1. **What is Git?**

**Ans.** Git is a version control tool that saves changes in groups of files and you can revert back if needed. It takes data more as a set of snapshots of a miniature filesystem. Whenever we commit or save our projects in git, it takes a picture of what all files look like and stores a reference of snapshot. It things data as stream of snapshots.

We can setup a repository. It allows us to create a repository and share amongst our team members. They can clone it on to their local system and work on it. Git is a distributed version control.

In git files are stored in all 3 different stages i.e. modified files, Staged files and committed files. You can modify those file in working directory. You can notify those changes in staging area. After you commit then file changes that are saved to the Git directory.

1. **In what form is Git being used in your company?**

Ans. To share resources like text books, Quiz question papers and presentations etc; among a group of students. As a student, I used it to submit my works by pushing them on to master repo. I used command line to do all activities like push, pull, commit and status check of a git repo.

1. **What are the different tools that you integrated Git with in you company?**

**Ans.**

I learnt that we can integrate Git with Maven. By doing so we can achieve SCM implementations like different Fetch and Push URL’s. We can work with branches. By invoking maven-scm actions we can access the upstream repository.

Even we can integrate Git with Jenkins. Jenkins works best when builds are triggered after each and every commit. That provides you the developer with the most instant feedback possible. We can install Git plug in in Jenkins and it provides support in receiving push, pull notifications from Git commit hooks and use those notifications to trigger jobs with in Jenkins, so that we can setup push commit hooks.

Recently I leant that AWS codepipline allows to integrate Git, Jenkins into any stage of release process. By integrating them we can register a custom action that allows us to hook servers into a build pipeline.

