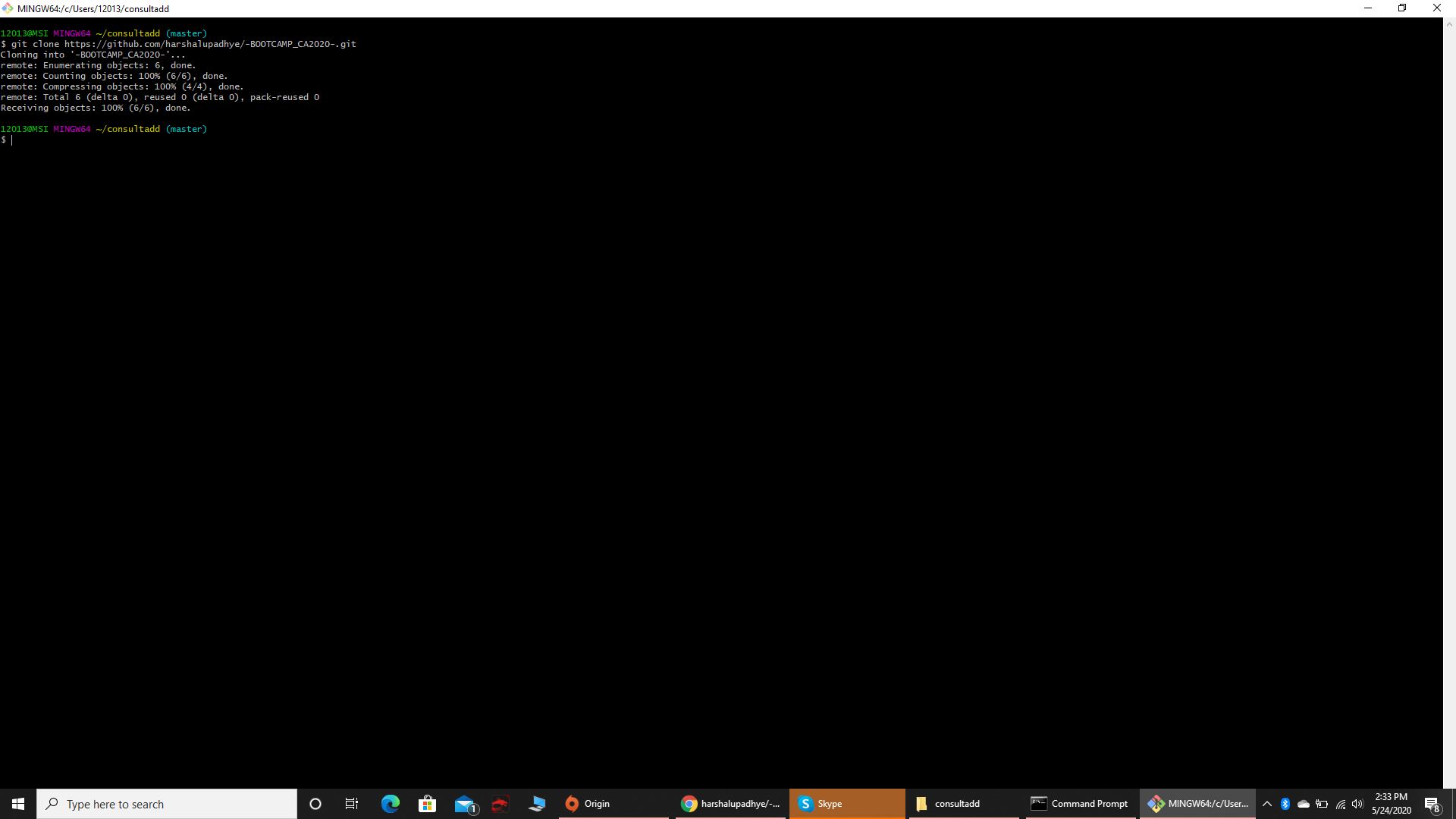
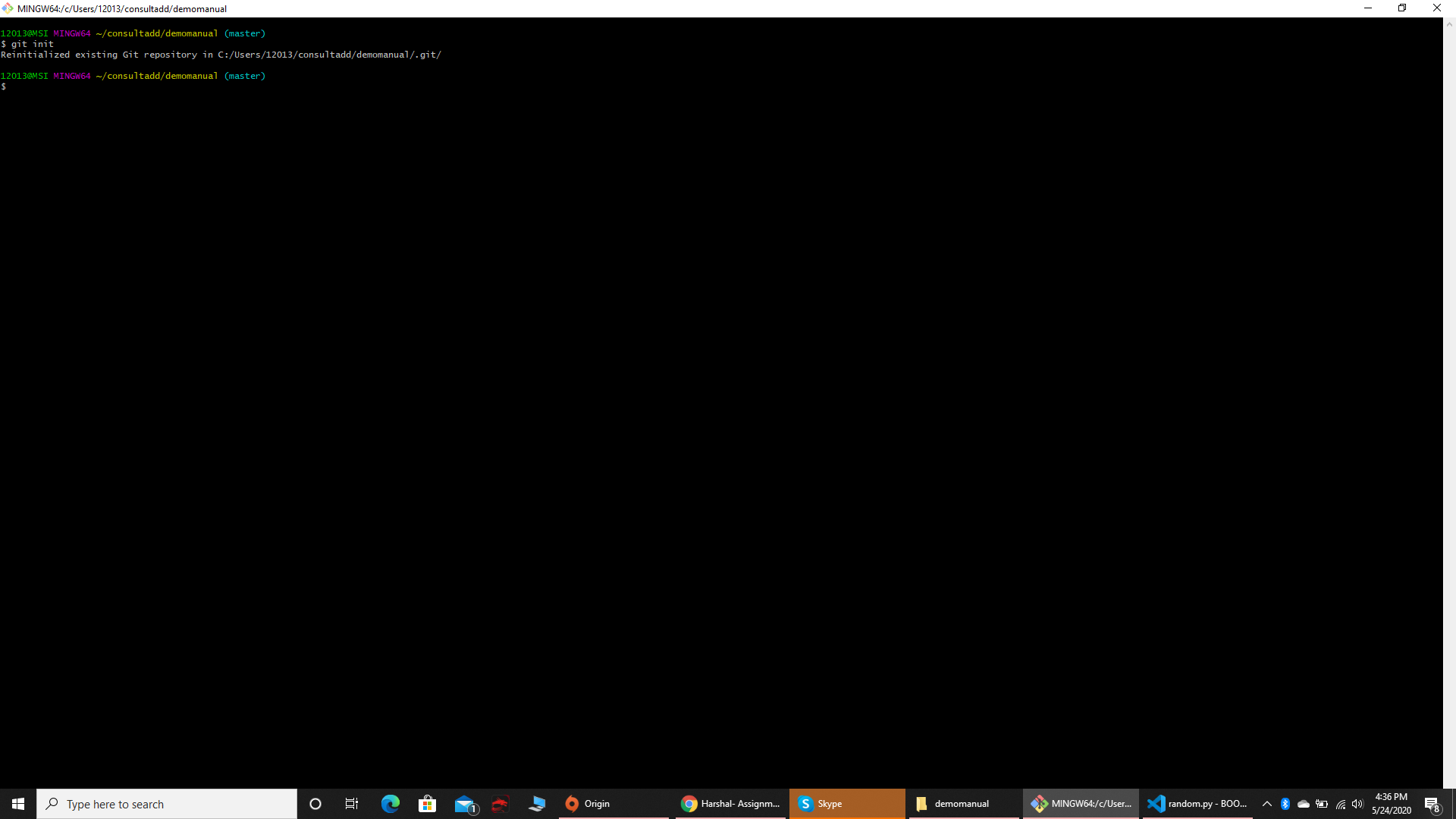
**CA2020 Bootcamp- Assignment Day-3**

