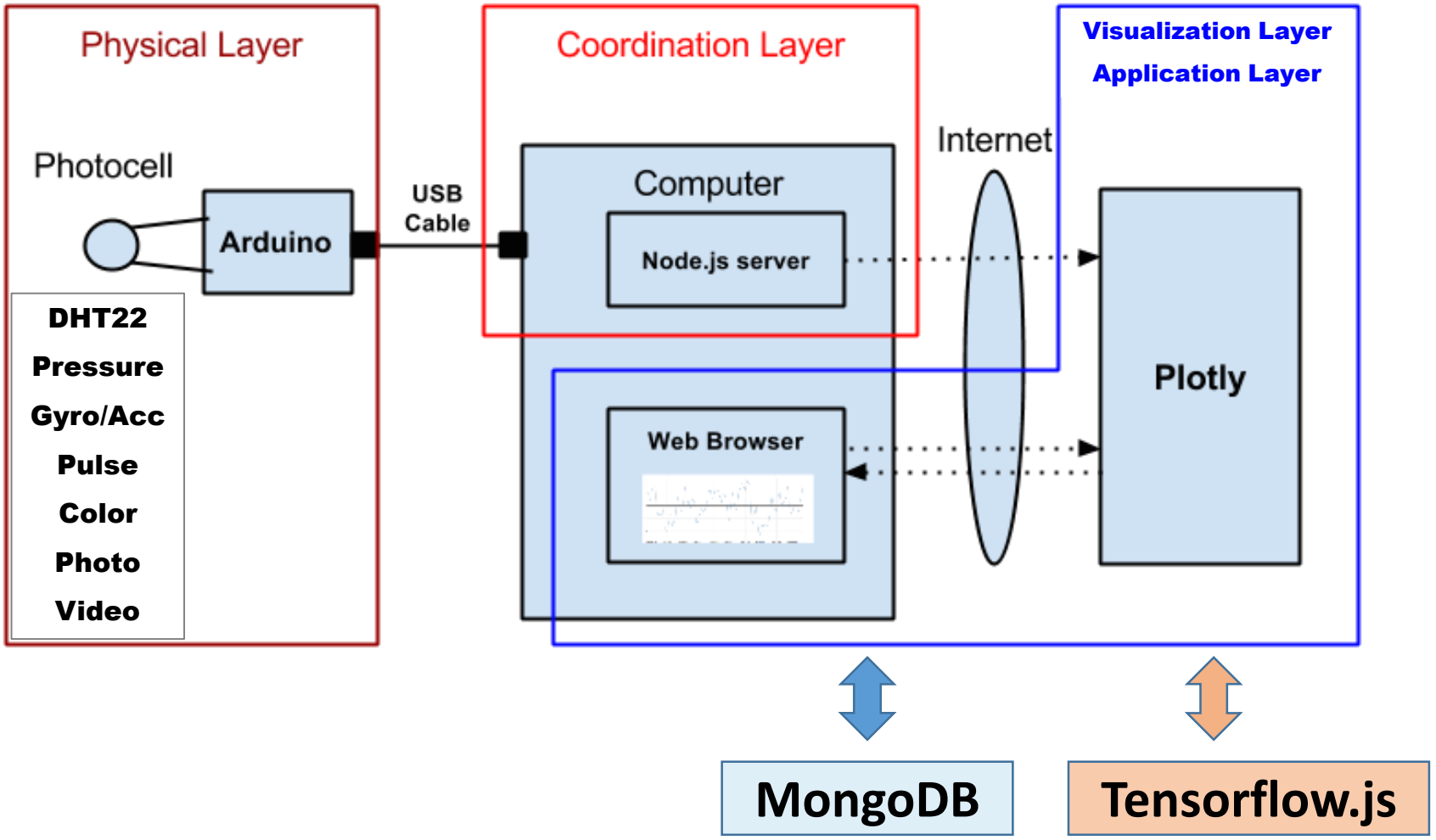


Arduino data on network socket

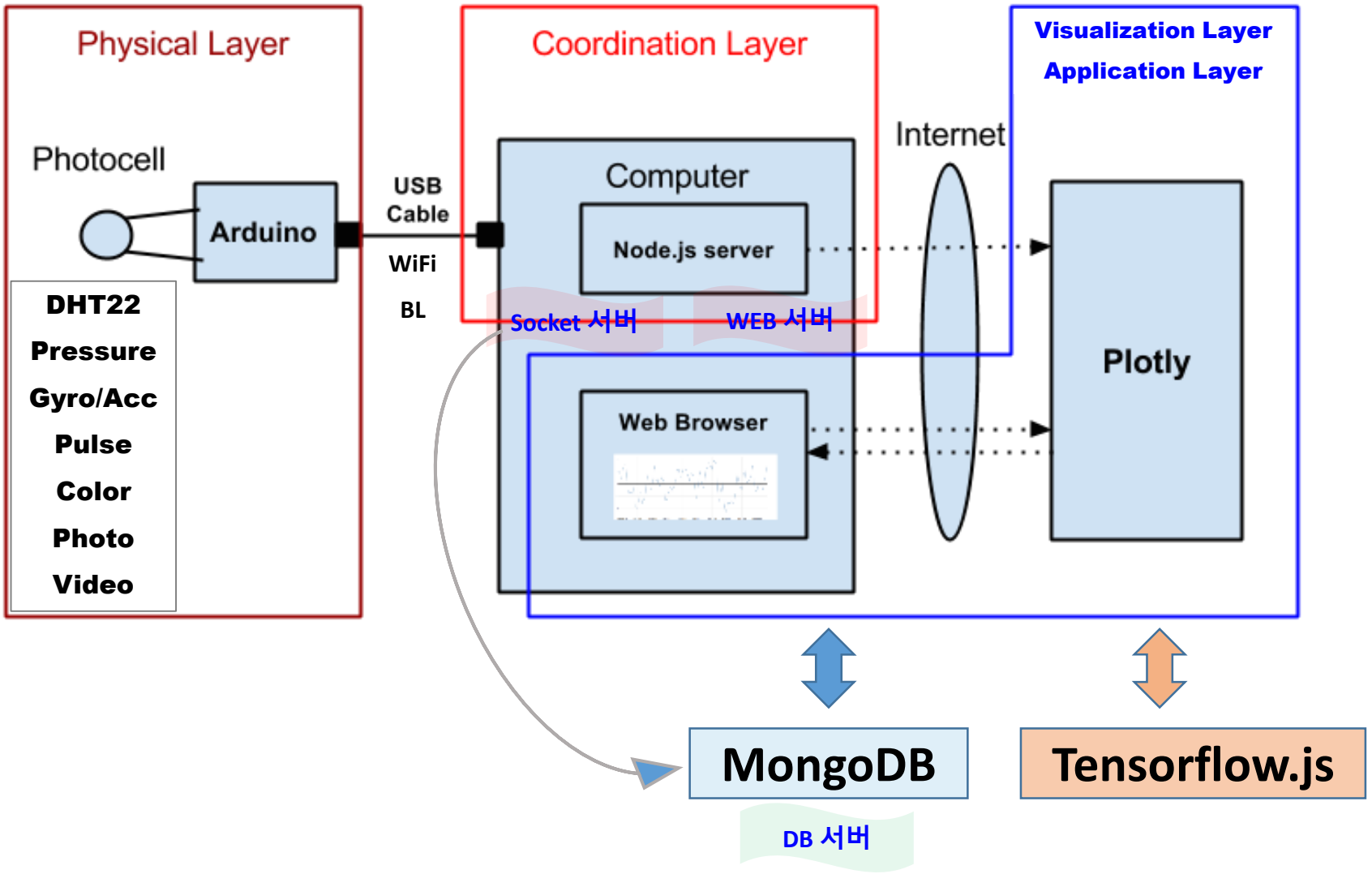
Arduino data + plotly



Layout [H S C]


Physical Layer
Photocell
Arduino
DHT22
Pressure
Gyro/Acc
Pulse
Color
Photo
Video
USB
Cable
Coordination Layer
Computer
Node.js server
Web Browser
Internet
Visualization Layer
Application Layer
Plotly
MongoDB
Tensorflow.js

Layout [H S C-IoT]



3-servers

mongoDB

socket.io

3000

Cloud (DB)

Network-Socket

node

express

3030

Services (Client)