1. **What are Git hooks?**

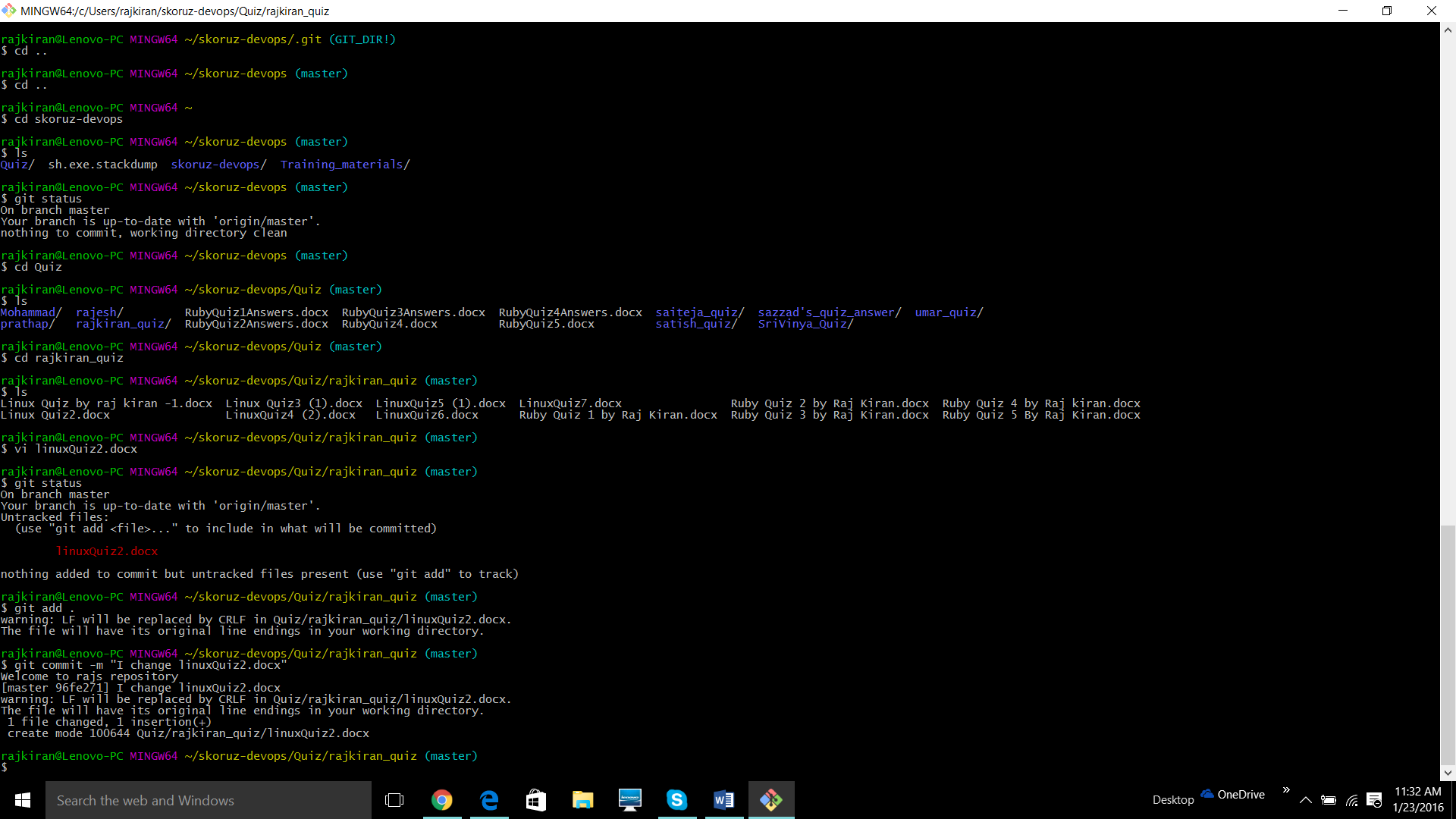
**Ans.**

Hooks are custom scripts when certain important actions occur in Git repository. All hooks are stored in hooks directory in Git directory. Hooks are divided into two types

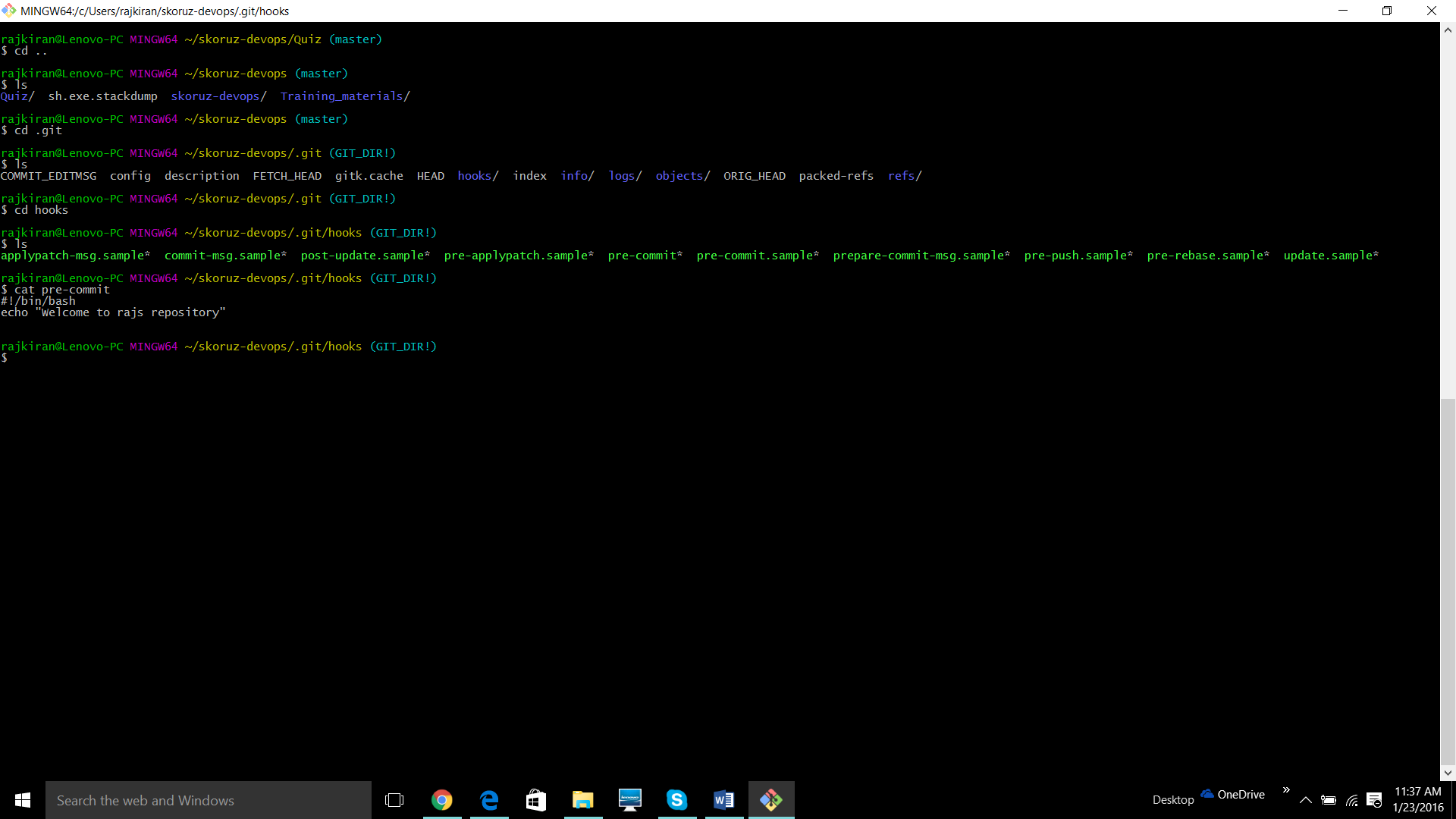
* Client side hooks include Commit workflow hooks like pre-commit – before, prepare-commit-msg, commit-msg and post-commit etc. This scripts is to enforce a policy.
* Server side hooks include pre-receive, update and post receive hooks. This hooks run whenever network operations such receive and update actions occur.