**Introduction Version Control System (Git)**

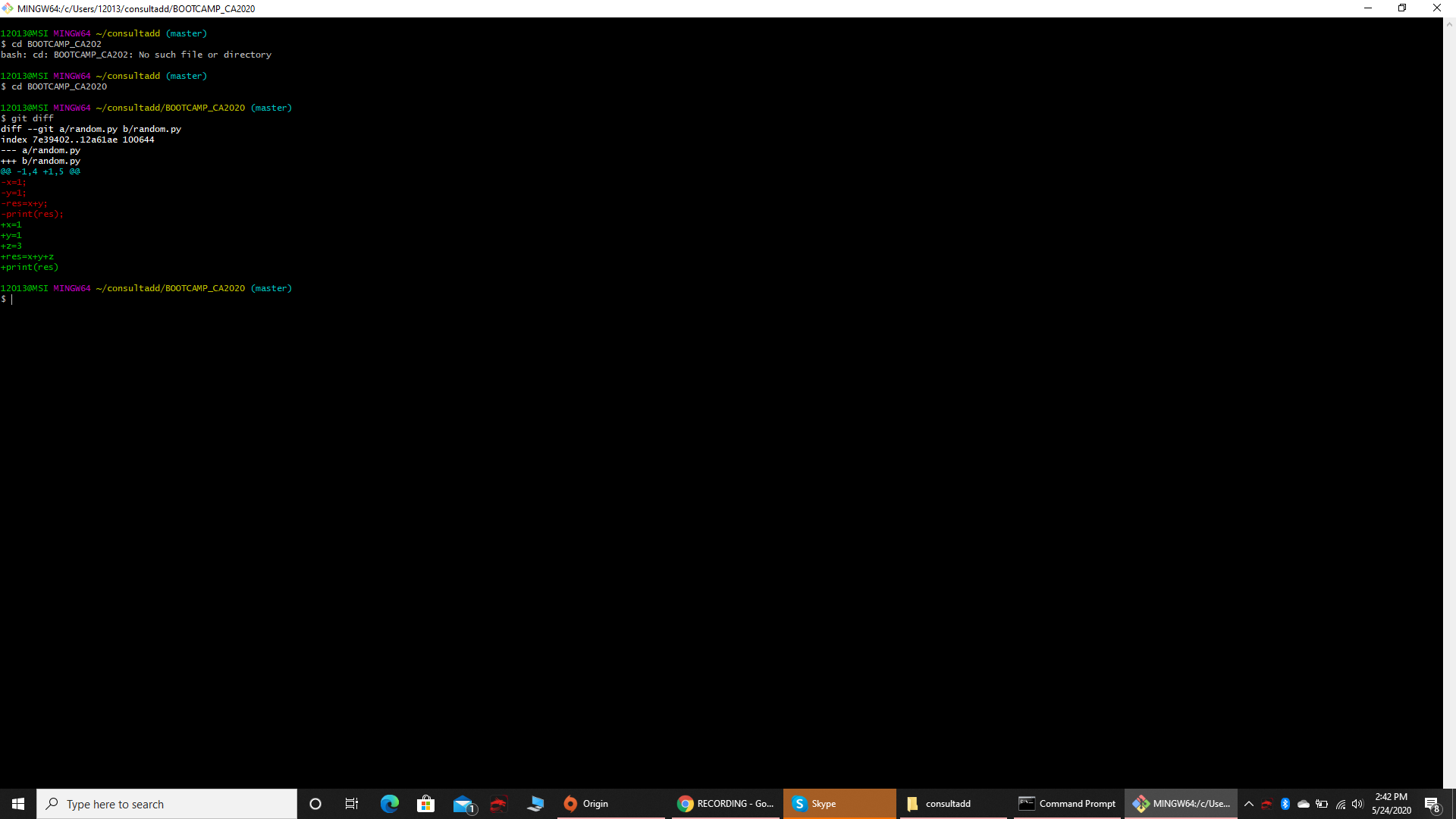
* Practice on the following commands:
  + Git clone