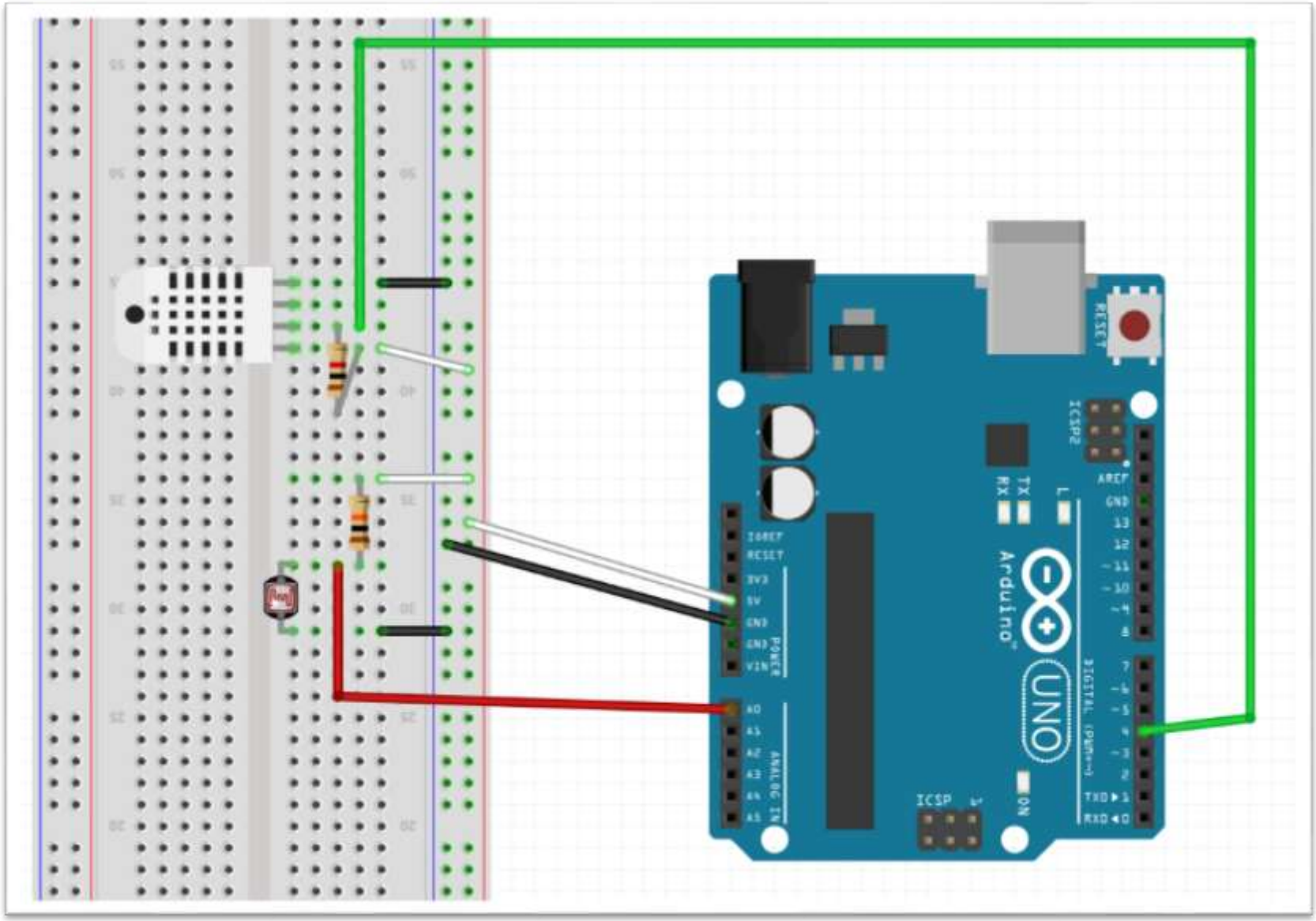
Arduino

& Node.js

& MongoDB

& Express server

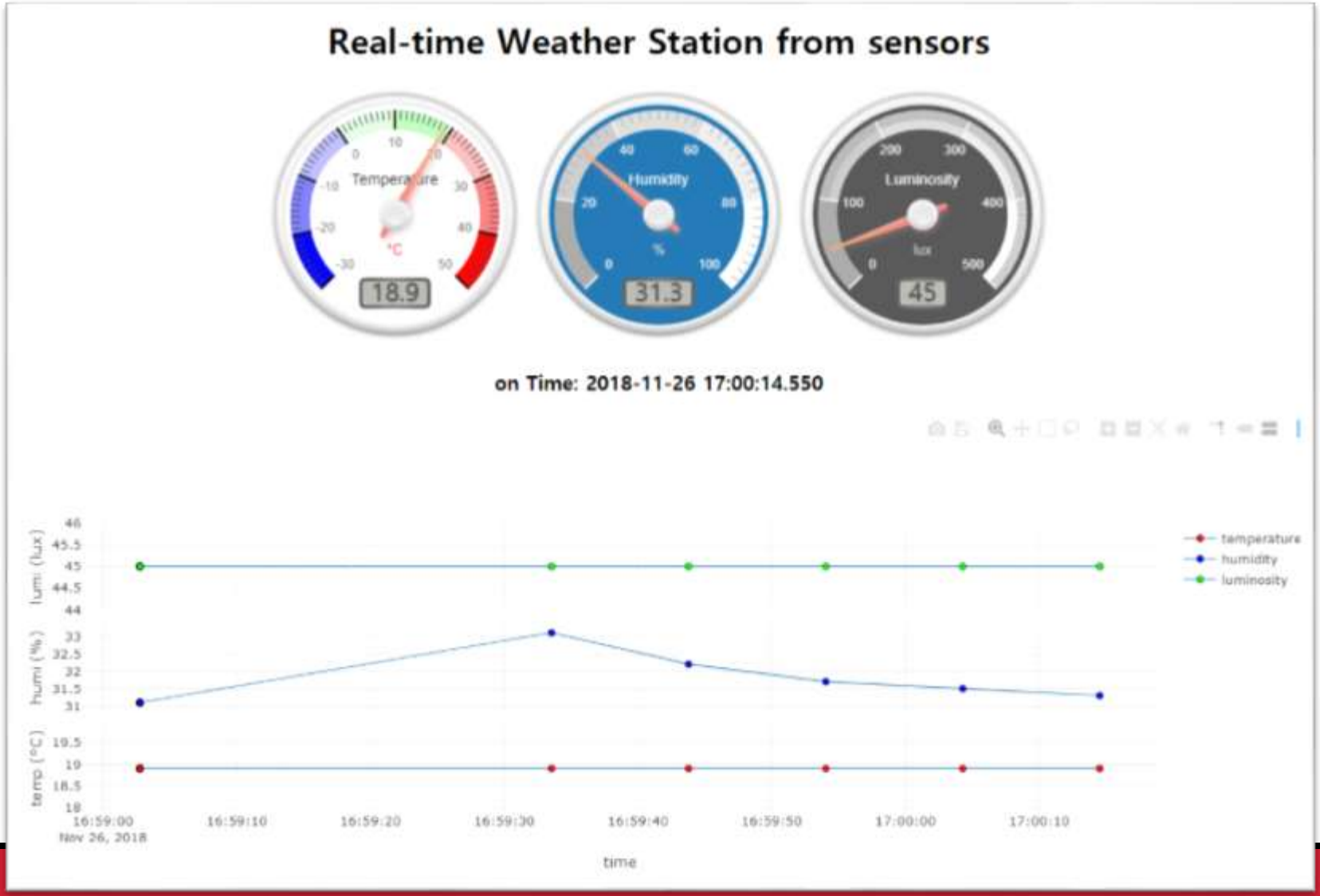
DHT22 + CdS : circuit



A5.9.6 DHT22 + CdS + Node.js + MongoDB

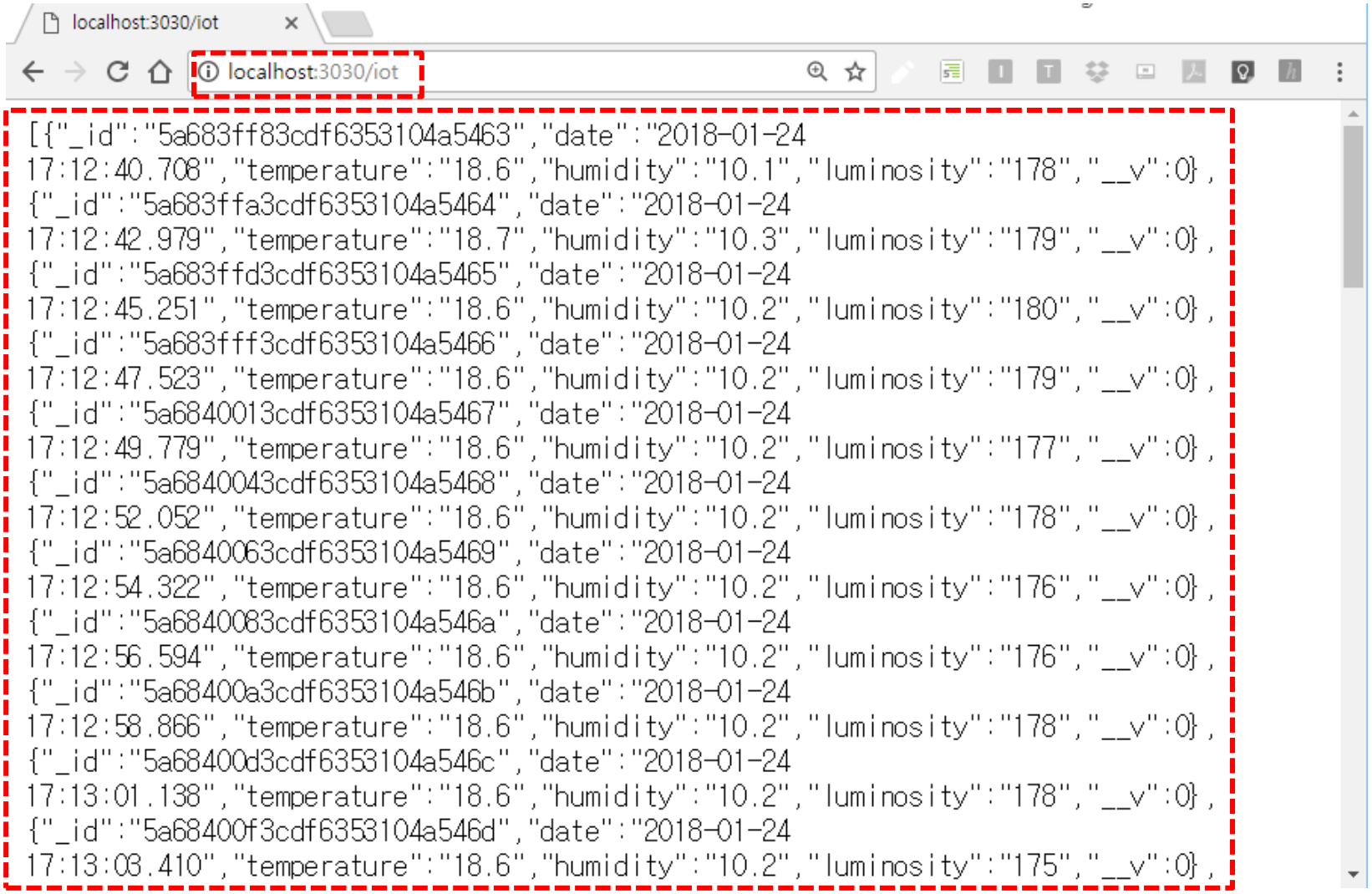
2.7 copy cds_dht22_client.html & gauge.min.js → ./public/ subfolder

http://localhost:3030/client_cds_dht22.html (web root folder)



