Lab 5: Version Control

Team Members:
1. Dhrumil Amish Shah

Date:

2021-02-11

—

Subject:

Software Development Concepts

—

Professor:

Matthew Amy

Part 1 – History

1. How many commits are there?
There is a total of **9** commits. (Using "git log --oneline" or "git shortlog")

2. What is the commit message associated with the most recent commit?
The commit message associated with the most recent commit is **"Whoops, forgot to ignore class files"**. (Using "git log --oneline" and the first commit is the most recent)

3. Which commit added the basePrices static variable?
Commit with commit id **"b590e9d"** (i.e., b590e9d454004f89a1af65c09295ea4c0f896171) and commit message **"Factored prices into an array"** added the basePrices static variable.
(Using "git diff b590e9d~ b590e9d")

4. How many branches are there?
- There are two local branches, **"coupons"** and **"master"**. (Using "git branch -l")
- There is one remote branch, **"origin/master"**. (Using "git branch -r")
- Total, there are three branches. (Using "git branch -a")

5. Which is the last commit on the coupons branch?
The last commit on the **"coupons"** branch has commit id **"67ba33e"** (i.e., 67ba33edfb465babbbae8d1cbe43cb4c78583cfe) and commit message **"Added coupon feature"**.

Part 2 – Basic operations

1. Fix the 3 remaining bugs from the Debugging lab. In particular:
- On lines 62, 64, and 66, topping == str should be topping.equals(str) - (Done)
- On lines 63, 65, and 67, price = should be price += - (Done)
- The final price should be price = price - price*coupon/100 - (Done)

2. Commit these bug fixes with a suitable commit message
- Add file. (Using git add HfxDonairExpress.java) (add)
- Commit file. (Using git commit -m "Corrected string comparison and price calculation logic") (commit)
- Commit id **073737d** (i.e., 073737d196b255905b63b558cf1dd151d2cc77c7) created

```
C:\Windows\System32\cmd.exe                                                              —   □   ×

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   HfxDonairExpress.java

no changes added to commit (use "git add" and/or "git commit -a")

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git add HfxDonairExpress.java
warning: LF will be replaced by CRLF in HfxDonairExpress.java.
The file will have its original line endings in your working directory

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git status
On branch master
Your branch is up to date with 'origin/master'.

Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   HfxDonairExpress.java

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git commit -m "Corrected string comparison and price calculation logic"
[master 073737d] Corrected string comparison and price calculation logic
 1 file changed, 7 insertions(+), 7 deletions(-)

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git log --oneline
073737d (HEAD -> master) Corrected string comparison and price calculation logic
12e89aa (origin/master) Whoops, forgot to ignore class files
7d9c8c6 Modified coupon calculation
0e58596 Merge branch 'coupons'
67ba33e (coupons) Added coupon feature
b97a7ab Bugfix - case statement falls through
9f2ec80 Added ability to purchase toppings
b590e9d Factored prices into an array
ae22dfe Basic functionality working
5e6d1a8 Initial commit

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>_
```

Part 3 – Branching

1. Create a new branch called burger. Your new branch should be off the repository's master branch
Using git checkout -b burger (Creates a "burger" branch from current branch i.e., master branch).

2. Add a new menu option for a burger. A burger at HfxDonairExpress costs $2.50
Added a new menu option for burger.

3. Commit this change on the burger branch
- Add file. (Using git add HfxDonairExpress.java) (add)
- Commit file. (Using git commit -m "Added a new option menu for a burger")
- Commit id **fe0124f** (i.e., fe0124f1da857339e7f0d797073bbd7e46a0c8e6) created

4. Finally, merge the burger branch into the main branch
- git checkout master
- git merge burger

```
C:\Windows\System32\cmd.exe                                                                    -  □  X
D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git status
On branch burger
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   HfxDonairExpress.java

no changes added to commit (use "git add" and/or "git commit -a")

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git add HfxDonairExpress.java
warning: LF will be replaced by CRLF in HfxDonairExpress.java.
The file will have its original line endings in your working directory

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git status
On branch burger
Changes to be committed:
  (use "git reset HEAD <file>..." to unstage)

        modified:   HfxDonairExpress.java


D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git commit -m "Added a new option menu for a burger"
[burger fe0124f] Added a new option menu for a burger
 1 file changed, 57 insertions(+), 41 deletions(-)

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git status
On branch burger
nothing to commit, working tree clean

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git checkout master
Switched to branch 'master'
Your branch is ahead of 'origin/master' by 1 commit.
  (use "git push" to publish your local commits)

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git merge burger
Updating 073737d..fe0124f
Fast-forward
 HfxDonairExpress.java | 98 +++++++++++++++++++++++++++++--------------------
 1 file changed, 57 insertions(+), 41 deletions(-)

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>git log --oneline
fe0124f (HEAD -> master, burger) Added a new option menu for a burger
073737d Corrected string comparison and price calculation logic
12e89aa (origin/master) Whoops, forgot to ignore class files
7d9c8c6 Modified coupon calculation
0e58596 Merge branch 'coupons'
67ba33e (coupons) Added coupon feature
b97a7ab Bugfix - case statement falls through
9f2ec80 Added ability to purchase toppings
b590e9d Factored prices into an array
ae22dfe Basic functionality working
5e6d1a8 Initial commit

D:\DALHOUSIE TOTAL\Study_Material\Term_1\CSCI_3901_SDC\Labs\Lab 5\HfxDonairExpress>_
```

Part 4 – Questions

1. When working alone on a project, how frequently should you commit your code to a version control system? Explain why.

- While working alone on a project, best practice is to code a bit and when we reach a stable position, commit but ensure it is complete, well tested and makes sense.

2. When working in a team, how frequently should you commit your code to a version control system? Explain why.

- Commit before signing off for the day so that changes are saved and continued later.
- Commit when there is a dependency from our end to someone.
- Commit frequently but ensure that commits are meaningful, well tested and doesn't break anyone else's stuff. (E.g., New method addition, Logic change, Feature change, etc.)

3. Why might you create branches for your project in your version control system?

- We create separate branches to have different modules, features, functionalities in their own branch so that the code in the main branch is not affected. General practice is to create a branch for every feature. For example: Suppose we are developing an application having modules like Trips, Marketplace, Profile etc., we can create a separate branch for each module and work in that branch only. After the module has developed and tested, we can merge it into the main branch for release.

- We create hotfix branches if code requires a quick fix so that we can quickly modify, test and release the code directly to production. Generally, the hotfix branch is created from the main branch.

- We create a separate branch for each developer working on the same project, so they have their own branch to work and main branch or sub-branches are not disturbed.