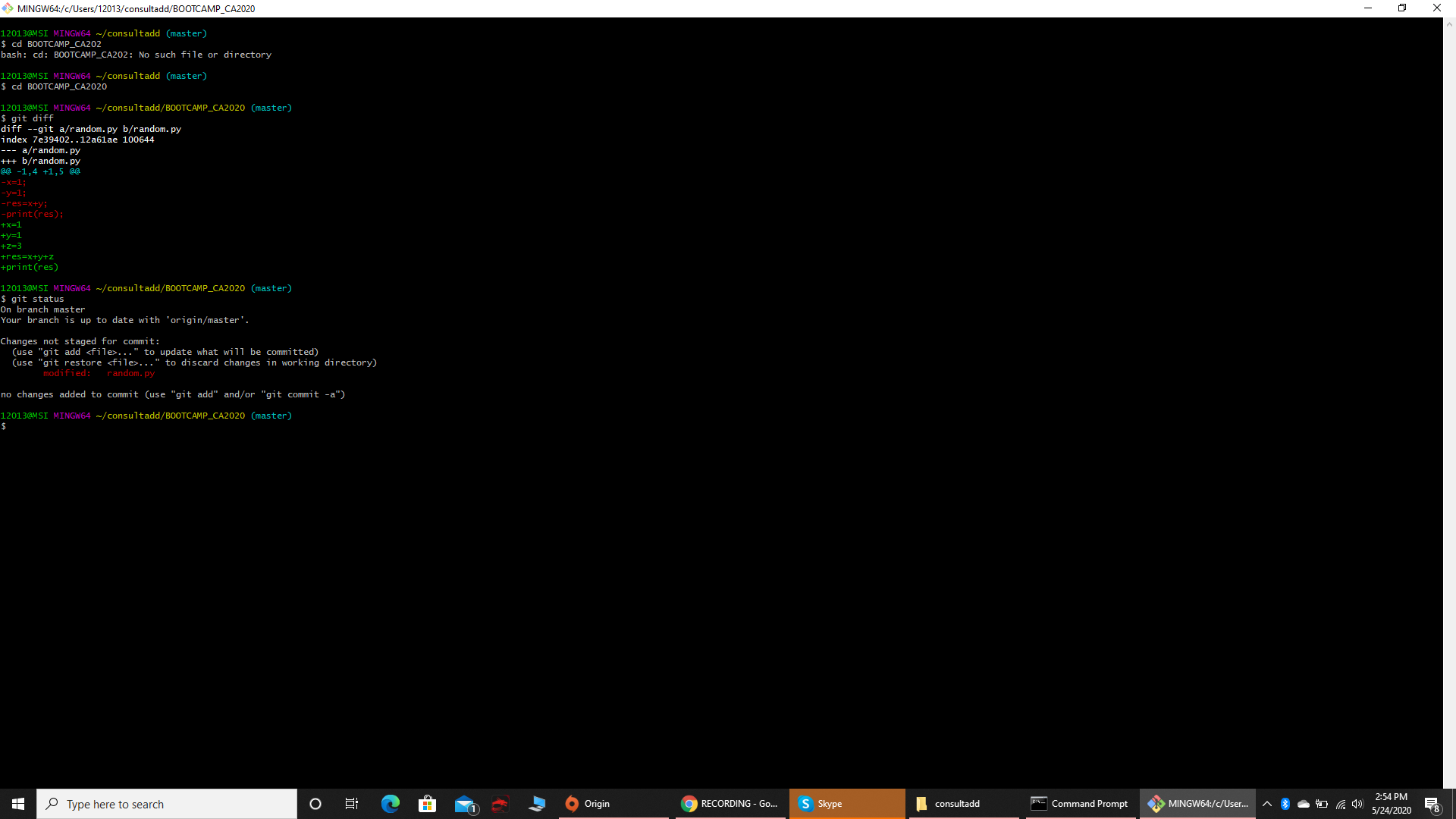
* + Git init

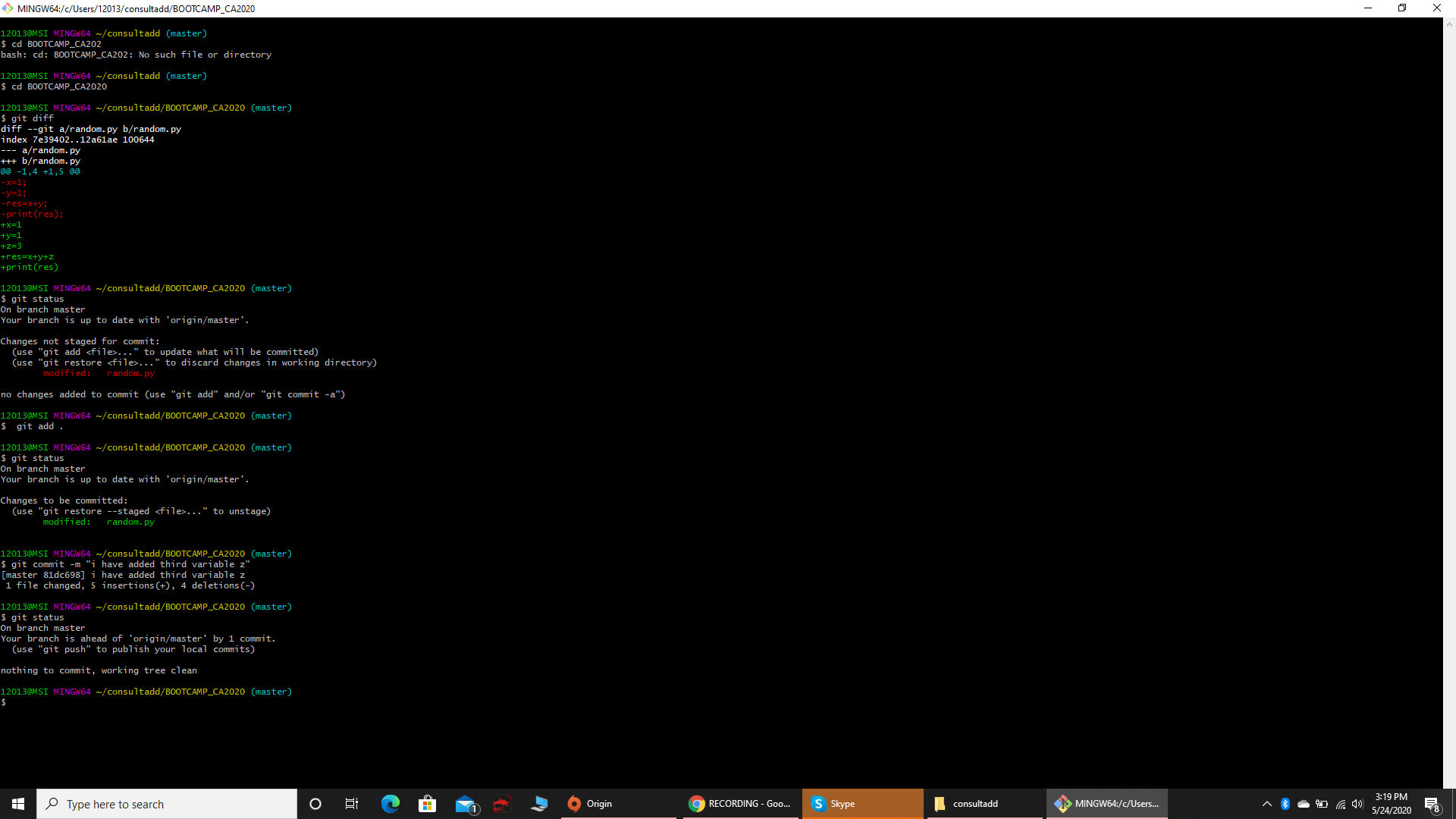


* + Git diff

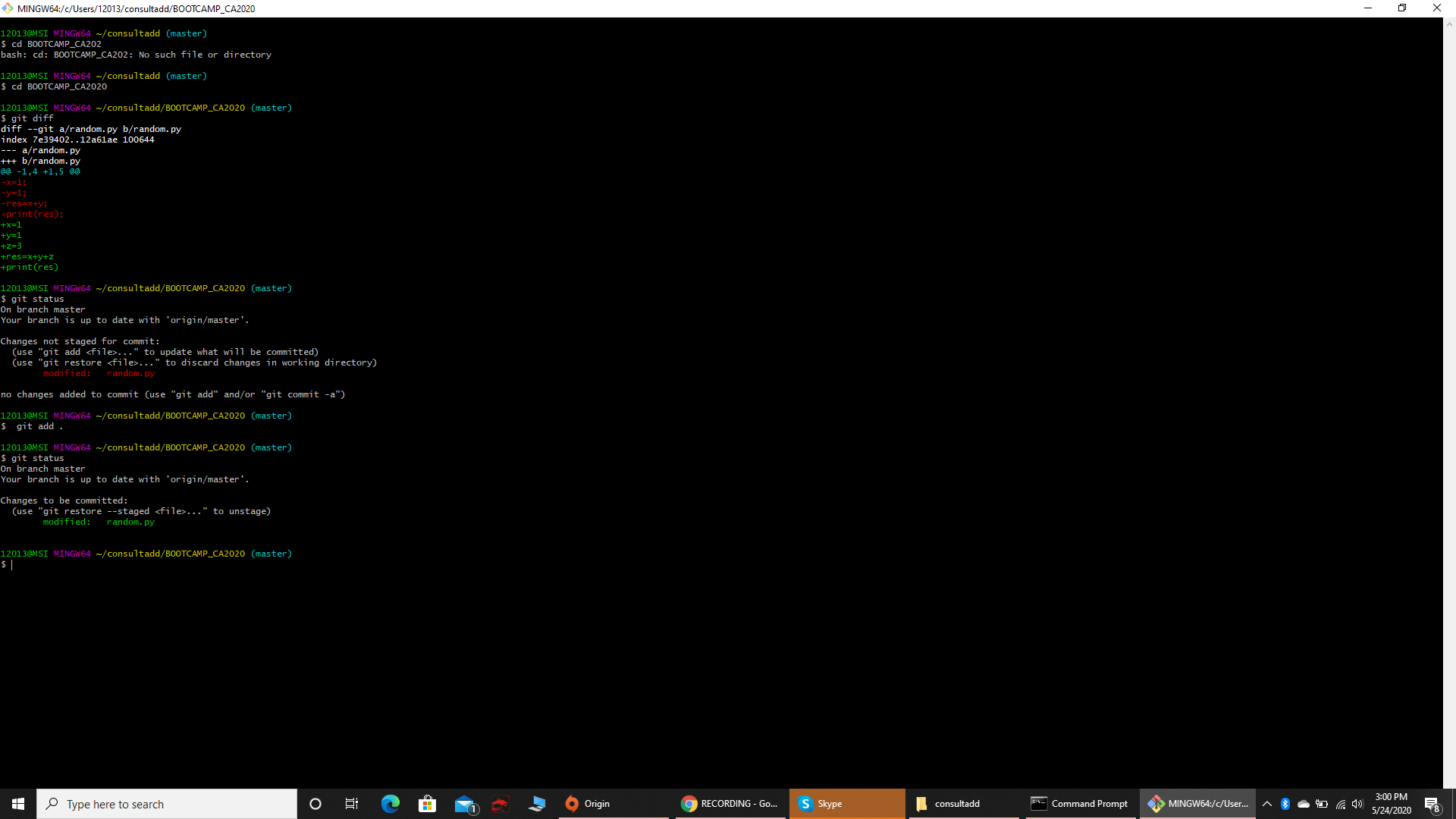


