Faizan Ahmed

COMP 150 - Assignment 1

1. Code specifies when an alarm clock should start making audible sounds.

Link – [Alarm-Clock/main.py at main · ROSHAN-KHANDAGALE/Alarm-Clock (github.com)](https://github.com/ROSHAN-KHANDAGALE/Alarm-Clock/blob/main/main.py)

Code - “status.config(text="Alarm is ringing!")

winsound.Beep(1000, 4000) # Beep sound for 2 seconds”

This is the code I found on the GitHub website. Here we have 2 specific lines, 16 and 17 where we can identify the code “alarm is ringing” and “winsound.Beep(1000, 4000)”. This explains that the clock is given an order to beep for 2 seconds with the text saying Alaram is ringing.

1. Code for a rocket targeting system.

Link - [python - AI projectile targeting, firing a rocket/bullet/etc. so that it intercepts a target - Stack Overflow](https://stackoverflow.com/questions/23842730/ai-projectile-targeting-firing-a-rocket-bullet-etc-so-that-it-intercepts-a-tar)

Code – “tx = target\_pos.x - self.pos.x

ty = target\_pos.y - self.pos.y

tvx = target\_vel.x

tvy = target\_vel.y”

I found this code on the website called Stack Overflow**.** There are no line numbers for the code provided, however, I believe its Line 1, 2, 3, and 4. There are 4 variables tx, ty, tvx, and tvy, each is given certain function.

1. File compression utility algorithm.

Link - [pptx-downsizer/pptx\_downsizer/pptx\_downsizer.py at master · scholer/pptx-downsizer (github.com)](https://github.com/scholer/pptx-downsizer/blob/master/pptx_downsizer/pptx_downsizer.py)

ap.add\_argument("--compress-type", metavar="ZIP-TYPE", default='ZIP\_DEFLATED', help=(

"Which zip compression type to use, e.g. ZIP\_DEFLATED, ZIP\_BZIP2, or ZIP\_LZMA."))

On lines 283 and 284, I think the code is used to decrease the size of the zip file.

1. Weather forecasting algorithm.

Link - https://github.com/yeyintthu9/multi-label-weather-classification/blob/master/main.py

Code - f"\n\nClassification report for {classifier\_name} weather classification model:" line 137

from dataset.multi\_label\_weather\_dataset import MultiLabelWeatherDataset line 20

I believe this is multi weather forecast code and on the line 20, the code orders to import Multi Label Weather Data set. On the line 137, I think the code set classier \_name so the output will be a certain name.

1. E-commerce checkout system process.

Link - [E-commerce-FastAPI/main.py at master · onionj/E-commerce-FastAPI (github.com)](https://github.com/onionj/E-commerce-FastAPI/blob/master/main.py)

Code - app = FastAPI(title="E-commerce API", version="0.1.1",

description=" E-commerce API created with FastAPI and jwt Authenticated")

This code (lines 38 and 39), uses a certain app called “E-commerce API” with version 0.1.1 to check out Electronic commerce. The application figures out the electronic buying and selling of products.

1. Social media post scheduler.

Link - [postScheduler/scheduler/schedule\_post.py at master · SREENATHPGS/postScheduler (github.com)](https://github.com/SREENATHPGS/postScheduler/blob/master/scheduler/schedule_post.py)

Code - username = headers.get("SCHEDULER-USERNAME", None)

api\_key = headers.get("SCHEDULER-API-KEY", None)

media\_link = data.get('media\_link', None)

media\_story = data.get('media\_story', None)

schedule\_date = data.get('schedule\_date', None)

post\_details = data.get('post\_details', {})

In this code (lines 47 – 50, I think it is for the output in the form of a schedule including USERNAME, API-KEY, date, link, story, and details.

1. Fitness app calorie counter.

Link - [caloriestracker/caloriestracker/caloriestracker.py at master · turulomio/caloriestracker (github.com)](https://github.com/turulomio/caloriestracker/blob/master/caloriestracker/caloriestracker.py)

Code - def on\_database\_created(connectionqt):

from caloriestracker.database\_update import database\_update

from caloriestracker.version import \_\_versiondatetime\_\_

I found this code on GitHub. On lines 10, 11, and 12 I believe these codes are importing the data from caloriestracker.database\_update. On the 10th line, the writer defines on\_database\_created.

- Online voting system mechanics.

Code –

data = connection.recv(1024) #4 Get Vote

print("Vote Received from ID: "+str(log[0])+" Processing...")

lock.acquire()

#update Database

if(df.vote\_update(data.decode(),log[0])):

print("Vote Casted Sucessfully by voter ID = "+str(log[0]))

connection.send("Successful".encode())

else:

print("Vote Update Failed by voter ID = "+str(log[0]))

connection.send("Vote Update Failed".encode())

From line 28 – 37, the code is set for the source website where when people vote correctly, it will appear as “ Vote Casted Successfully by the Voter…” or else “Vote Failed by Voter ID…”.

- Automated email response system.

Code –

if \_\_name\_\_ == "\_\_main\_\_":

os.environ.setdefault("DJANGO\_SETTINGS\_MODULE", "colossus.settings")

try:

from django.core.management import execute\_from\_command\_line

except ImportError as exc:

raise ImportError(

"Couldn't import Django. Are you sure it's installed and "

"available on your PYTHONPATH environment variable? Did you "

"forget to activate a virtual environment?"

) from exc

I believe this is the set-up code for the auto email’s description when someone encounter error while emailing. From line 11- 13, This sentences will automatically appear to individuals who encounters the error.