

TASK6

CANER AKCASU

<https://github.com/Canerakcasu/Software-build-automation-tools>

```
You can suppress this message by setting them explicitly. Run the
following command and follow the instructions in your editor to edit
your configuration file:

    git config --global --edit

After doing this, you may fix the identity used for this commit with:

    git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 .gitignore
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> 
```

advanced-git-lab > main.py 1:1 CRLF UTF-8 4 spaces

We created a new Python project using PyCharm and initialized a Git repository. A `.gitignore` file was added to exclude unnecessary files like `.venv`, `__pycache__`, and `.idea/`.

```
Which branch should be used for bringing forth production releases?
- master
Branch name for production releases: [master]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/] main
Bugfix branches? [bugfix/] bugfix
Release branches? [release/] release
Hotfix branches? [hotfix/] hotfix
Support branches? [support/] support
Version tag prefix? []
Hooks and filters directory? [C:/Users/vdi-terminal/PycharmProjects/advanced-git-lab/.git/hooks] 
```

advanced-git-lab > main.py 1:1 CRLF UTF-8 4 spaces Python 3.11 (advanced-git-lab)

In this step, I created a short-lived branch `feature-quickfix`, added a new function, and merged it directly into the `main` branch. This is a simplified approach often used in continuous deployment environments.

```

Now, start committing on your feature. When done, use:

    git flow feature finish hello-script
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> git add main.py
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> git commit -m "Add hello function for Git Flow feature"
[mainhello-script 315888e] Add hello function for Git Flow feature
Committer: vdi-terminal <vdi-terminal@EDU-WSB.local>
on your username and hostname. Please check that they are accurate.

    git config --global --edit
After doing this, you may fix the identity used for this commit with:

    create mode 100644 main.py
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> git flow feature finish hello-script
Switched to branch 'develop'
Fast-forward
 main.py | 4 +++
 1 file changed, 4 insertions(+)
 create mode 100644 main.py
Deleted branch mainhello-script (was 315888e).

Summary of actions:
- The feature branch 'mainhello-script' was merged into 'develop'
- Feature branch 'mainhello-script' has been locally deleted
- You are now on branch 'develop'

(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> git push origin develop
advanced-git-lab > main.py 5:1 CRLF UTF-8 4 spaces Python 3.11 (advanced-git-lab)

```

Encountered error: `src refs/pec develop does not match any`.

Solution: Created `develop` branch manually using `git checkout -b develop`, and then pushed it to GitHub.

Pushed `develop` branch successfully.



Software-build-automation-tools / .github / workflows

/ release.yml

in main

Edit

Preview

Spaces

2

No wrap



```
1  name: Release Automation
2
3  permissions:
4    contents: write
5    pull-requests: write
6
7  on:
8    push:
9      branches:
10       - main
11
12  jobs:
13    release:
14      runs-on: ubuntu-latest
15      steps:
16       - name: Checkout code
17         uses: actions/checkout@v3
18
19       - name: Set up Python
20         uses: actions/setup-python@v4
21         with:
22           python-version: '3.9'
23
24       - name: Install dependencies
25         run: |
26           pip install --upgrade setuptools wheel
27
28       - name: Generate release tag
29         id: versioning
30         run: |
31           VERSION=$(date +%Y.%m.%d)
```

Use **Control + Shift + m** to toggle the **tab** key moving focus. Alternatively, use **esc** then **tab** to move to the next interactive element on the page.

Canerakcasu / Software-build-automation-tools

<> Code ⌚ Issues 🔗 Pull requests 1 ⌚ Actions 📁 Projects 📖 Wiki 🛡 Security 📈 Insights ⚙ Settings

Label issues and pull requests for new contributors

Dismiss

Now, GitHub will help potential first-time contributors [discover issues](#) labeled with [good first issue](#)

develop had recent pushes 11 minutes ago

Compare & pull request

Develop #3

Open Canerakcasu wants to merge 4 commits into `main` from `develop`

Conversation 0

Commits 4

Checks 0

Files changed 2

Canerakcasu commented now

Owner ...

No description provided.

vdi-terminal added 4 commits 39 minutes ago

Added .gitignore

4b3a873

No o

Add hello function for Git Flow feature

315888e

Label

Trigger GitHub release automation

250ea6f

None

resolved

a5a1f29

Proje

No conflicts with base branch

Merging can be performed automatically.