* + Git status

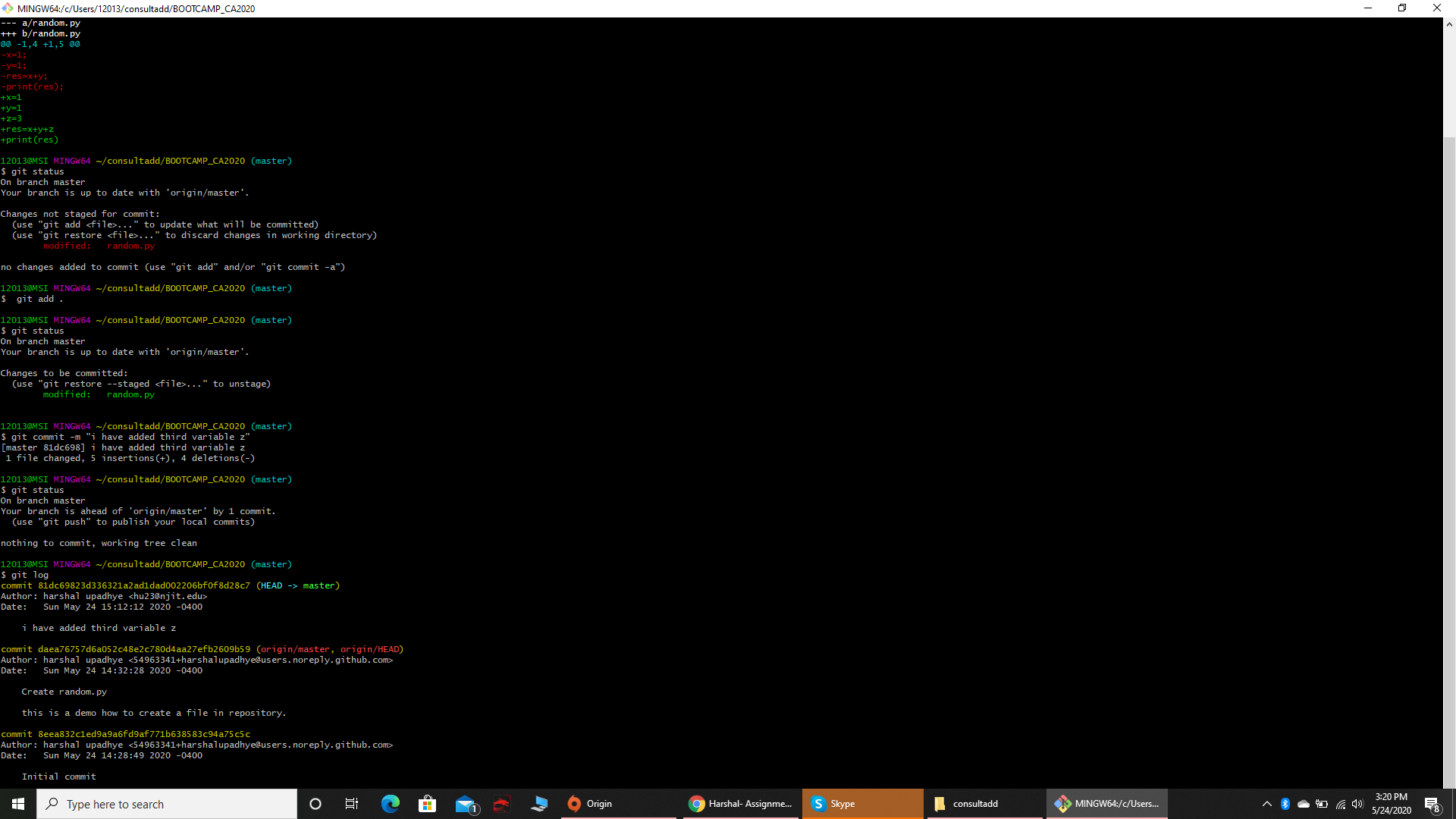




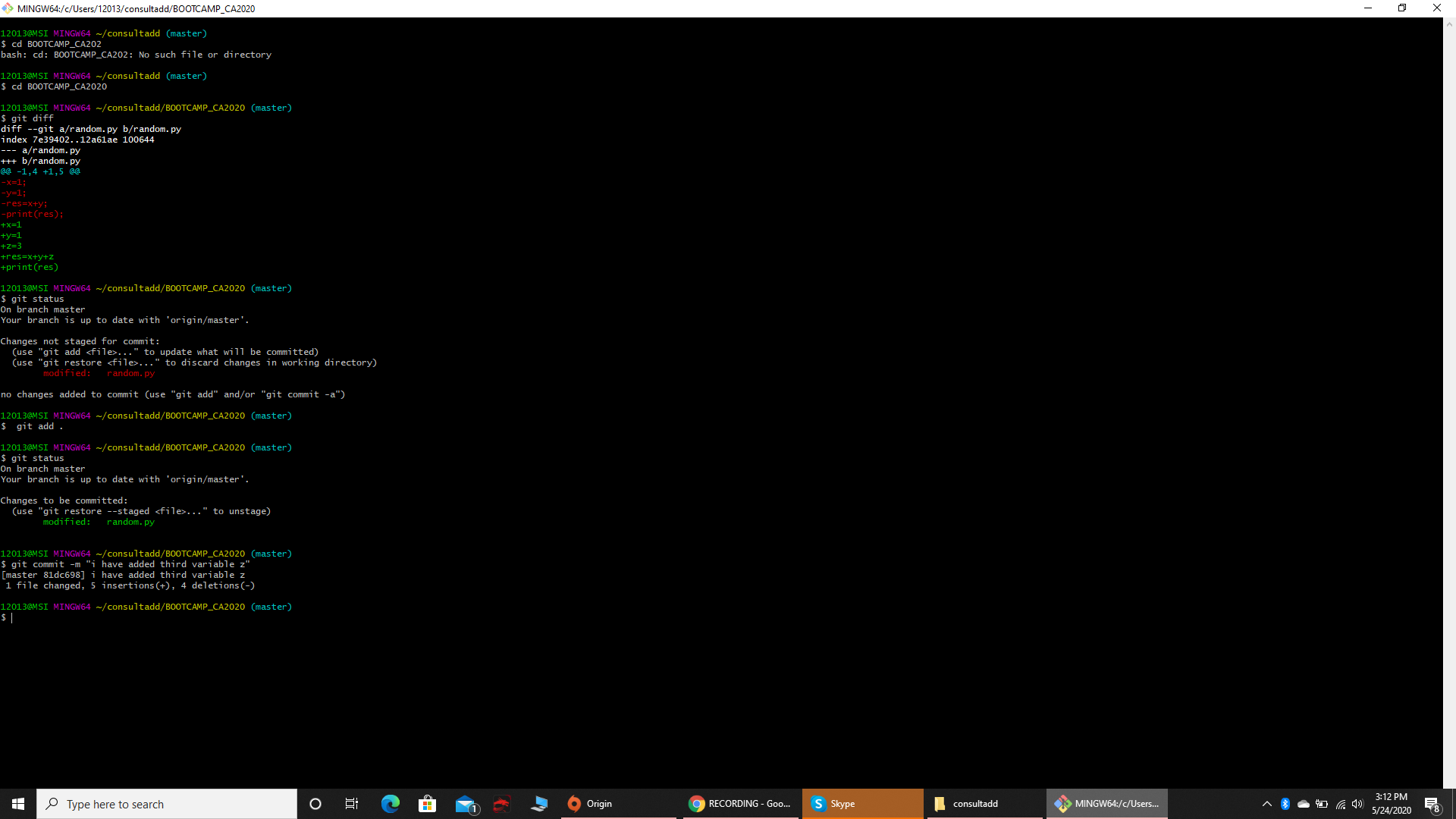
* + add . & Git add filename