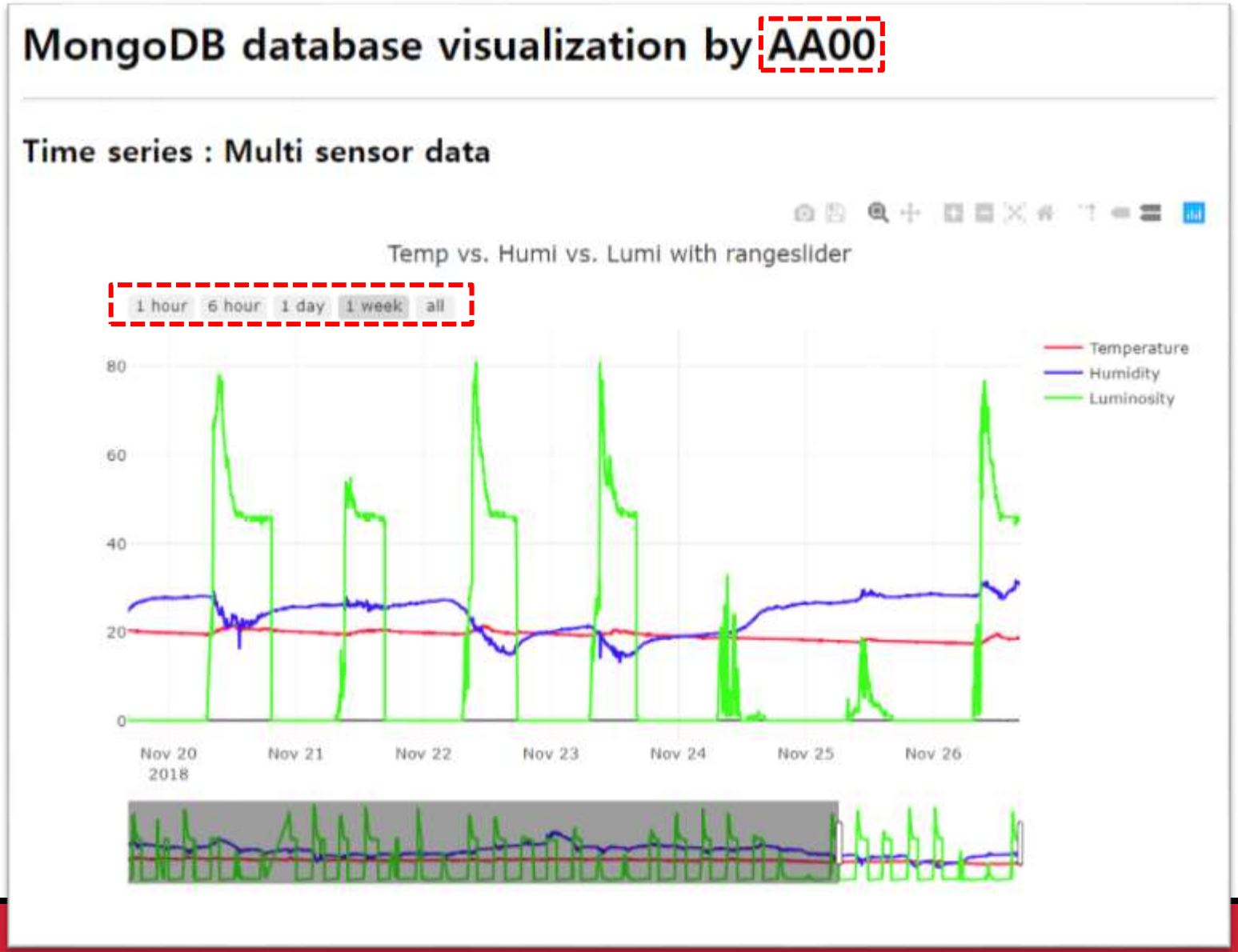
A5.9.6 DHT22 + CdS + Node.js + MongoDB

2.5 cds_dht22_express.js → routing2 http://localhost:3030/iot



A5.9.7 DHT22 + CdS + Node.js + MongoDB

3.5 Web client: client_iotDB.html – iot DB monitoring (public 폴더에서 제공)



MongoDB data management

- Query in mongo shell
- Export & import MongoDB
- Using and understanding iot data with Python (or R)

A5.9.8 MongoDB management

1. Query in Mongo shell

db.sensors.count() → sensors collection에 있는 도큐먼트 (문서)의 수

db.sensors.find().sort({_id: 1}).limit(10) → 오래된 document 10개 추출

db.sensors.find().sort({_id: -1}).limit(10) → 최근 document 10개 추출

db.sensors.find({date: {$gt: "2019-11-26 22:26:05"}}) → 특정 시간 이후 document 추출

db.sensors.find({temperature: {$gt: 29}}) → 온도가 29도를 넘는 document 추출

https://docs.mongodb.com/manual/tutorial/query-documents/

A5.9.8 MongoDB management

2. Import or export MongoDB (windows cmd 창에서 실행)

- **mongoimport -d dbName -c collectionName --type csv --headerline --file fileName.csv**

- **mongoexport -d dbName -c collectionName --fields <field1,field2,…> --limit=nn --type csv --out fileName.csv**

json 또는 csv 파일로 import/export

https://docs.mongodb.com/manual/reference/program/mongoimport/

https://docs.mongodb.com/manual/reference/program/mongoexport/

A5.9.8 MongoDB management

[Tip] iot db의 최근 데이터 500개를 csv 파일 (s500.csv)로 저장할 때,

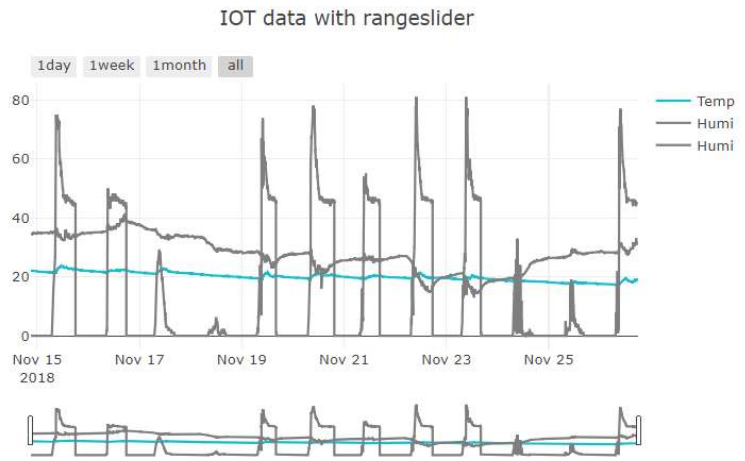
- **mongoexport –d iot -c sensors --sort “{_id: -1}” --limit=500 --fields date,temperature,humidity,luminosity --type=csv --out s500.csv**

```
C:₩Users₩biochaos>mongoexport -d iot11 -c sensors --sort "{_id:-1}" --limit=100000 --type=csv --fields date,temperature,
humidity,luminosity --out iot_chaos.csv
2018-11-26T17:50:23.577+0900    connected to: localhost
2018-11-26T17:50:24.576+0900    [###############.........]  iot11.sensors  64000/100000  (64.0%)
2018-11-26T17:50:24.797+0900    [########################]  iot11.sensors  100000/100000  (100.0%)
2018-11-26T17:50:24.798+0900    exported 100000 records
```

	A	B	C	D
1	date	temperatu	humidity	luminosity
2	50:18.6	18.9	31.6	45
3	50:08.4	18.9	31.6	45
4	49:58.1	18.9	31.6	45
5	49:47.8	19	31.7	45
6	49:37.6	19	31.7	45
7	49:27.3	18.9	31.7	45
8	49:17.1	18.9	31.6	45

Data visualization by AAnn

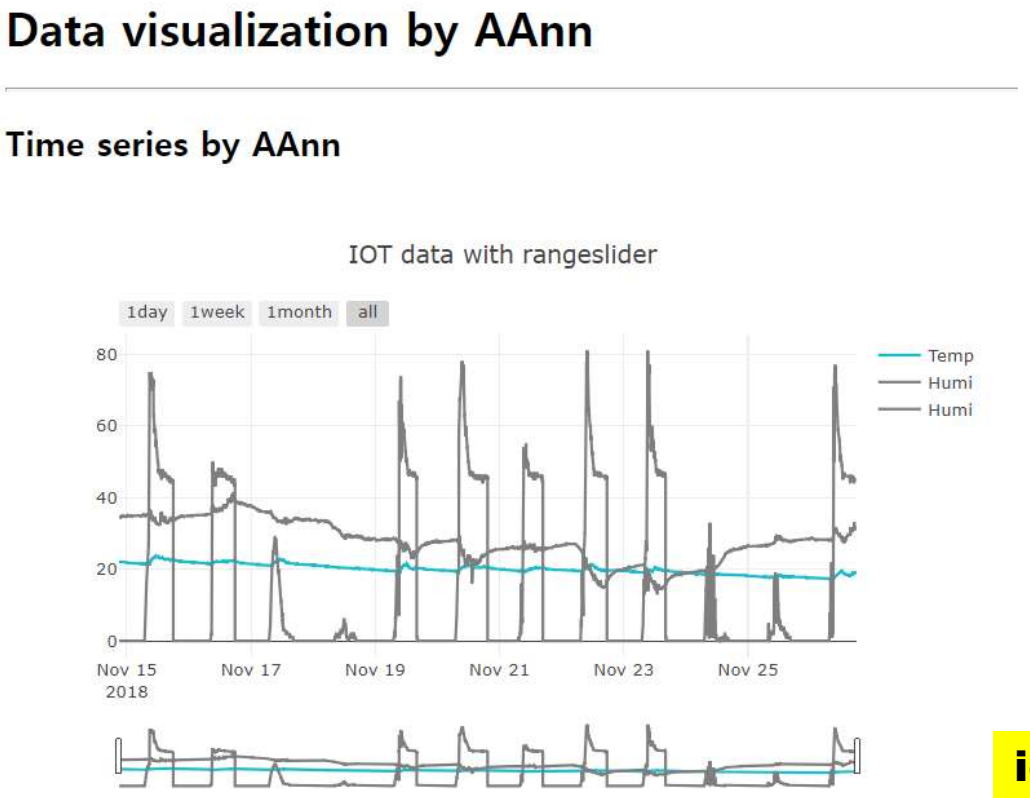
Time series by AAnn



A5.9.8 MongoDB management

[DIY]