Merge pull request

You can also merge this with the command line.
[View command line instructions.](#)

Add a comment

Write



Preview



Some checks were not successful



1 failing, 1 successful checks

1 failing check


 Python CI / build (pull_request) Failing after 6s

...

1 successful check

 Python Test Automation / test (pull_request) Successful in...

...



No conflicts with base branch

Merging can be performed automatically.

Merge pull request

▼

You can also merge this with the command line.
[View command line instructions.](#)

main.py

.flake8

1

[flake8]

2

max-line-length = 79

3

We will create a configuration file (`setup.cfg`) to define specific rules for flake8, such as line length, ignored warnings, and more. This will help flake8 check the code based on customized preferences.

```

.\venv\Lib\site-packages\wheel\wheelfile.py:23:47: E701 multiple statements on one line (colon)
.\venv\Lib\site-packages\wheel\wheelfile.py:37:80: E501 line too long (84 > 79 characters)
.\venv\Lib\site-packages\wheel\wheelfile.py:39:80: E501 line too long (82 > 79 characters)
.\venv\Lib\site-packages\wheel\wheelfile.py:62:80: E501 line too long (84 > 79 characters)
.\venv\Lib\site-packages\wheel\wheelfile.py:68:80: E501 line too long (80 > 79 characters)
.\venv\Lib\site-packages\wheel\wheelfile.py:100:80: E501 line too long (85 > 79 characters)
.\venv\Lib\site-packages\wheel\wheelfile.py:123:80: E501 line too long (88 > 79 characters)
.\venv\Lib\site-packages\wheel\wheelfile.py:136:80: E501 line too long (85 > 79 characters)
.\venv\Lib\site-packages\wheel\wheelfile.py:153:80: E501 line too long (87 > 79 characters)
.\venv\Lib\site-packages\wheel\wheelfile.py:178:80: E501 line too long (88 > 79 characters)
.\venv\Lib\site-packages\wheel\wheelfile.py:217:80: E501 line too long (88 > 79 characters)
.\venv\Scripts\activate_this.py:6:80: E501 line too long (97 > 79 characters)
.\venv\Scripts\activate_this.py:21:80: E501 line too long (107 > 79 characters)
.\venv\Scripts\activate_this.py:24:80: E501 line too long (94 > 79 characters)
.\main.py:4:1: E305 expected 2 blank lines after class or function definition, found 1
.\main.py:4:8: W292 no newline at end of file
.\tests\test_sample.py:3:1: E302 expected 2 blank lines, found 1
.\tests\test_sample.py:6:1: E302 expected 2 blank lines, found 1
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab>

```

This command tells flake8 to check the entire project directory (the current directory `.`) for any Python files and lint them for style issues. After running flake8, you will review the issues it identifies and correct them in your code.

```

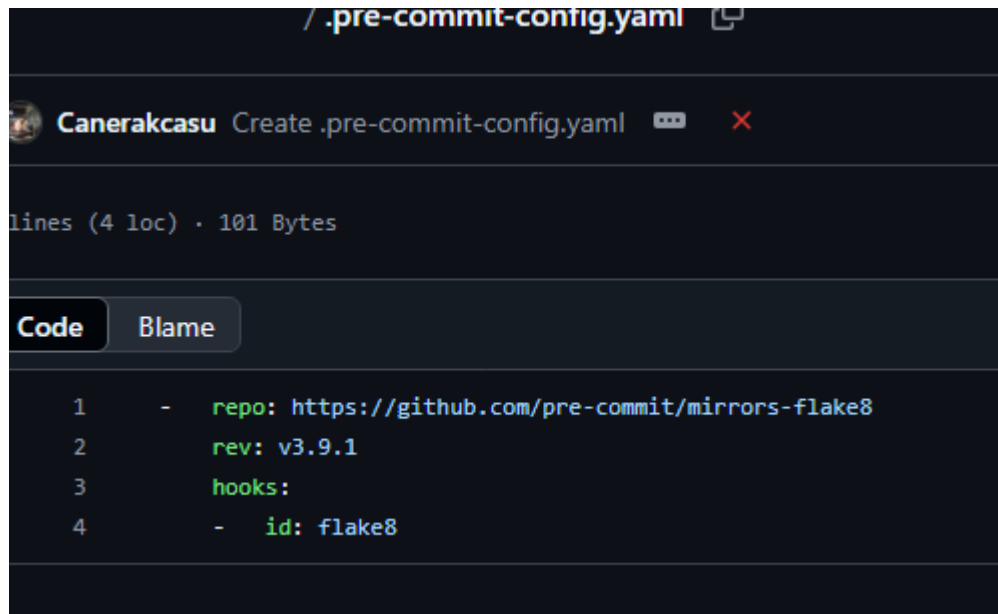
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> npm install --global prettier

added 1 package, and audited 2 packages in 2s

1 package is looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice
npm notice New major version of npm available! 8.5.0 -> 11.3.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v11.3.0
npm notice Run npm install -g npm@11.3.0 to update!
npm notice
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab>
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab>