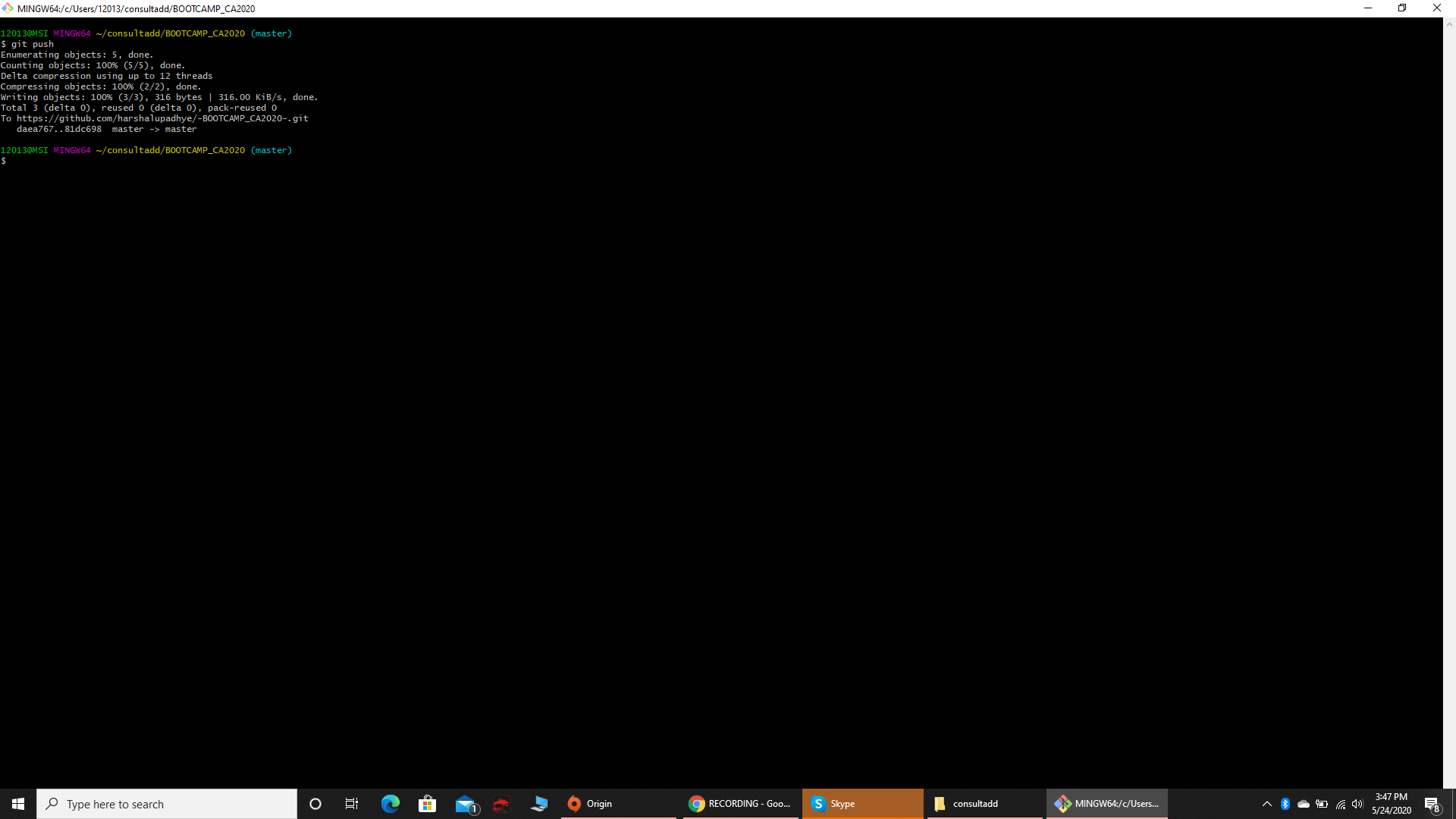
* + Git log



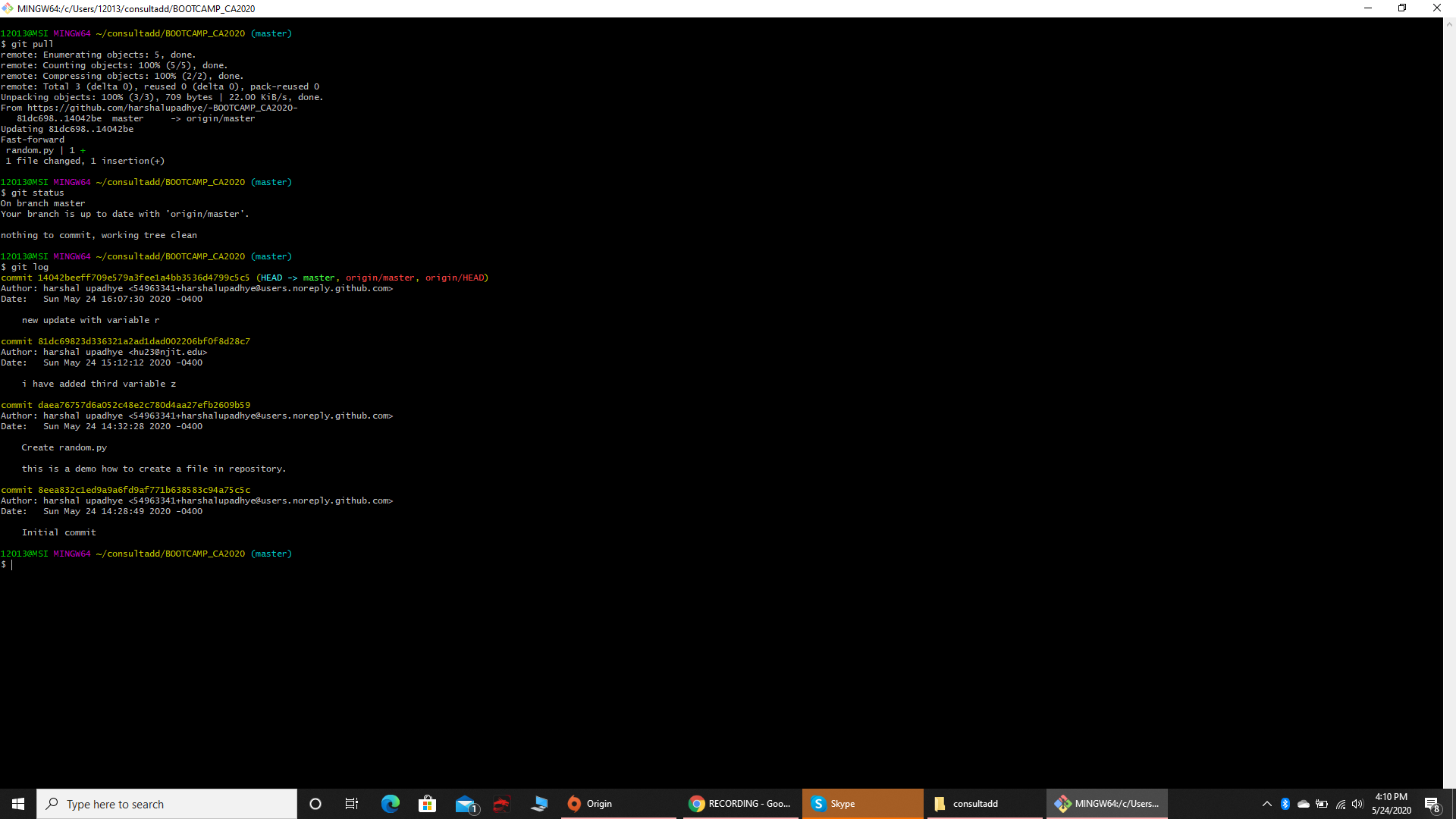
* + Git commit -m “message here”



