

Quiz: Git (Practice Problems)

1 Basic Problems

Problem 1. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ touch README
$ mkdir foo
$ touch foo/README2
$ cd foo
$ touch README3
$ ls
```

Note: The first four lines of every problem will always be the same. Variations of this problem include: (1) removing the `git init` command, (2) removing or adding `touch` / `mkdir` / `cd` commands, (3) changing the referenced filenames, and (4) adding the `-a` flag to the final `ls` command.

Problem 2. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ echo "hello_world" > README_EN
$ echo "hola_mundo" > README_ES
$ mkdir foo
$ echo "hello_again" > foo/README_EN
$ echo "hola_otra_vez" >> README_ES
$ ls -a
```

Note: Variations of this problem include all the variations above, plus: (1) changing the final command to `cat README`, (2) changing the name of any referenced file, (3) adding/removing `echo` commands, and (4) changing the `>` to `>>` (or vice versa).

Problem 3. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ echo "print('hello_world')" > foo.py
$ echo "print('hola_mundo')" >> bar.py
$ echo 'print("hola_otra_vez")' >> bar.py
$ echo "print(\"hello again\")" >> foo.py
$ python3 foo.py
```

Note: Variations of this problem include all the variations above, plus: (1) changing the `python3` command to a `cat` or `ls` command, and (2) changing the type of quotation mark used at any location. Any python code provided will only contain simple print statements.

Problem 4. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ echo "print('hello_world')" > foo.py
$ git add foo.py
$ git commit -m "added_foo"
$ git branch foo
$ git checkout foo
$ echo "print('hola_mundo')" >> foo.py
$ git add foo.py
$ git commit -m "modified_foo"
$ git checkout master
$ python3 foo.py
```

Note: Variations of this problem include all the variations above, plus: adding and removing the commands `git add`, `git commit`, `git branch`, or `git checkout` at arbitrary locations.

2 More Variations of the Basic Problems

Problem 5. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ touch .README
$ mkdir .foo
$ touch .foo/README2
$ touch README3
$ ls .foo
```

Problem 6. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ mkdir .foo
$ echo "hello_world" >> .foo/README
$ echo "hello_again" > .foo/README
$ cat .foo/README
```

Problem 7. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ mkdir foo
$ echo "print('hello_world')" >> foo.py
$ echo "print('hola_mundo')" >> foo/foo.py
$ echo 'print("hola_otra_vez")' >> foo/bar.py
$ echo "print(\"hello again\")" >> foo/.foo.py
$ python3 foo/foo.py
```

Problem 8. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ echo "print('hello_world')" > foo.py
$ echo "print('hola_mundo')" > bar.py
$ git add foo.py
$ git commit -m "first_commit"
$ git branch foo
$ git checkout foo
$ echo "print('hello_again')" >> foo.py
$ git add foo.py
$ git add bar.py
$ git commit -m "second_commit"
$ git checkout master
$ echo "print('hola_otra_vez')" >> bar.py
$ python3 bar.py
```

Problem 9. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ touch .README
$ mkdir .foo
$ touch .foo/README2
$ touch README3
$ ls .foo
```

Problem 10. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ mkdir .foo
$ echo "hello_world" >> .foo/README
$ echo "hello_again" > .foo/README
$ cat .foo/README
```

Problem 11. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ mkdir foo
$ echo "print('hello_world')" >> foo.py
$ echo "print('hola_mundo')" >> foo/foo.py
$ echo 'print("hola_otra_vez")' >> foo/bar.py
$ echo "print(\"hello again\")" >> foo/.foo.py
$ python3 foo/foo.py
```

Problem 12. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ echo "print('hello_world')" > foo.py
$ echo "print('hola_mundo')" > bar.py
$ git add foo.py
$ git commit -m "first_commit"
$ git branch foo
$ git checkout foo
$ echo "print('hello_again')" >> foo.py
$ git add foo.py
$ git add bar.py
$ git commit -m "second_commit"
$ git checkout master
$ echo "print('hola_otra_vez')" >> bar.py
$ python3 bar.py
```

3 Fun with git and glob

Problem 13. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ touch hello world
$ touch .salve .munde
$ git add *e*
$ git commit -m 'first commit'
$ git checkout -b foo
$ git add *
$ git commit -m 'second commit'
$ git checkout master
$ git checkout -b bar
$ git add .
$ git commit -m 'third commit'
$ git checkout master
$ ls -a
```

Problem 14. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ touch hello world
$ touch .salve .munde
$ git add .
$ git commit -m 'first commit'
$ git checkout -b foo
$ touch '*'
$ git add *
$ git commit -m 'second commit'
$ git checkout master
$ git checkout -b bar
$ echo "help_me" > test
$ git add *
$ git commit -m 'third commit'
$ git checkout foo
$ ls -a
```

Problem 15. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ mkdir test
$ touch test/hello world
$ touch test/.salve .munde
$ cd test
$ git add .*
$ git commit -m 'first commit'
$ git checkout -b foo
$ git add .
$ git commit -m 'second commit'
$ git checkout master
$ ls -a
```

Problem 16. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ mkdir test
$ touch hola mundo
$ touch test/'hello world'
$ touch test/' .salve .munde'
$ cd test
$ for i in *; do git add $i; done
$ git commit -m 'first commit'
$ git checkout -b foo
$ git add .
$ git commit -m 'second commit'
$ ls -a
```

Problem 17. Write the output of the final command in the following terminal session. If the command has no output, then leave the problem blank.

```
$ cd; rm -rf quiz; mkdir quiz; cd quiz
$ git init
$ echo evil > -a
$ touch hola mundo
$ touch test/'hello world'
$ touch test/' .salve .munde'
$ cd test
$ git add .
$ git commit -m 'first commit'
$ git checkout -b foo
$ git add ..
$ git commit -m 'second commit'
$ cd $HOME/quiz
$ git checkout master
$ ls *
```