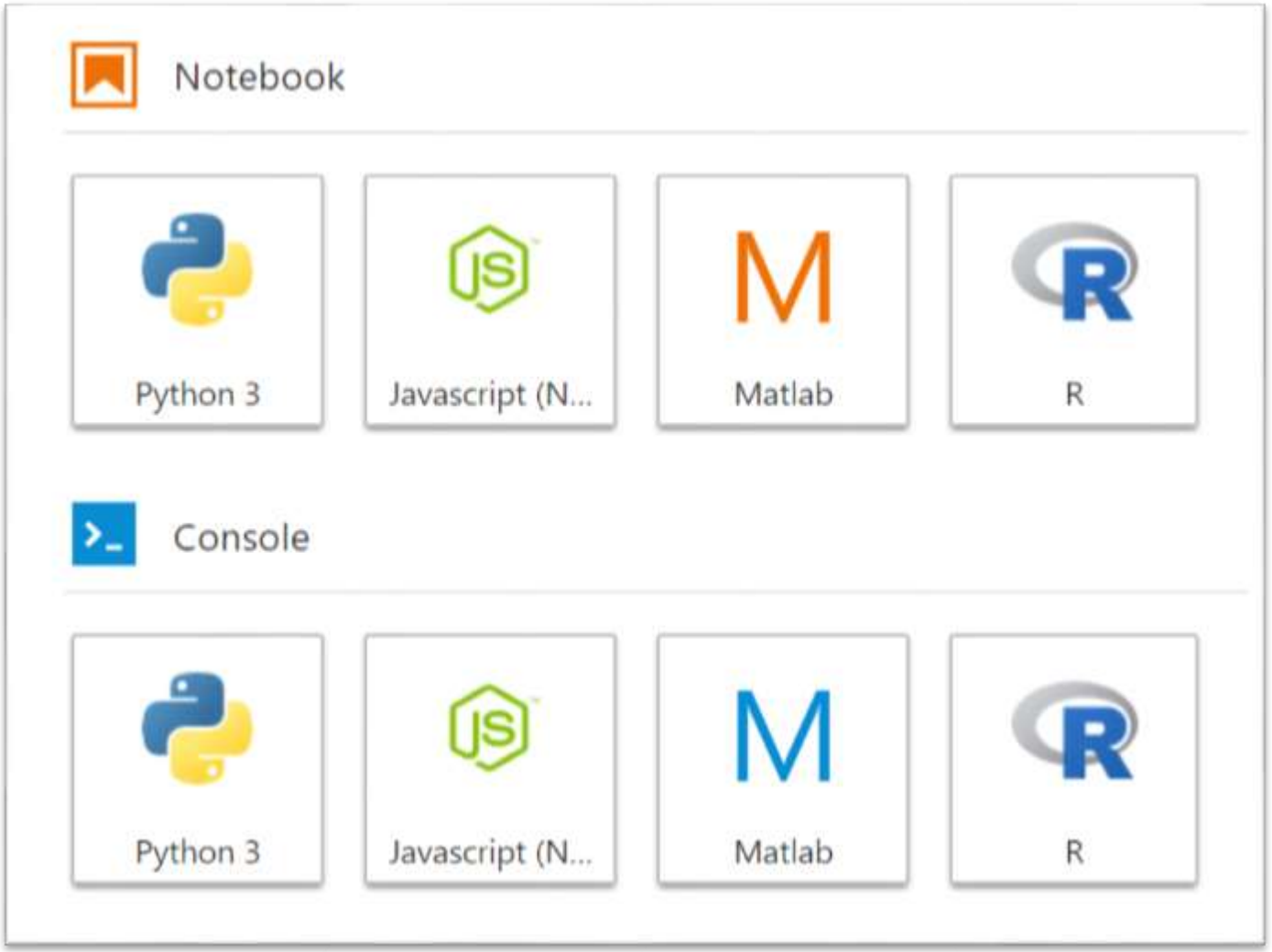
1. **iot db의 최근 데이터 1000개를 csv 파일 (AAnn_s1000.csv)로 저장하시오.**
2. **저장된 AAnn_s1000.csv 파일을 public/data 폴더에 복사.**
3. **csv 파일을 이용하는 Rangeslider가 포함된 웹 클라이언트 client_iot.html 파일을 만드시오.**



iot_chaos.html

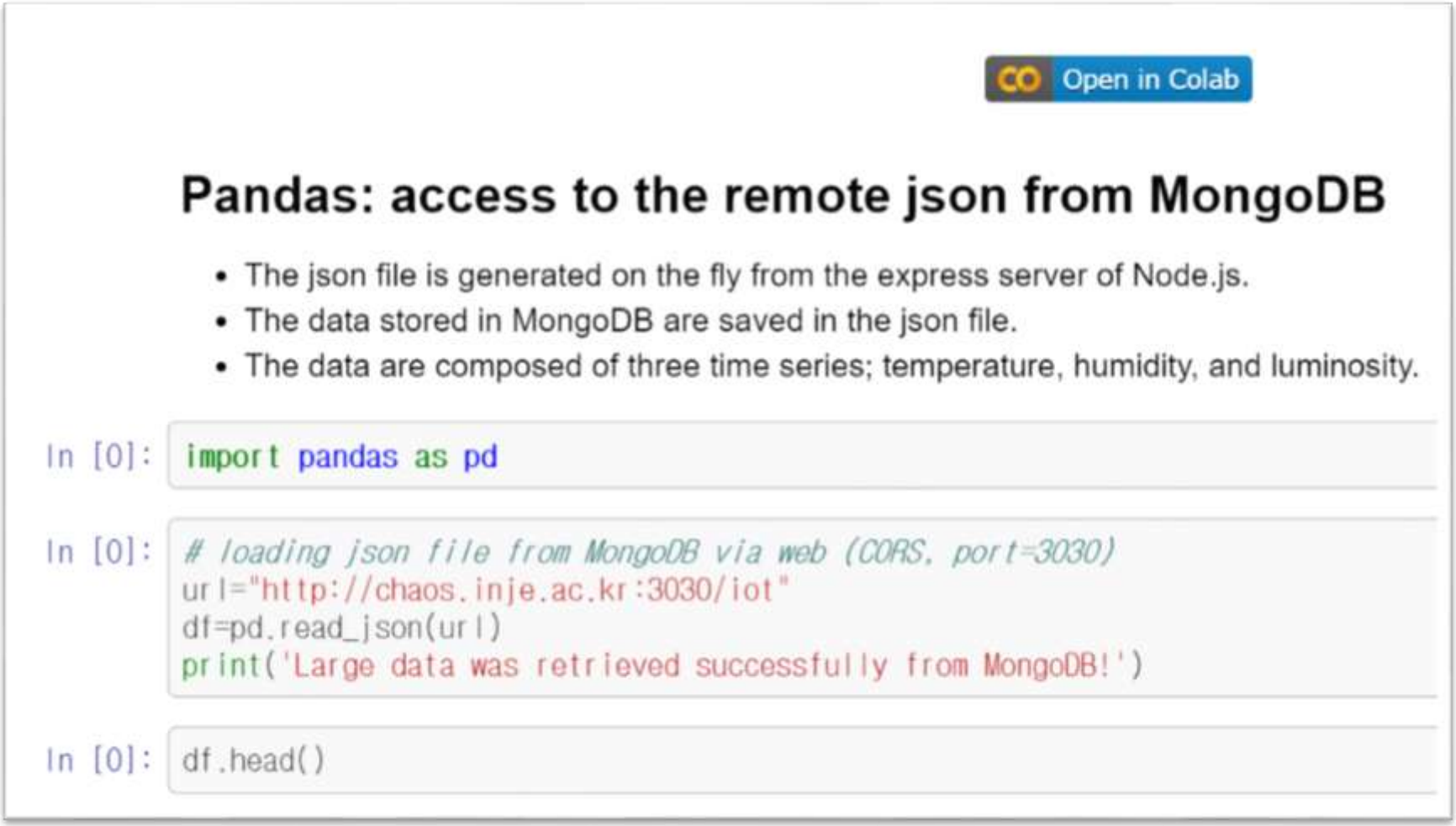
IoT data mining

3. How to use and understand iot data? → Python(or R) in Colab/Jupyter lab



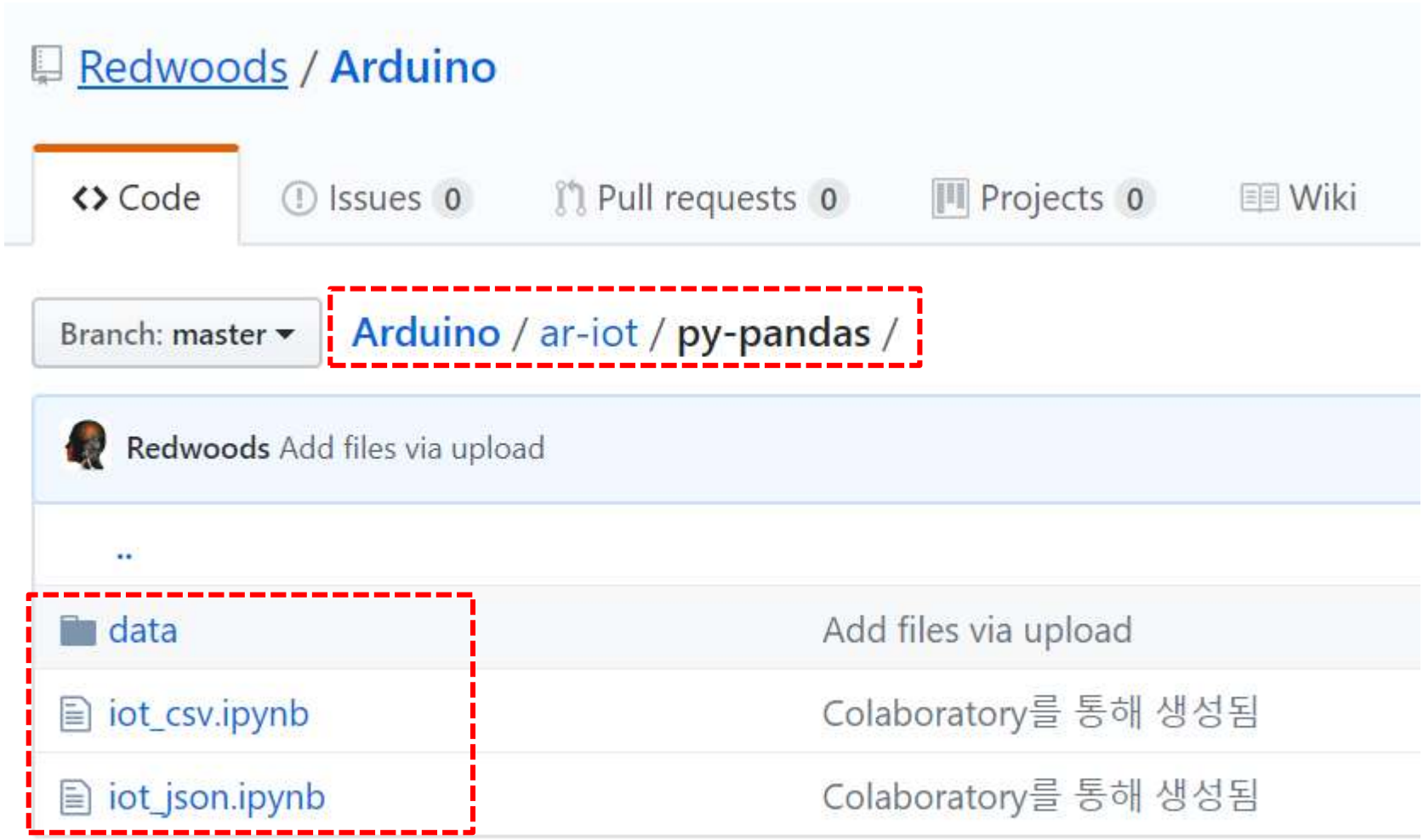
IoT data mining

How to use and understand iot data? → Google Colab



A5.9.8 IOT data mining

3.1 How to use and understand iot data? → iot_csv.ipynb, iot_json.ipynb



A5.9.8 MongoDB management

3.2 Loading data … → iot_json.ipynb

[1]
```
1 import pandas as pd
```

[2]
```
1 # loading json file from MongoDB via web (CORS, port=3030)
2 url="http://chaos.inje.ac.kr:3030/iot"
3 j1=pd.read_json(url)
```