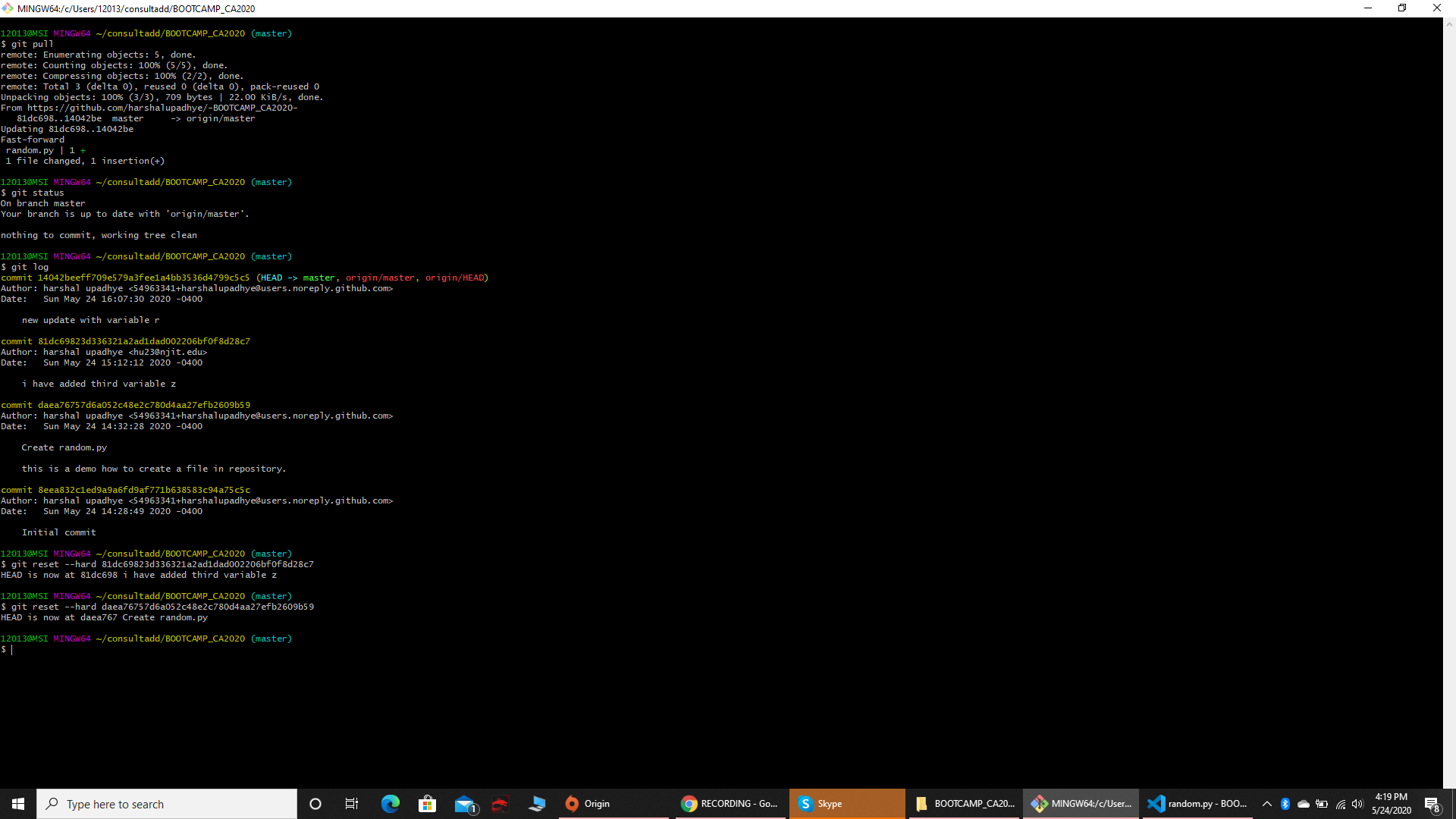
* + Git push



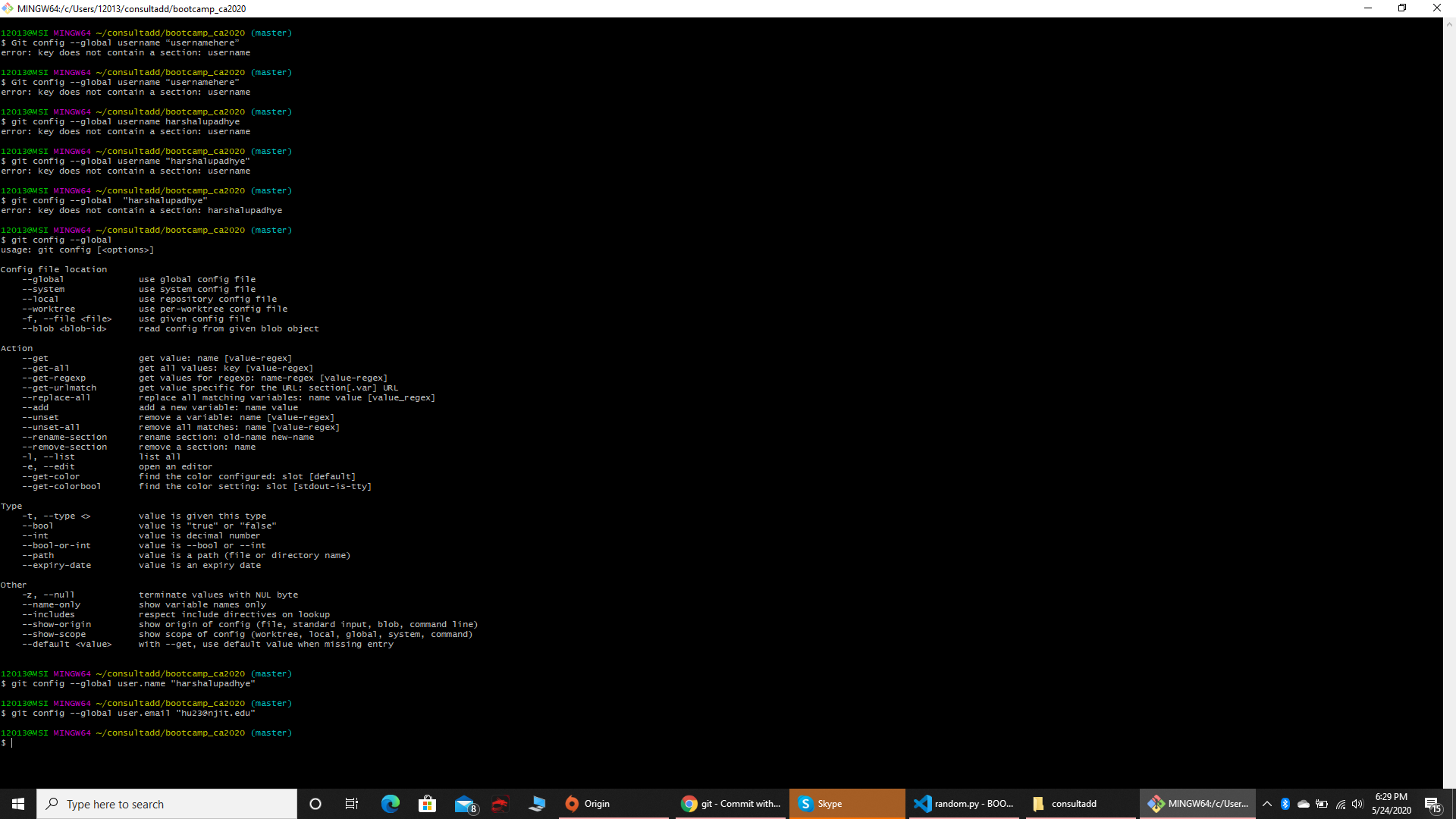
* + Git pull



* + Git reset

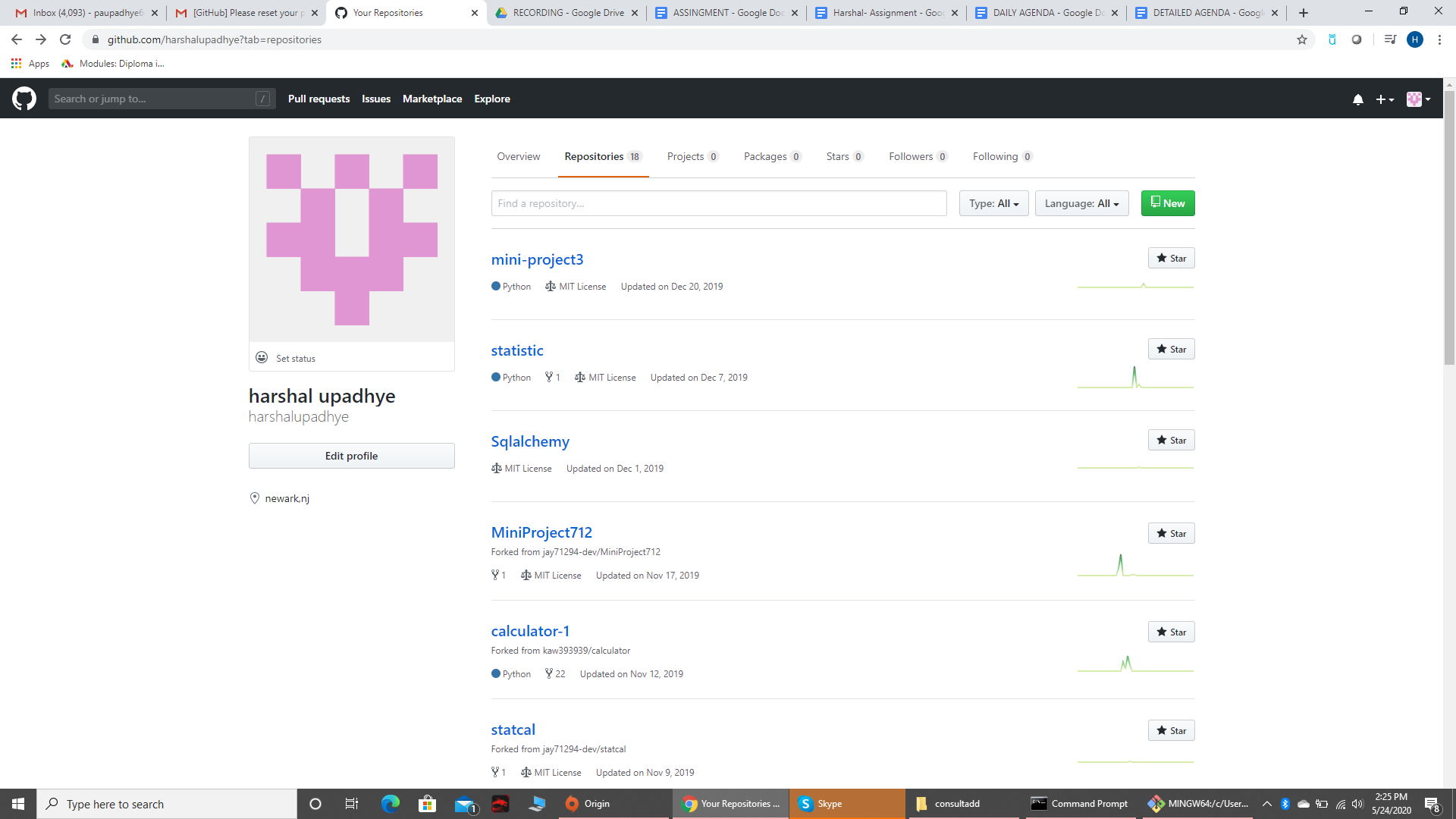