1. **Give us an example of a Git hook that you implemented?**

**Ans.**



I tried to implement changes in pre commit hook. I actually added following script into pre-commit hook



You can see that change had effected my output in figure 1. And welcome message is displayed when a commit is executed. It is example of hook implementation.

1. **List all commands a developer uses to push to git starting from clean workspace**

**Ans**

* + cd gitrepodirectory
  + git status
  + ls
  + git pull
  + git status
  + git add .
  + git commit –m “I made x change in x file”
  + git push origin master
  + git status
  + git checkout

1. **Difference between Git push and Git pull**

**Ans**

**Git pull.** This command is used to capture changes in master repository. Once the changes have been delivered, others can pull them down and work with them in our local system..

**Git push**. is sending your committed coded to a remote server. And you can update your work on to master or branch.

1. **Difference between Git Rebase and Git marge**

**Ans :** Both commands are designed to integrate changes from one branch into another.

git rebase: You can move entire branch to begin on the tip of the master branch using rebase. You can get much cleaner project history then marge. You can see upstream changes.

git merging avoids all of the potential pitfalls of rebasing. If you want to preserve the complete history of project and avoid risk of re writing public commits, you can stick with git merge.

1. **How do you what branch you are currently on?**

**Ans** git branch

1. **How do you know what other git branches exist on the git server?**

**Ans.**  git branch –a

We can even use command called gitk and look into GUI. There we can clearly view all branches, master etc.

1. **What is a git remote?**

**Ans.** It manages set of repositories whose branches we track. It shows list of all remotes. By using this command we can perform different actions like add a remote for a repository at particular url, rename, remove, set-branch, set-url etc.

1. **How can you temporarily save changes before pulling/ merging or switching branches?**

**Ans.** Rebase option can be used to ensure a linear history.

1. **Difference between git braches and git tags?**

**Ans.**

A branch is used if we have 2 different versions of repository at the same time.

A tag is a way to mark a point in time in your repository. We should add tag to mark a released version.

1. **Different types of git tags available? give exact commands on how to create each type of tag?**

Ans.

* + Annotated tags – git tag –a v1.1 – m “my version 1.1”
  + Light weight Tags – git tag v1.1-1w
  + Shared Tags – git push origin v1.1
  + Checking out Tags – git checkout –b version v1.1.1

1. **What command do you use to merge two branches ? give exact commands. How do you know what files need conflict resolution? How do you proceed after resolving a conflict?**

**Ans.**

* + **git checkout master**
  + **git merge newTest**

One after running above two commands, you can see a message in the end as **CONFLICT:** xyz. You can resolve it manually. By making changes in same file. After resolving the conflict you need to follow bellow steps.

* **git add filepath**
* **git commit –m “Conflict resolved”**

1. **How do you know current state of your workspace to that on git server ?**

**Ans:**

We can use commands like git reflog to check updates in the repository. You can save current status using git stash command.

1. **What is command used to fetch the latest updates in others?**

**Ans.** git pull is used to fetch. git statue is used to check status..

1. **What is the command is used to create a git repo for your personal development?**

**Ans.**

**git init** command creates a new git repository. git init <directory>

1. **Explain the relationship between the working directory, the index and repository?**

**Ans.**

Git index is where our place files that are committed to the git repository is stored. Working directory is where we do our works like add, remove and modify etc. Files in the git index are files that git would commit to the git repository if we use the command **git commit**. You can see the git index of files by

**git ls-files**

1. **what command do you use to know list of previous commits ?**

**Ans: git log**

1. **What is 3a525393f6a5c47fa08d91ef16c16927ed3cd33a? What are the benefits of this?**

**Ans.** It is a SHA1 hashs. git commits are stored in terms of hashs. By locking into hash values using commands like git log ---petty = format: “%h %s” –graph, we can go through branching and merging.

1. **command to rename a file in Git?**

**Ans.** git mv oldfile newfilename

1. **What git command is used to undo changes made to your local repo? What are the variations and how do they work?**

**Ans git revert <commit>**

**25) What is the command to use**

**1) to delete a local git branch?**

**Ans.** git branch –d localbranch

**2) to delete git branch on remote server**

Ans. git push origin :the\_remote\_branch