Part 2

If there is a web service such as NOAA that provides historical data, then there is no need to set up automated collection. We can just write a script like what was requested in part 1 of this task to collect and just store the data in a sequel database or csv file, depending on the size of the data.

But I'm guessing this question is asking about how to set up a cloud computer that scraps the internet for data.

I would

- write a bash script for collecting data. (run every minutes)
- write a python script to clean up / process the data. (runs every hour)
- write a bash script to connect and update the RDS database. (runs every hour)
- Use AWS cloudwatch to monitor the EC2 (the linux cloud computer that runs all the scripts)

Collecting data :
This really depends on where you are getting the data from but there are many python libraries that help you scrape websites. You can even use BashScript and wget, if not much re-formatting or modification needs to be done to the data.

Running the script :
Assuming that you have a script, which by running collects data that you want. We need to set up a way of running it automatically at a specific time of day. I personally only have experience with “cron” on linux but you can rent an AWS EC2 instance and set up your script to be run automatically using “crontab” or “cronjob”.

Saving the data : *saving data should also be a script which run on crontab.
Two ways of doing this. 1. Everytime you get the data, make a query and add the data to the database. 2. Save the data in csv or sqlite and make a query to the database once or twice a day to update the database. I would pick option 2, but a cloud computer usually does not have a large disk, so if it is big data, you might want to consider option 1. Although the bad thing about option 1 is that making tons of queries to the database slows down the operation. (probably want to pick somewhere in between. Update DB every hour or something). I am guessing the database is on a different machine like AWS RDS.
Or if you are not using a database. You can also upload data onto cloud storage which can be accessed from your “colleague”, such as AWS S3 or even google drive. Both of these options can actually be done through terminal, meaning writing a BashScript for it is possible.

Handling Errors:
A lot of things can go wrong when you are running a cloud computer. (failed reboot, CPU usage, Memory usage, Disk usage, etc). Again, I can come up with 2 options. 1. Write a bash script that checks the state of the computer, and if it is over some limit, send email/text notification. (this can be done with AWS S3) 2. Set up a cloud computer monitoring system such as AWS cloudwatch. Option 1 has some limitations as if the computer does not successfully reboot after a crash, then the script will not run and therefore you will never get notification. (but since you are writing a script, you can check a more variety of information about the computer.) Option 2, can guarantee that the computer is actually running and couple other statistic, (also has nice visualisation of time-series graph) but the categories of statistics you monitor is limited.

Security :
I'm not sure if this is necessary but just a couple things.
Use network security group and setup inbound/outbound rule to limit number of ip/devices to access database and the linux computer. (also use fingerprint for EC2)