* + Git config --global username “usernamehere”
  + Git config --global user.email “emailhere”

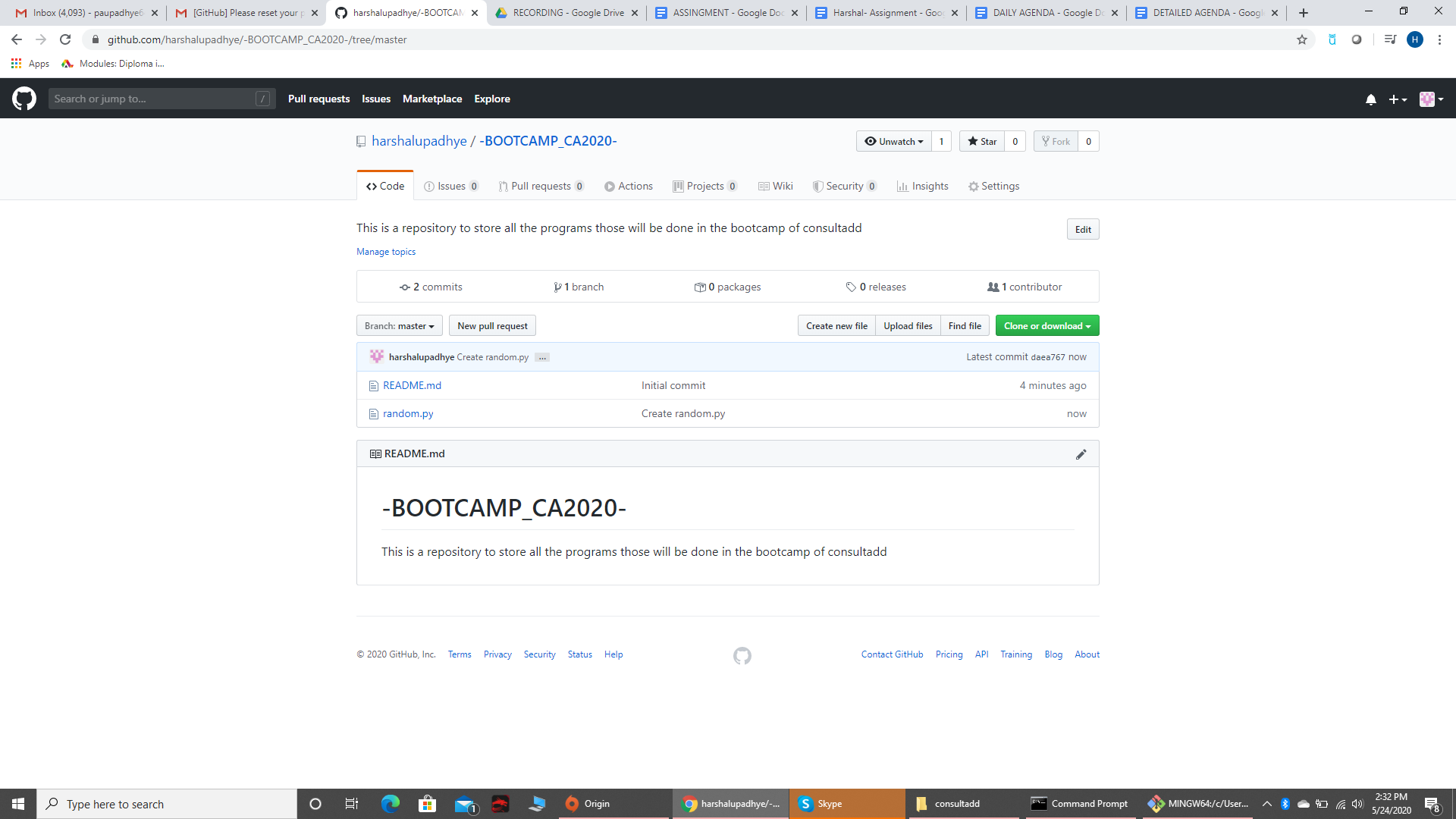


**TASK 02:**

* Create an account on GitHub.