[3]
```
1 j1.head()
```

1. Express 서버에서 MongoDB에 접속한다.
2. 아두이노에서 만들어져 전송되어 MongoDB에 저장되고 있는 센서 데이터를 json 파일로 가져온다.

	__v	_id	date	humidity	luminosity	temperature
0	0	5bce24218d1ec32774d781a9	2018-10-23 04:25:21.349	39.7	0	23.2
1	0	5bce242b8d1ec32774d781aa	2018-10-23 04:25:31.594	39.7	0	23.2
2	0	5bce24358d1ec32774d781ab	2018-10-23 04:25:41.855	39.7	0	23.2
3	0	5bce24408d1ec32774d781ac	2018-10-23 04:25:52.100	39.7	0	23.2
4	0	5bce244a8d1ec32774d781ad	2018-10-23 04:26:02.360	39.7	0	23.2

3.3 Make dataframe from json data

Dataframe with date and three sensor values(temperature, humidity, luminosity)

```
[ ]  1 iot_data = j1[['date', 'temperature', 'humidity', 'luminosity']]
```

```
[ ]  1 iot_data.shape
```

```
(340230, 4)
```

Json 객체에서 필요한 항목을 선택해서 pandas의 dataframe을 구성한다.

```
[ ]  1 iot_data.head()
```

	date	temperature	humidity	luminosity
0	2018-10-23 04:25:21.349	23.2	39.7	0
1	2018-10-23 04:25:31.594	23.2	39.7	0
2	2018-10-23 04:25:41.855	23.2	39.7	0
3	2018-10-23 04:25:52.100	23.2	39.7	0
4	2018-10-23 04:26:02.360	23.2	39.7	0

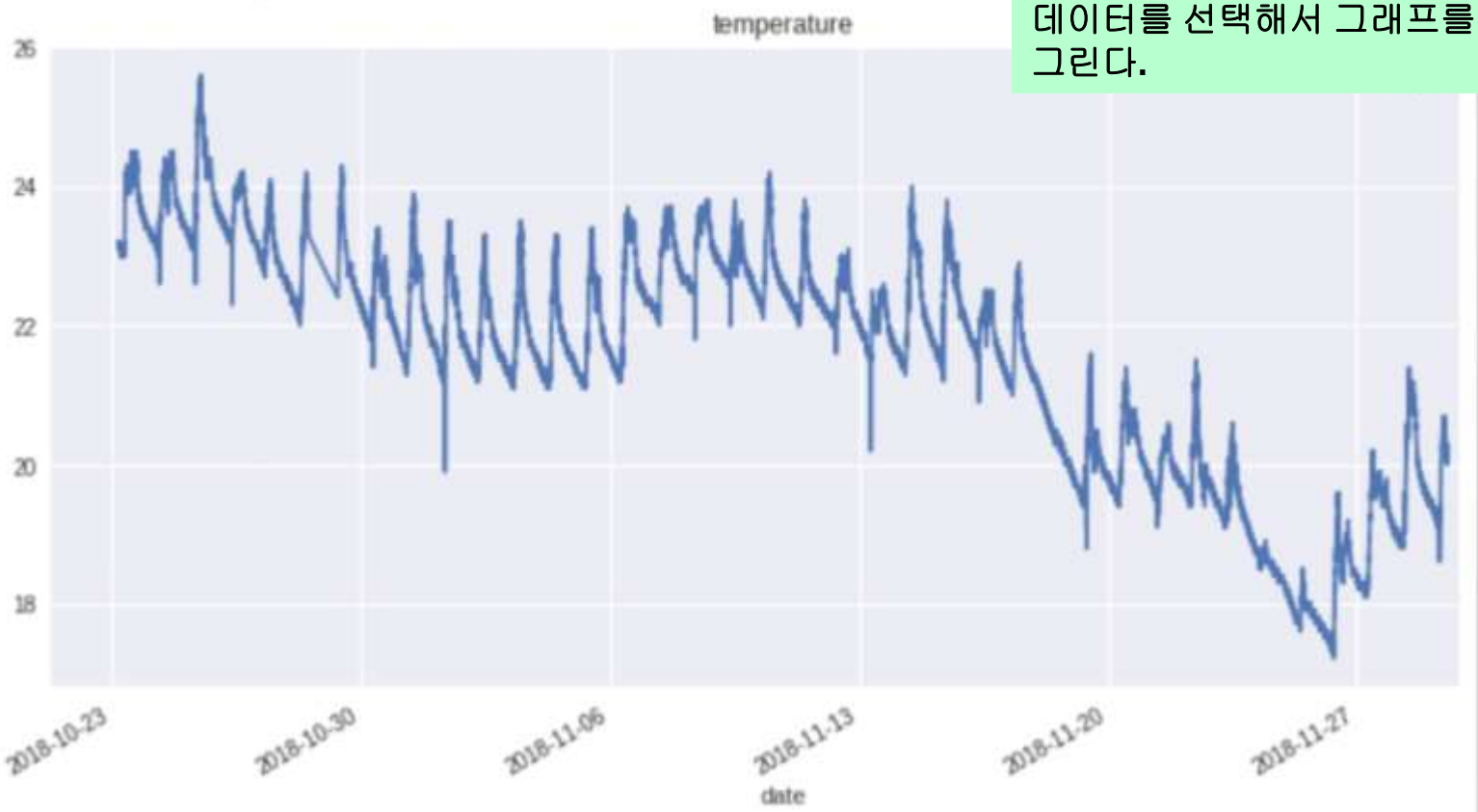
A5.9.8 IOT data mining

3.4.1 Plot iot data (time series)

Plot time series of sensor data

```
[ ]  iot_data.plot(x='date', y='temperature', figsize=(12,6), title='temperature')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5b2596e438>
```



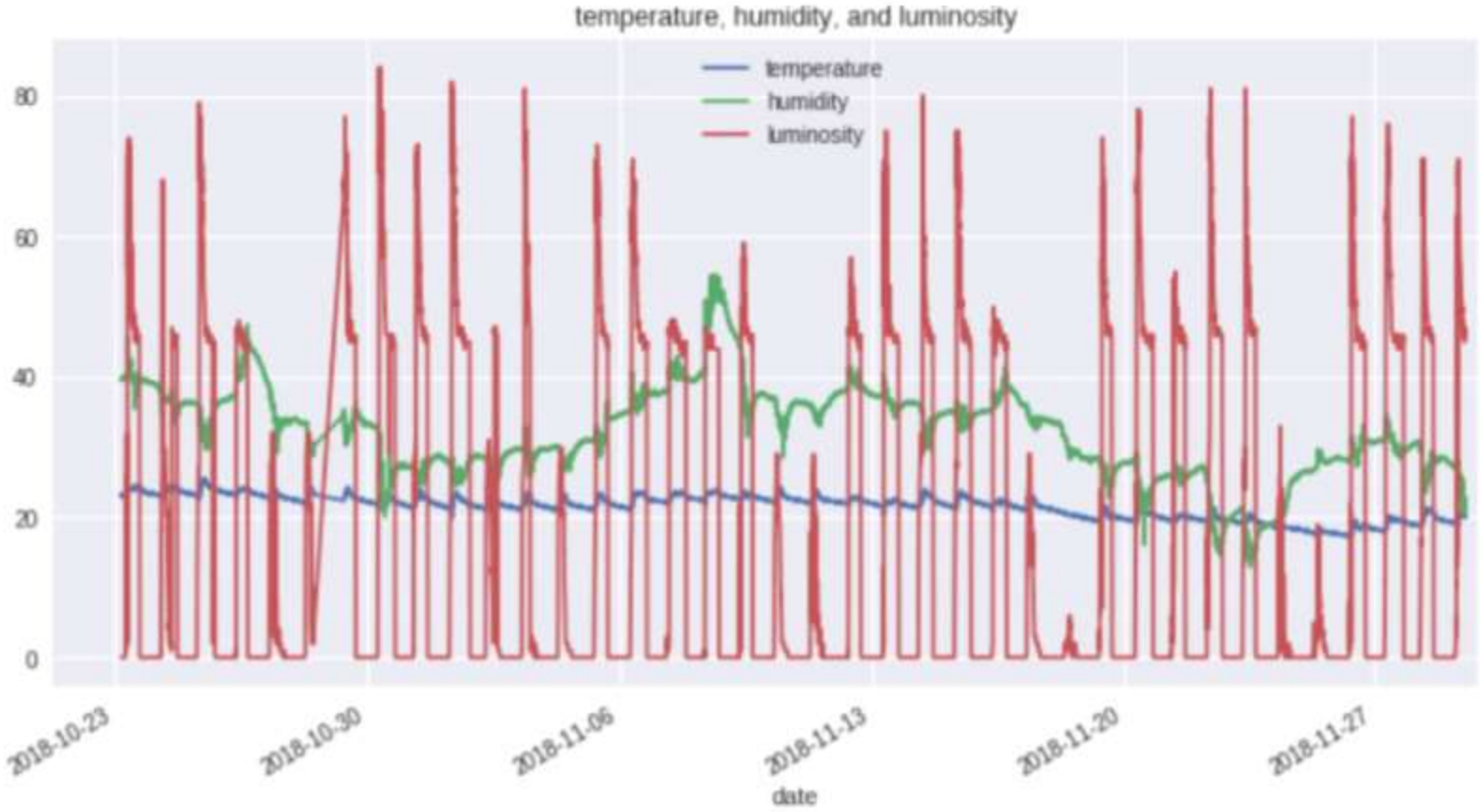
Dataframe에서 시간과 온도 데이터를 선택해서 그래프를 그린다.

