In-class exercise 1: Advanced uses of Git

February 25, 2021
Types of VCS -- Distributed

- Each developer has their own repository.
  - Created by the developer, or
  - Cloned from an existing (remote) repository
- Developers work on their own repos
  - They can commit, branch, etc.
  - Activity is local unless it is pushed to remote repo
  - Remote activity is not seen until developer fetches from the remote repo
- Examples: Mercurial (hg), git
https://github.com/
Git -- Three Main Stages

1. **Committed**: Everything in the file is currently in the database

2. **Modified**: Changed the file but have not committed to the database

3. **Staged**: Marked the file for addition to the database in the next commit

Note that all of the above pertain to *tracked* files.
Git -- Git repo abstraction
Git -- Git Commit Object

message:
```
basic functionality with sanity check done
```

parent:
```
Rico Angell committed on Oct 2, 2017
```

hash:
```
1 parent 1c82903  commit 2c2ec8a72ce21615daee3868af32b929852807a65
```

author or committer:
```
Author or committer along with date
```

initially entire file then diffs:
```
Showing 2 changed files with 108 additions and 64 deletions.
```

```python
...@@ -1,3 +1,4 @@
  import pytest
  import themis

+ def test_get_test_result():
+    t = themis Themis(xml_fname="settings.xml")
+    _p = t.group_discrimination(i_fields=["Sex", "Race"])
+    print "Sex and Race: ", p
+    _p = t.group_discrimination(i_fields=["Race"])
+    print "Race: ", p
+    _p = t.group_discrimination(i_fields=["Sex"])
+    print "Sex: ", p
```

```python
...@@ -26,23 +27,23 @@ def test_get_test_result():
  t = themis Themis(xml_fname="settings.xml")
  _p = t.group_discrimination(i_fields=["Sex", "Race", "Age", "Income"]):
+    print "\nGroup:"
+    for f in t_.all_relevant_subs(["Sex", "Race", "Age", "Income"]):
+      _p = t.group_discrimination(i_fields=f)
+      print f, "\n\n", p
```

```python
...@@ -38,3 +38,3 @@ def test_causal_discrimination():
  t = themis Themis(xml_fname="settings.xml")
  _p = t.causal_discrimination(i_fields=["Sex", "Race"])
```

```python
...@@ -40,3 +40,3 @@ def test_causal_discrimination():
  t = themis Themis(xml_fname="settings.xml")
  _p = t.causal_discrimination(i_fields=["Sex", "Race"])
+    print "\nCausal:"
```
Today’s in-class exercise

• Instructor: Heather Conboy

• Grader: Sayantan Bhowmik

NOTE) May also get help from git man pages, lecture materials, and other on-line resources

https://people.cs.umass.edu/~hconboy/class/2021Spring/CS520/in-class1.pdf
Group selection

- Form 4- or 5-person teams
  - If you need more members in your team, raise your hand and ask the instructor
- Use Moodle to self-select a team

- Select a team member responsible for submitting the completed exercise by next Wednesday
Set up

1. Check if you are using a newer version of git (version \(\geq 2.7.4\))

2. Clone the basic-stats git repository:
   git clone https://github.com/LASER-UMASS/basic-stats basic-stats

3. Create a (second) local fork by locally cloning again, this time from the first local clone:
   git clone basic-stats basic-stats-fork

NOTE) Recommend taking notes as you go for writing up the answers to the six (6) questions as well as capturing some of the git logs along the way