**Practice Quiz: Before Version Control**

**TOTAL POINTS 5**

1.

Question 1

Your colleague sent you a patch called fix\_names.patch, which fixes a config file called fix\_names.conf. What command do you need to run to apply the patch to the config file?

**1 / 1 point**



diff names.conf fix\_names.conf



patch fix\_names.conf names.conf



patch fix\_names.conf < fix\_names.patch



diff names.conf\_orig names.conf\_fixed > fix\_names.conf

**Correct**

Nice job! The patch command with the file to be patched, followed by the filename of the patch, will apply it.

2.

Question 2

You're helping a friend with a bug in a script called fix\_permissions.py, which fixes the permissions of a bunch of files. To work on the file, you make a copy and call it fix\_permissions\_modified.py. What command do you need to run after solving the bug to send the patch to your friend?

**1 / 1 point**



diff fix\_permissions.py fix\_permissions\_modified.py > fix\_permissions.patch



patch fix\_permissions.py < fix\_permissions\_modified.py



patch fix\_permissions.py > fix\_permissions.patch



diff fix\_permissions.py fix\_permissions.diff

**Correct**

Awesome! The diff command will allow us to compare and apply the differences between the files.

3.

Question 3

The \_\_\_\_\_ commandhighlights the words that changed in a file instead of working line by line.

**1 / 1 point**



diff



diff -u



vimdiff



patch

**Correct**

Right on! The vimdiff commandhighlights the words that changed in a file by color, in addition to working line by line.

4.

Question 4

How can we choose the return value our script returns when it finishes?

**1 / 1 point**



Using the exit command from the sys module



Use the patch command



Use the diff command



Use meld

**Correct**

Great work! A script can use sys.exit to finish processing and return the number passed as an argument as the script's return code.

5.

Question 5

In addition to the original files, what else do we need before we can use the patch command?

**1 / 1 point**



Diff file



exit command of the sys module



Version control



Full copy of the new files

**Correct**

Woohoo! We need to use the patch command with the diff file to apply new changes to the original file.

**Diff and Patch Cheat Sheet**

### **diff**

diff is used to find differences between two files. On its own, it’s a bit hard to use; instead, use it with diff -u to find lines which differ in two files:

### **diff -u**

diff -u is used to compare two files, line by line, and have the differing lines compared side-by-side in the same output. See below:

~$ cat menu1.txt

Menu1:

Apples

Bananas

Oranges

Pears

~$ cat menu2.txt

Menu:

Apples

Bananas

Grapes

Strawberries

~$ diff -u menu1.txt menu2.txt

--- menu1.txt 2019-12-16 18:46:13.794879924 +0900

+++ menu2.txt 2019-12-16 18:46:42.090995670 +0900

@@ -1,6 +1,6 @@

-Menu1:

+Menu:

Apples

Bananas

-Oranges

-Pears

+Grapes

+Strawberries

### **Patch**

Patch is useful for applying file differences. See the below example, which compares two files. The comparison is saved as a .diff file, which is then patched to the original file!

~$ cat hello\_world.txt

Hello World

~$ cat hello\_world\_long.txt

Hello World

It's a wonderful day!

~$ diff -u hello\_world.txt hello\_world\_long.txt

--- hello\_world.txt 2019-12-16 19:24:12.556102821 +0900

+++ hello\_world\_long.txt 2019-12-16 19:24:38.944207773 +0900

@@ -1 +1,3 @@

Hello World

+

+It's a wonderful day!

~$ diff -u hello\_world.txt hello\_world\_long.txt > hello\_world.diff

~$ patch < hello\_world.diff

patching file hello\_world.txt

~$ cat hello\_world.txt

Hello World

It's a wonderful day!

There are some other interesting patch and diff commands such as patch -p1, diff -r !

Check them out in the following references:

* <http://man7.org/linux/man-pages/man1/diff.1.html>
* <http://man7.org/linux/man-pages/man1/patch.1.html>

## **More Information About Git**

Check out the following links for more information:

* <https://git-scm.com/doc>
* <https://www.mercurial-scm.org/>
* <https://subversion.apache.org/>
* <https://en.wikipedia.org/wiki/Version_control>

**Practice Quiz: Version Control Systems**

**TOTAL POINTS 5**

1.

Question 1

How can a VCS (Version Control System) come in handy when updating your software, even if you’re a solo programmer? Check all that apply.

**1 / 1 point**



Git retains local copies of repositories, resulting in fast operations.

**Correct**

Awesome! Git's distributed architecture means each person contributing to a repository retains a full copy of the repository locally.



If something breaks due to a change, you can fix the problem by reverting to a working version before the change.

**Correct**

Nice job! With version control, if something goes wrong, we can fix it immediately and figure out what happened later.



Git relies on a centralized server.



Git allows you to review the history of your project.

**Correct**

Right on!

2.

Question 2

Who is the original creator and main developer of the VCS (Version Control System) tool Git?

**1 / 1 point**



Bill Gates



Guido van Rossum



Linus Torvalds



James Gosling

**Correct**

Nailed it! Linus Torvalds developed Git in 2005 to better facilitate the process of developing the Linux kernel with developers across the globe.

3.

Question 3

\_\_\_\_\_ is a feature of a software management system that records changes to a file or set of files over time so that you can recall specific versions later.

**1 / 1 point**



A repository



sys.exit()



Version control



IDE

**Correct**

Right on! A version control system keeps track of the changes that we make to our files.

4.

Question 4

A \_\_\_\_\_ is a collection of edits which has been submitted to the version control system for safe keeping.

**1 / 1 point**



IDE



version control system



commit



repository

**Correct**

Awesome! We call the collection of edits we are making at one time a commit.

5.

Question 5

Within a VCS, project files are organized in centralized locations called \_\_\_\_\_ where they can be called upon later.

**1 / 1 point**



commits



repositories



IDE



yum

**Correct**

Excellent. A repository is a central location in which data is stored and managed.