3.4.2 Plot iot data (time series)

```
# Plot of ['temperature','humidity','luminosity']
iot_data.plot(x='date', y=['temperature','humidity','luminosity'], figsize=(12,6),
              title='temperature, humidity, and luminosity')
```

```
/usr/local/lib/python3.6/dist-packages/pandas/plotting/_core.py:1716:
  series.name = label
<matplotlib.axes._subplots.AxesSubplot at 0x7f5b28813128>
```

Dataframe에서 시간과 세 개의 센서 데이터를 전부 선택해서 그래프를 그린다.

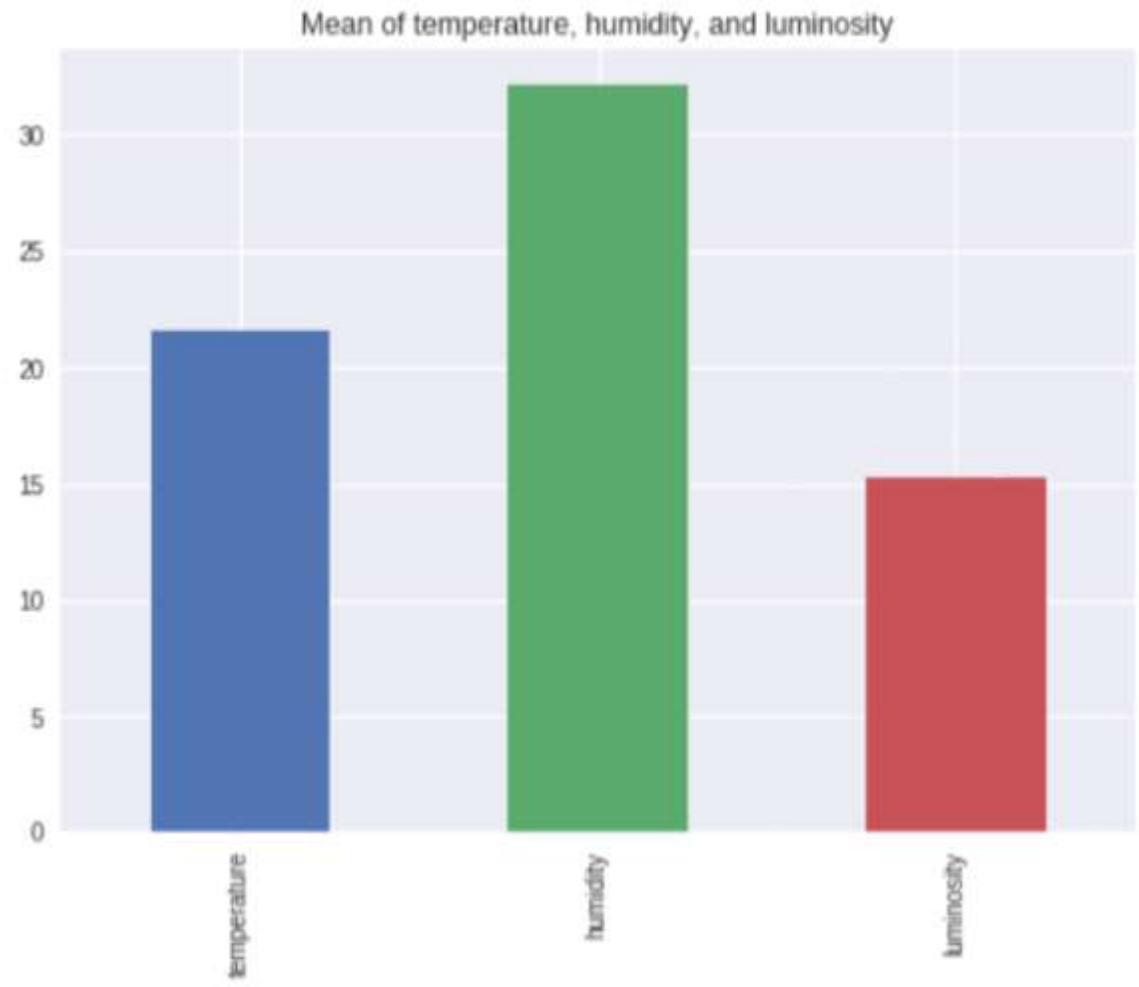


A5.9.8 IOT data mining

3.5 Plot mean of sensor data

```
1 iot_data[['temperature','humidity','luminosity']].mean().plot.bar(figsize=(8,6),
2                                                                   title="Mean of temperature,
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5b297d9470>
```



Dataframe에서 세 개의 센서 데이터의 평균을 구해서 그래프를 그린다.

A5.9.8 IOT data mining

3.6.1 Plot the change of sensor data over various time spans.

Set date as index of timestamp

```
[ ]  1 iot_data.set_index('date',inplace=True)
```

```
[ ]  1 iot_data.info() # timestamp index
```

```
<class 'pandas.core.frame.DataFrame'>
DatetimeIndex: 307849 entries, 2018-10-23
Data columns (total 3 columns):
temperature    307849 non-null float64
humidity       307849 non-null float64
luminosity     307849 non-null int64
dtypes: float64(2), int64(1)
memory usage: 9.4 MB
```

```
1 iot_data.head()
```

date	temperature	humidity	luminosity
2018-10-23 04:25:21.349	23.2	39.7	0
2018-10-23 04:25:31.594	23.2	39.7	0
2018-10-23 04:25:41.855	23.2	39.7	0
2018-10-23 04:25:52.100	23.2	39.7	0
2018-10-23 04:26:02.360	23.2	39.7	0

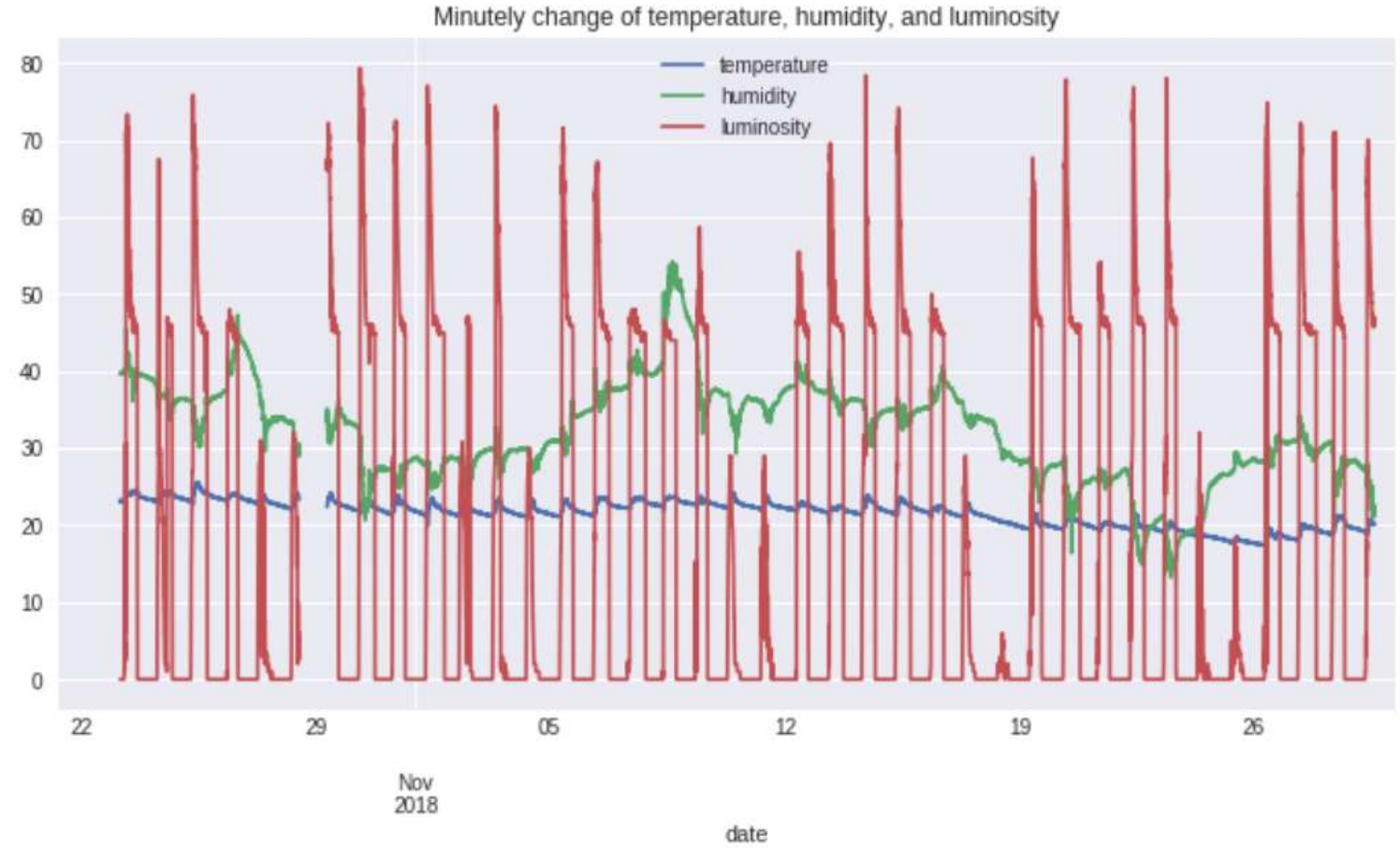
시간(date)을 **timestamp** 형태의 **Index**로 변경해서 데이터를 재구성한다.