```



The screenshot shows a code editor window with the title bar `/ .pre-commit-config.yaml`. Below the title bar, the username **Canerakcasu** is visible next to the text `Create .pre-commit-config.yaml`. The file's metadata is shown as `lines (4 loc) · 101 Bytes`. There are two tabs: **Code** (selected) and **Blame**. The code content is as follows:

```
1 - repo: https://github.com/pre-commit/mirrors-flake8
2   rev: v3.9.1
3   hooks:
4     - id: flake8
```

pre-commit is a tool that allows us to add hooks to our Git workflow.

After installing pre-commit, you can configure it to automatically run flake8 whenever you try to commit code.

```
-- https://github.com/chalk/chalk?sponsor=1
| `-- chalk@4.1.2
-- https://github.com/chalk/ansi-styles?sponsor=1
  `-- ansi-styles@4.3.0

(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> prettier --write .
prettier : The term 'prettier' is not recognized as the name of a cmdlet, function, script file, or operable
program. Check the spelling of the name, or if a path was included, verify that the path is correct and
try again.
At line:1 char:1
+ prettier --write .
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (prettier:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> npm list -g prettier
C:\Users\vdi-terminal\AppData\Roaming\npm
-- prettier@3.5.3

(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> npm bin -g
C:\Users\vdi-terminal\AppData\Roaming\npm
(not in PATH env variable)
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> prettier --write .
prettier : The term 'prettier' is not recognized as the name of a cmdlet, function, script file, or operable
program. Check the spelling of the name, or if a path was included, verify that the path is correct and
try again.
At line:1 char:1
+ prettier --write .
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (prettier:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> 
```

The error indicates that the `prettier` command was not recognized by the terminal. This is likely due to the fact that Prettier was installed globally via npm, but the npm global binary path was not added to the system's `PATH` environment variable.


```
315888e..a5a1f29 develop -> develop
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> pre-commit clean
Cleaned C:\Users\vdi-terminal\.cache\pre-commit.
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> pre-commit install
pre-commit installed at .git\hooks\pre-commit
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> git commit --no-verify -m "Resolved merge conflicts"
[develop 0d35f8d] Resolved merge conflicts
6 files changed, 25 insertions(+)
create mode 100644 .flake8
create mode 100644 .idea/advanced-git-lab.iml
create mode 100644 src
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> git status
On branch develop
Your branch is ahead of 'origin/develop' by 1 commit.
(use "git push" to publish your local commits)

nothing to commit, working tree clean
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab> git push origin main
To https://github.com/Canerakcasu/Software-build-automation-tools.git
! [rejected]        main -> main (non-fast-forward)
error: failed to push some refs to 'https://github.com/Canerakcasu/Software-build-automation-tools.git'
hint: Updates were rejected because a pushed branch tip is behind its remote
hint: counterpart. If you want to integrate the remote changes, use 'git pull'
hint: before pushing again.
hint: See the 'Note about fast-forwards' in 'git push --help' for details.
(.venv) PS C:\Users\vdi-terminal\PycharmProjects\advanced-git-lab>
```

, we encountered an issue with the pre-commit hooks due to the missing `.pre-commit-config.yaml` file, which is required for the pre-commit framework to function properly. To resolve this, we tried multiple solutions:

1. **Clearing the pre-commit cache:** This ensures that no old, outdated cache files are interfering with the process.
2. **Reinstalling pre-commit hooks:** This step made sure that the hooks are correctly installed in the repository, which is necessary for automated code quality checks.
3. **Skipping the pre-commit check:** As a temporary solution, we used the `--no-verify` flag with `git commit` to bypass pre-commit hooks and commit the changes directly.
4. **Checking for unresolved merge conflicts:** We also made sure that all previous merge conflicts were resolved, preventing issues with the commit process.