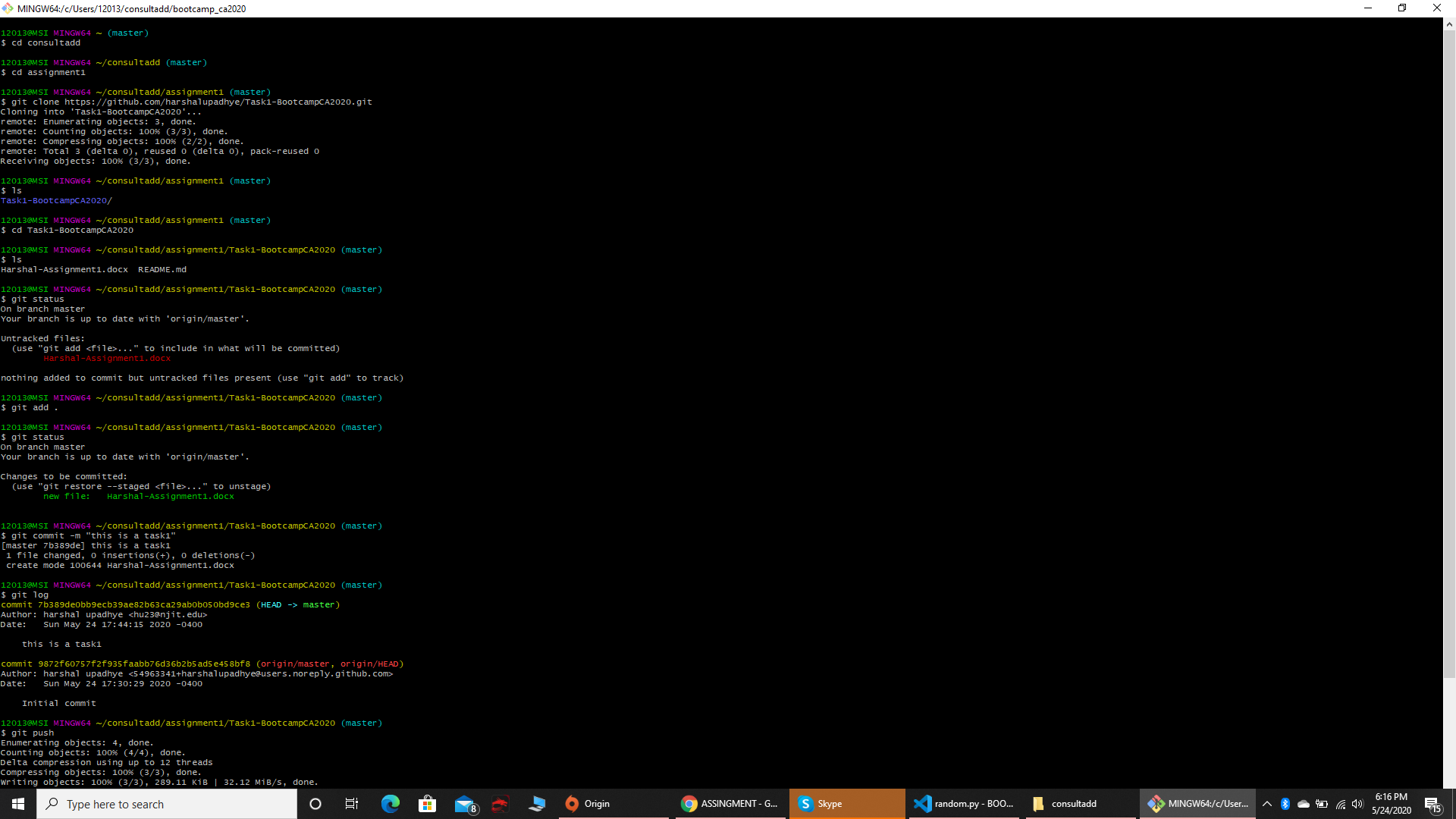


* Create a repository with the name “BOOTCAMP\_CA2020”.
* Make sure to create this repo with README.md file where you can write necessary information of what this repo is all about.



* Kindly, push both of your assignments as a separate file of each day.

**NOTE:** Files that you have created on Google Doc please download that as doc file (Separate day1 and day2 task in individual file ) and push them on git.



* Learn the concepts of Branching and Merging and create a document on it. Push the same document on Git as a separate file named it as day3\_task.txt.

**Ans:**

**Branching: this is the technique by which we can create an identical copy of a code or project and then can make new changes on the copy so that if some error comes then it will not affect the original code or project and we can easily go back to previous version of a code.**

**Steps for Branching:**

1. **Do some work on a website.**
2. **Create a branch for a new user story you’re working on.**
3. **Do some work in that branch.**

**Basic commands for branching:**

1. **Git branch : to see all the branches those are made.**
2. **Git branch <branch\_name>: to create a new branch from master**
3. **Git checkout -b <branch\_name>: this is to create a new branch and automatically switch on it.**
4. **Git checkout <branch\_name>: this is to switch the branch from one branch to another.**
5. **Git commit -a: commit all the changes those are made on the particular file while being on particular branch**
6. **Git branch -d <branch\_name>: this is to delete a mentioned branch.**

**Steps for Merging:**

1. **Generate a pull request and that pull request has to be accepted**
2. **Push the code to the master branch.**

**Basic commands for merging:**

1. **Git merge <branch\_name>: to merge particular branch to master branch**

**TASK 03:**

* What is an Elastic IP and how it is different from Dynamic IP?

**Ans:**

**What is an Elastic IP?**

**Elastic IP addresses are used by AWS to manage its dynamic cloud computing services. Within the AWS infrastructure, customers have** [**virtual private clouds (VPCs)**](http://blog.datapath.io/the-easy-setup-to-redundant-connectivity-between-aws-vpcs)**. Within the VPCs, users have instances. The Elastic IP address is what is used to advertise the data within the instance to the public internet.**

**Within the AWS cloud environment, AWS states** [**that the Elastic IP is used for dynamic cloud computing**](http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/elastic-ip-addresses-eip.html)**.an Elastic IP is a combination of a public IP address and a static IP address. It allows you to continue to advertise AWS instances within your AWS network infrastructure.**

**Difference between Elastic IP and Dynamic IP:**

|  |  |
| --- | --- |
| **Elastic IP** | **Dynamic IP** |
| **The IP is assigned to the account even if you stop the session and rejoin it, the same IP will be assigned** | **The IP would be dynamic if you stop the session and rejoin it then new public IP would be assigned.** |
| **It stays same for the particular account** | **It is assigned to you from the IP Pool.** |
| **The elastic IP address is persistent. It's associated with your account until you choose to release it.** | **The Dynamic IP address is not persistent. It's associated with your account until you choose to release the terminated session so in the new session you can have a new one.** |
| **If you have Elastic IP in your account and it's not in use, then you will be charged for it.** | **Over here as soon as you stop the session it goes to IP pool so you would not be charged for it** |

* What is the Client Server Model, Explain in detail?

**The client-server model describes how a servers provides resources and services to one or more clients. Examples of servers include web server, main server, and file servers. Each of these servers provide resources to client devices, such as desktops,laptops, tablets, and smartphones. Most servers have a one-to-many relationship with clients, meaning a single server can provide resources to multiple clients at one time.**

**When a client requests a connection to a server, the server can either accept or reject the connection. If the connection is accepted, the server establishes and maintains a connection with the client over a specific Protocol. For example, an email client may request an SMTP connection to a mail server in order to send a message. The SMTP application on the mail server will then request authentication from the client, such as the email address and password. If these credentials match an account on the mail server, the server will send the email to the intended recipient.**

**Online multiplayer gaming also uses the client-server model. One example is Blizzard's Battle.net service, which hosts online games for World of Warcraft, StarCraft, Overwatch, and others. When players open a Blizzard application, the game client automatically connects to a Battle.net server. Once players log in to Battle.net, they can see who else is online, chat with other players, and play matches with or against other gamers.**

**While Internet servers typically provide connections to multiple clients at a time, each physical machine can only handle so much traffic. Therefore, popular online services distribute clients across multiple physical servers, using a technique called distributed****computing. In most cases, it does not matter which specific machine users are connected to, since the servers all provide the same service.**