A5.9.8 IOT data mining

3.6.2 Plot the change of sensor data over various time spans.
1 분당 평균 그래프

```
# Plot mean of the iot data per every minute
iot_data.resample('60S').mean().plot(figsize=(12,6),
                                     title='Minutely change of temperature, humidity, and lumi
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5b2b57c630>
```

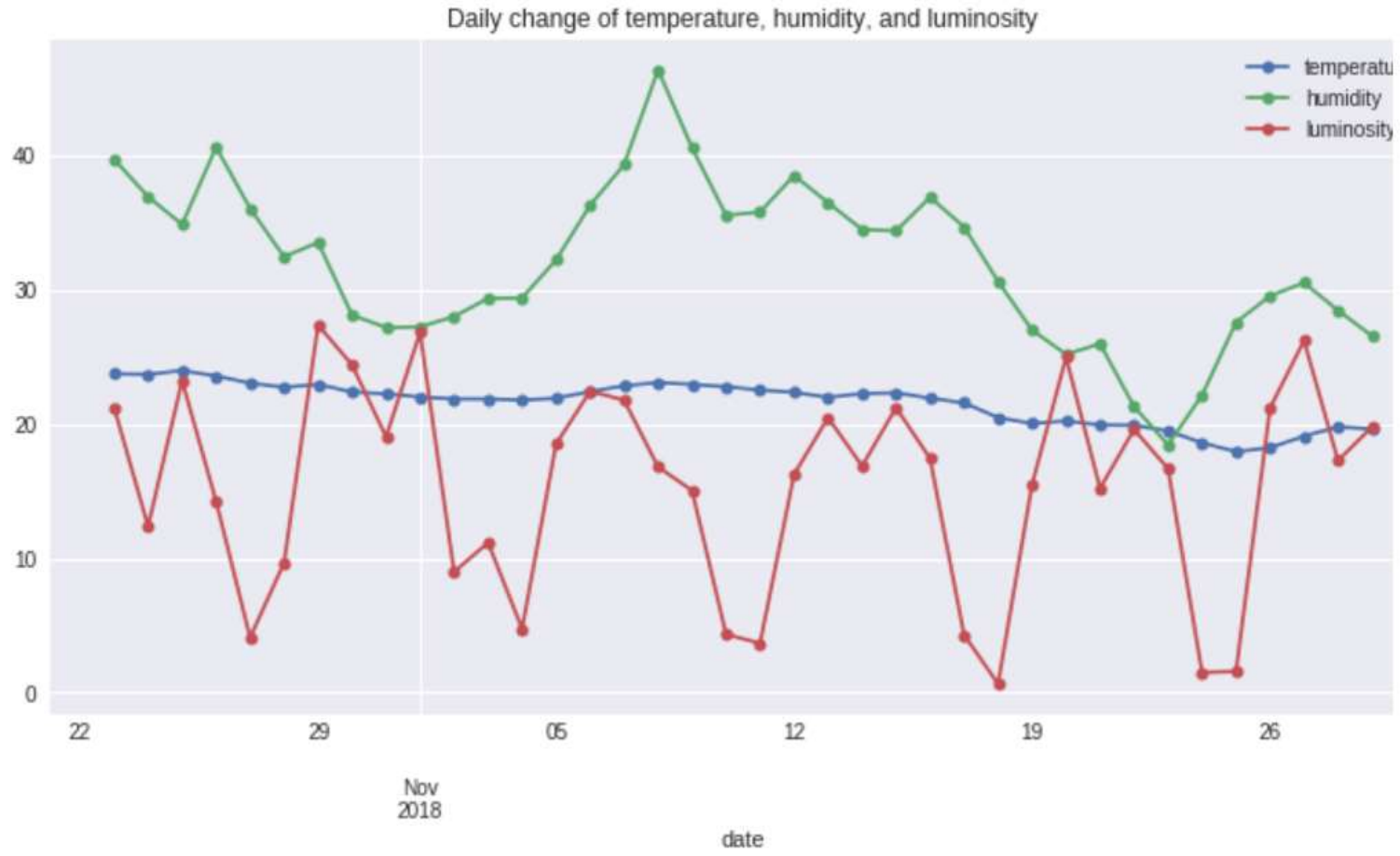


A5.9.8 IOT data mining

3.6.3 Plot the change of sensor data over various time spans.
1 일당 평균 그래프

```
# Plot mean of the iot data per every day
iot_data.resample('D').mean().plot(kind='line', marker='o', ms=6, figsize=(12,6),
                                   title='Daily change of temperature, humidity, and luminosit
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5b2c7fb7f0>
```

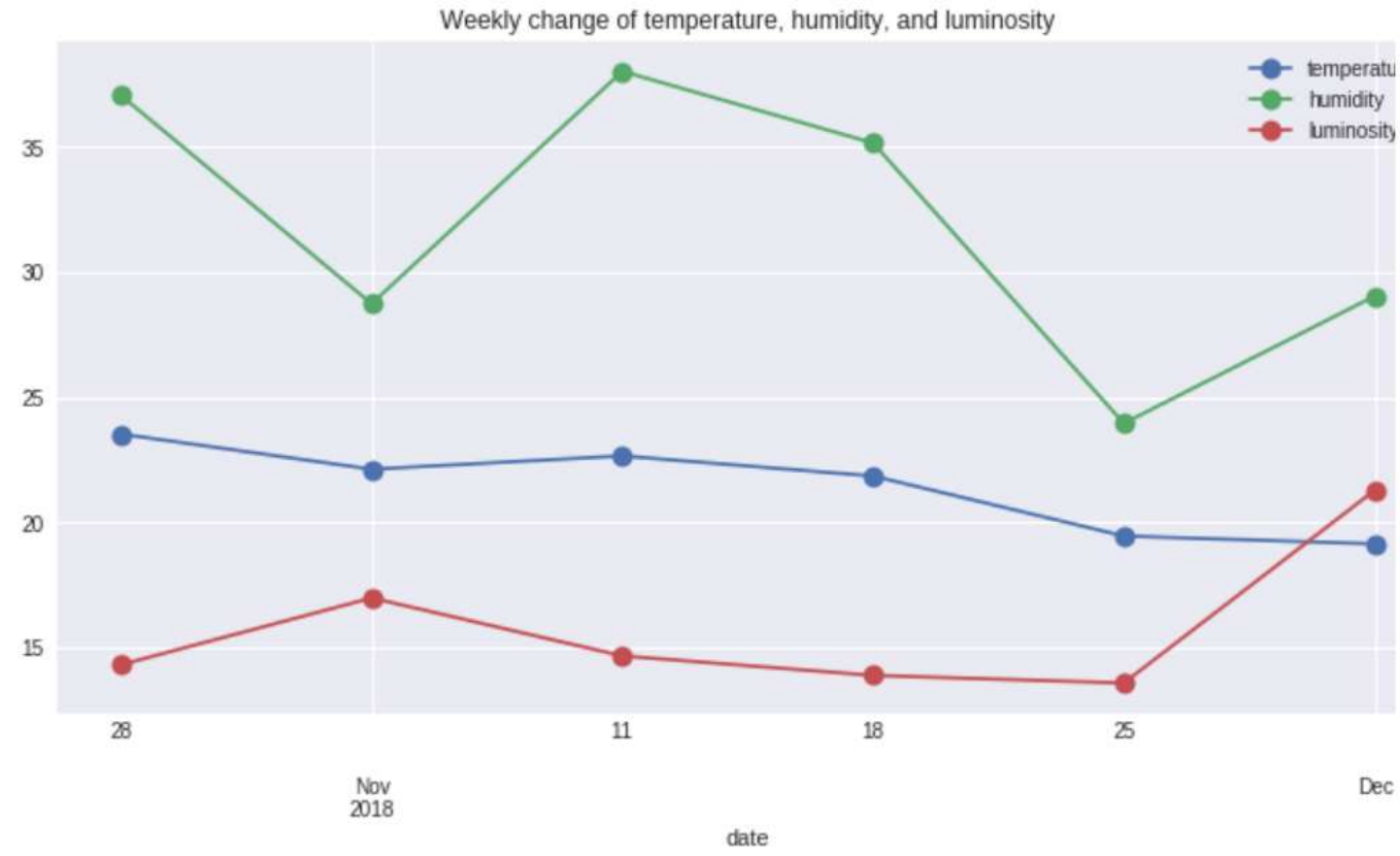


A5.9.8 IOT data mining

3.6.3 Plot the change of sensor data over various time spans.
1 주당 평균 그래프

```
# Plot mean of the iot data per every week
iot_data.resample('W').mean().plot(kind='line', marker='o', ms=10,
                                   figsize=(12,6),
                                   title='Weekly change of temperature, humidity, and luminosi
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f5b2c8f8748>
```



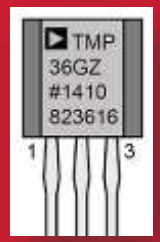
A5.9.8 IOT data mining - DIY

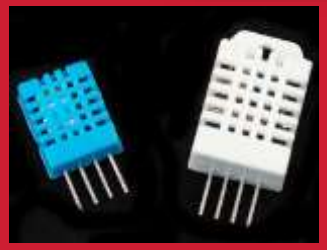
[DIY] 2주 동안 MongoDB에 저장된 데이터를 “AAnn_all.csv” 로 추출한다.

그리고 데이터를 colab으로 업로딩한다.

- iot_csv.ipynb 파일를 iot_csv_aann.ipynb로 저장한다.
- AAnn_all.csv” 데이터를 이용해서 적절한 시간 간격으로 평균 그래프를 그린다.
- Colab에서 만든 iot_csv_aann.ipynb 파일을 github에 올린다.
- 사용한 AAnn_all.csv 파일은 “arnn_rpt12”안에 ‘data’폴더를 만들어서 올린다.

[Practice]





- [wk14]
  - RT Data mining with Google Colab
  - Multi-sensor circuits(cds-dht22)
  - Complete your project
  - Upload folder: AAnn_Rpt12

wk14 : Practice : AAnn_Rpt12

- [Target of this week]
  - Complete your works
  - Save your outcomes and upload outputs in github

제출폴더명 : **AAnn_Rpt12**

- 제출할 파일들

① iot_csv.ipynb
② iot_json.ipynb
③ iot_csv_aann.ipynb
④ All *.ino
⑤ All *.js
⑥ public/All *.html
⑦ public/data/All data (*.csv)

[Upload to github]

