**SUSAN BAILEY HOMEWORK WEEK 4**

It’s worth stating that Git is different to Git Hub.

Git is a version control tool is provides a history of content changes and facilitates collaborative working by recording changes to files.

Git Hub is a platform to access GitHub, but also to link many other accounts ( such as Code wars, VS Code ) and store all your coding repositories in one place.

**QUESTION 1**

GITHUB WORKFLOW FUNDAMENTALS

Any Git project will consist of three sections: the Git directory, the working tree, and the staging area. The Git directory contains the history of all the files and changes.

**Working Directory**

The Working Directory is the current local directory that I am working on and is typically the folder that is my .git folder. When I make changes to any files I will save them locally. It’s where I can try out changes before committing them to my staging area.

**Staging Area**

This area is where I save files that will be part of my next commit. I can add files one at a time, and when I am ready to commit, GitHub can bundle them together to the local repo.

**Local Repo ( head )**

The local repo head is a pointer to the current branch reference, which in turn is also the last commit made that was checked out into my working directory. This also means it is the parent of my next commit. More simply, it’s the latest ‘snap shot’ of my last commit.

**Remote repo ( master / main )**

This is the main remote file,

WORKING DIRECTORY STATES

**Staged**

This means I have a modified file that is ready to be committed.

**Modified**

This means I have changed the file but haven’t committed it to my database yet. It could be that i am still woking on a file and I am not ready to pass it on yet.

**Committed**

This means the data is stored in my local database.

GIT COMMANDS

**Git Add**

$ git add <file>

This adds files to the staging area.

**Git Commit**

$ git commit

A commit is s type of snapshot of your file, a work-in-progress. You can decide went do this by making a ‘commit’. You can access this ‘snapshot’ at any time.

This will take everything that’s in your index or staging area and save it in the local repository.

**Git Push**

$ git push

This command uploads all local branch commits that you have created and push them to a remote repository such as GitHub.

**Git Fetch**

$ git fetch

This downloads all history from the remote tracking branches.

**Git Merge**

$ git merge

Merging will combine forked history. The ‘git merge’ command lets you take individual lines of development created by git branch and and bring them together in a single branch.

**Git Pull**

$ git pull

This updates your current local working branch with new commits from the corresponding remote branch in GitHub. ‘git pull’ is a combination of ‘git fetch’ and ‘git merge’.

If someone has made a change to a file in the remote repository and you want to see the latest changes, you action a ‘git pull’.

**QUESTION 2**

EXCEPTION HANDLING

**Question 1**

**Simple ATM program**

print(**'Welcome to Bailey Bank\n'**)

balance = 100

pin\_code = 0000

tries = 0

cust\_no = int(input(**"Please insert your card and enter your 4 number pin code\n"**))

**try**:

**if** cust\_no == pin\_code:

withdraw = int(input(**'Thank you please enter your withdrawal amount\n'**))

**if** withdraw <= balance:

print(**'You have withdrawn £{} please wait for your cash'**.format(withdraw))

**else**:

print(**'Value exceeds your balance, please contact your bank.'**)

**except** ValueError:

print(**'Sorry, please enter 4 numbers'**)

**except** Exception **as** e:

print(**'Sorry. Something went wrong {}'**.format(e))

**elif** cust\_no != pin\_code:

**while** tries <= 1:

print(**'Incorrect pin. Please try again.'**)

cust\_no = int(input(**"Please insert your card and enter your 4 number pin code\n"**))

tries += 1

**else**:

print(**'Too many pin attempts. Please contact your bank.'**)

**except** ValueError:

print(**'Sorry, please enter 4 numbers'**)

**except** Exception **as** e:

print(**'Sorry. Something went wrong {}'**.format(e))