1. Repository Initialization
- Created a new Git repository for the project.
- Set up the `main` branch as your production-ready baseline.

```
[malvikabhadoriya@Malvikas-MacBook-Air Desktop % cd DevOps_Practice
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % code .
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git add main.py
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git commit -m "Initial commit on main: base greeting program"
[main c18e1e3] Initial commit on main: base greeting program
 1 file changed, 5 insertions(+)
 create mode 100644 main.py
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git branch -M main
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git checkout -b develop
Switched to a new branch 'develop'
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git remote add origin https://github.com/Malvericus/DevOps_Practice.git
error: remote origin already exists.
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 389 bytes | 389.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Malvericus/DevOps_Practice.git
   0bb44c2..c18e1e3  main -> main
branch 'main' set up to track 'origin/main'.
```

2. Created the `develop` Branch
- Branched off `main` to start the integration branch (`develop`), where all feature work is accumulated before going into production.
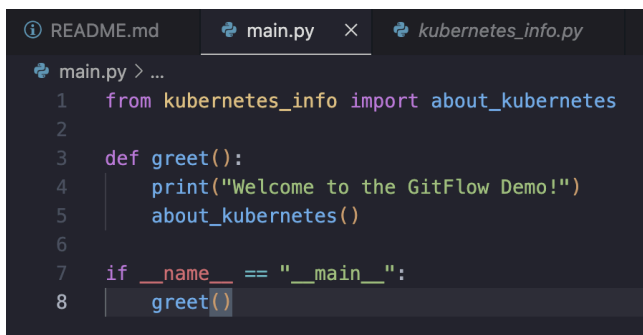
3. Developed a New Feature
- Created the branch `feature/add-kubernetes-support` from `develop`.
- Added a new module/file (`kubernetes_info.py`) with the function `about_kubernetes()`.

```
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git checkout develop
M       README.md
Switched to branch 'develop'
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git checkout -b feature/add-kubernetes-support
Switched to a new branch 'feature/add-kubernetes-support'
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % touch kubernetes_info.py
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git add kubernetes_info.py
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git commit -m "Add about_kubernetes function"
[feature/add-kubernetes-support 4002672] Add about_kubernetes function
 1 file changed, 2 insertions(+)
 create mode 100644 kubernetes_info.py
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git push -u origin feature/add-kubernetes-support
Enumerating objects: 4, done.
```

4. Integrated the Feature
- Modified `main.py` to import and call the new Kubernetes function, ensuring the feature is usable and visible.
- Committed the changes on the feature branch.

```python
from kubernetes_info import about_kubernetes


def greet():
    print("Welcome to the GitFlow Demo!")
    about_kubernetes()


if __name__ == "__main__":
    greet()
```

5. Merged Feature Branch to Develop
- Merged `feature/add-kubernetes-support` into `develop`, following best practice for feature integration.
- Resolved (if needed) any conflicts and pushed the updated `develop` branch.
6. Tested the Integration
- On your `develop` branch, ran the code to confirm the new function works as intended.

```
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git checkout develop
Switched to branch 'develop'
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git pull origin develop
From https://github.com/Malvericus/DevOps_Practice
 * branch            develop    -> FETCH_HEAD
Already up to date.
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git merge feature/add-kubernetes-support
Updating c18e1e3..4eefaa9
Fast-forward
 README.md           | 14 +++++++++++++-
 kubernetes_info.py  |  2 ++
 main.py             |  5 ++++-
 3 files changed, 19 insertions(+), 2 deletions(-)
 create mode 100644 kubernetes_info.py
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % git push origin develop
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 399 bytes | 399.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/Malvericus/DevOps_Practice.git
   c18e1e3..4eefaa9  develop -> develop
[malvikabhadoriya@Malvikas-MacBook-Air DevOps_Practice % python3 main.py
Welcome to the GitFlow Demo!
Kubernetes manages containerized applications at large scales.
```

https://github.com/Malvericus/DevOps_Practice/tree/main