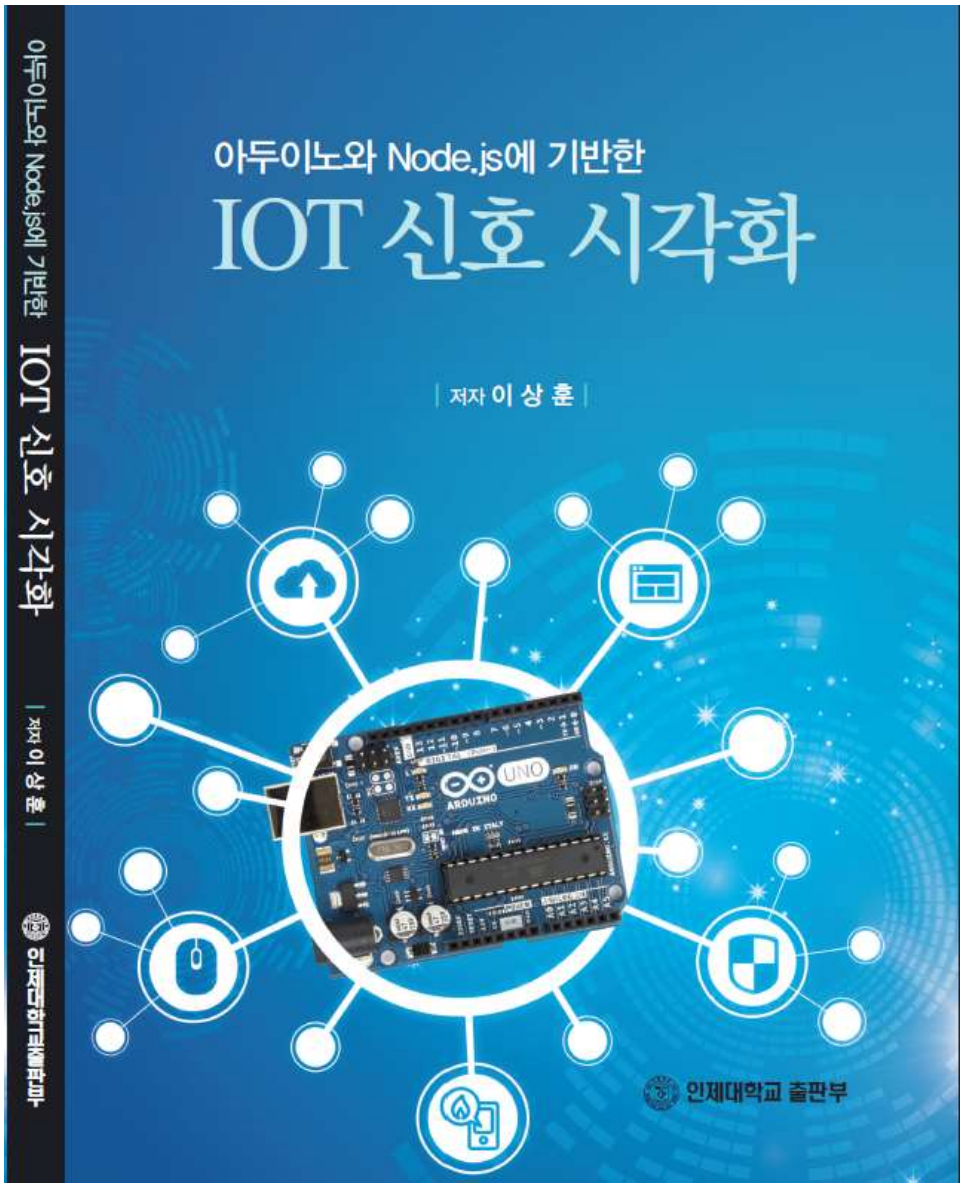
- [wk14]
  - upload all work of this week
  - Use your repo “aann” in github
  - upload folder “aann_rpt12” in your github.

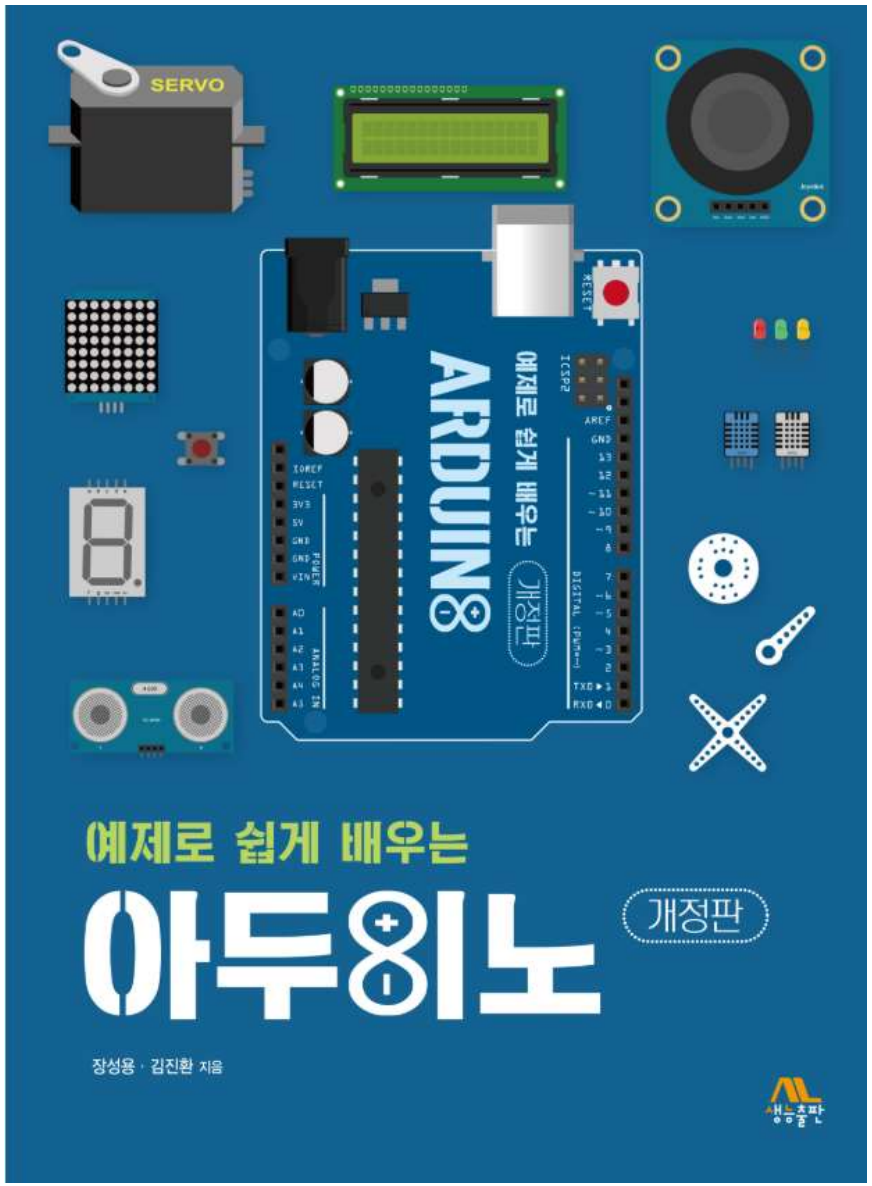
Lecture materials

References & good sites

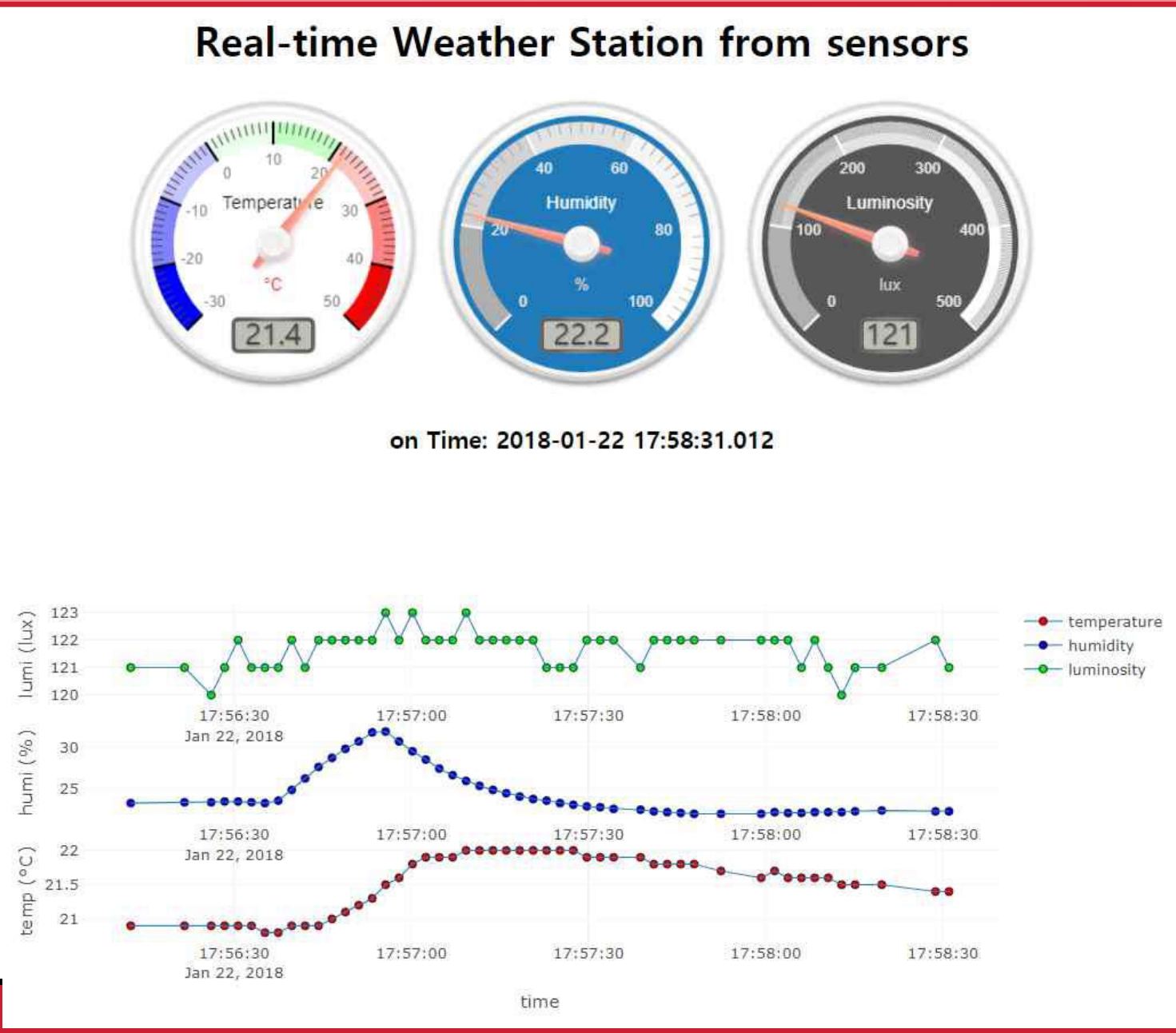
- ✓ http://www.arduino.cc Arduino Homepage
- ✓ http://www.nodejs.org/ko Node.js
- ✓ https://plot.ly/ plotly
- ✓ https://www.mongodb.com/ MongoDB
- ✓ http://www.w3schools.com By w3schools
- ✓ http://www.github.com GitHub

주교재 및 참고도서


아두이노와 Node.js에 기반한
IOT 신호 시각화
| 저자 이 상 훈 |
인제대학교 출판부
아두이노와 Node.js에 기반한 IOT 신호 시각화
| 저자 이 상 훈 |
인제대학교 컴퓨터공학과


SERVO
ARDUINO
예제로 쉽게 배우는
개정판
예제로 쉽게 배우는
아두이노
개정판
장성용 · 김진환 지음

Target of this class



Another target of this class