Github Project: Git Commands Documentation Template

Programming for Data Science Nanodegree Program

You will use this template to copy and paste the git commands you used to complete all tasks on your local and remote git repository for this project. This file will serve as your submission for the GitHub project.

Instructions:

1. Make a copy of this Git Commands Documentation template on your Google Drive.
2. Complete the four sections in this document with the appropriate git commands.
3. Download this document as a PDF file.
4. Submit this on the Project Submission page within the Udacity Classroom.

1. Set Up Your Repository

The following are the steps you will take to create your git repository, add your python code, and post your files on GitHub.

Step 1. Create a GitHub profile (if you don't already have one).
Step 2. Fork a repository from Udacity's GitHub Project repository and provide a link to your forked GitHub repository here:

GitHub Repository Link
https://github.com/Nawaf-Khalid-Alyousef/pdsnd_github

Step 3. Complete the tasks outlined in the table below and copy and paste your git commands into the "Git Commands" column. The first git command is partially filled out for you.

	Tasks	Git Commands
A.	Clone the GitHub repository to your local repository.	$git clone https://github.com/Nawaf-Khalid-Alyousef/pdsnd_github
B.	Move your bikeshare.py and data files into your local repository.	**No git command needed (you can use cp or a GUI)**
C.	Create a .gitignore file containing the name of your data file.	**No git command needed (you can use touch or a GUI)**
D.	List the file names associated with the data files you added to your .gitignore	**No git command needed (add the file names into your .gitignore file)**
E.	Check the status of your files to make sure your files are not being tracked	`$ git status`
F.	Stage your changes.	`$ git add .`

G.	Commit your changes with a descriptive message.	$git commit -m "docs:adding bikeshare.py file and update data"
H.	Push your commit to your remote repository.	$ git push

2. Improve Documentation

Now you will be working in your local repository, on the BikeShare python file and the README.md file. You should repeat steps C through E three times to make at least three commits as you work on your documentation improvements.

	Tasks	Git Commands
A.	Create a branch named *documentation* on your local repository.	$ git branch documentation
B.	Switch to the *documentation* branch.	$ git checkout documentation
C.	Update your README.md file.	**No git command needed (edit the text in your README.md file)**
D.	Stage your changes.	$ git add .

E.	Commit your work with a descriptive message.	$ git commit -m "docs:I was update the read me file by adding the date and title statement with description "
		$ git commit -m "docs: 2 update the read me file by adding the Files used"
		$ git commit -m "docs: 3 updates the read me file by adding new title"
		$ git commit -m "docs: 4 updates the read me file by adding a new date"
		$ git commit -m "docs: 5 updates the read me file by adding a new date, Credits, Files used"

F.	Push your commit to your remote repository branch.	$ git push origin documentation

G.	Switch back to the master branch.	$ git checkout master

3. Additional Changes to Documentation

In a real world situation, you or other members of your team would likely be making other changes to documentation on the documentation branch. To simulate this follow the tasks below.

	Tasks	Git Commands
A.	Switch to the *documentation* branch.	$ git checkout documentation

B.	Make at least 2 additional changes to the documentation - this might be additional changes to the README or changes to the document strings and line comments of the bikeshare file.	(base)
		MacBook-Pro-alkhas-b-Nawaf:pdsnd_github nawaf$ git status
		On branch documentation
		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: README.md
		
		no changes added to commit (use "git add" and/or "git commit -a")
		(base)
		MacBook-Pro-alkhas-b-Nawaf:pdsnd_github nawaf$ git diff
		diff --git a/README.md b/README.md
		index f8dba84..7d4e256 100644
		--- a/README.md
		+++ b/README.md

@@ -15,3 +15,7 @@ i was used the jupyter nootbook, with data file it was chicago,new_york_city,was
Credits
 will unfortunately until now there is. no credits

+## Conclusion
+The project was very useful and I learned a lot from it about how data analysis can help us make the right decisions
+
+

base)
MacBook-Pro-alkhas-b-Nawaf:pdsnd_github nawaf$ git diff
diff --git a/README.md b/README.md
index 7d4e256..d5ebf3e 100644
--- a/README.md
+++ b/README.md
@@ -1,21 +1,26 @@
@@ -1,21 +1,26 @@
 >**Note**: Please **fork** the current Udacity repository so that you will have a **remote** repository in **your** Github account. Clone the remote repository to your local machine. Later, as a part of the project "Post your Work on Github", you will push your proposed changes to the remote repository in your Github account.

 ### Date created
-The project was created by Nawaf on moday 2022
+The project was created by Nawaf on moday 12/2022

 ### Project Title
-Exploring data to take action
+Explore US Bikeshare Data

 ### Description
-Exploring data by python lang
+In this project, you will make use of Python to explore data related to bike share systems for three major cities in the United States—Chicago, New York City, and Washington. You will write code to import the data and answer interesting questions about it by computing descriptive statistics. You will also write a script that takes in raw input to

create an interactive experience in the terminal to present these statistics.

Files used
i was used the jupyter nootbook, with data file it was chicago,new_york_city,washington, csv

Credits
-will unfortunately until now there is. no credits
+will unfortunately until now there is. no credits.
+
+## Useful website:
+https://stackoverflow.com/
+https://www.python.org/shell/

Conclusion
-The project was very useful and I learned a lot from it about how data analysis can help us make the right decisions
+The project was very useful and I learned a lot from it about how data analysis can help us make the right decisions.

ESCOD
>**Note**: Please **fork** the current Udacity repository so that you will have a **remote** repository in **your** Github account. Clone the remote repository to your local machine. Later, as a part of the project "Post your Work on Github", you will push your proposed changes to the remote repository in your Github account.

Date created
-The project was created by Nawaf on moday 2022
+The project was created by Nawaf on moday 12/2022

Project Title
-Exploring data to take action
+Explore US Bikeshare Data

Description
-Exploring data by python lang
+In this project, you will make use of Python to explore data related to bike share systems for three major cities in the United

States—Chicago, New York City, and Washington. You will write code to import the data and answer interesting questions about it by computing descriptive statistics. You will also write a script that takes in raw input to create an interactive experience in the terminal to present these statistics.

Files used
 i was used the jupyter nootbook, with data file it was chicago,new_york_city,washington, csv

Credits
-will unfortunately until now there is. no credits
+will unfortunately until now there is. no credits.
+
+## Useful website:
+https://stackoverflow.com/
+https://www.python.org/shell/

Conclusion
-The project was very useful and I learned a lot from it about how data analysis can help us make the right decisions
+The project was very useful and I learned a lot from it about how data analysis can help us make the right decisions.
+

 ESCOD
@@ -1,21 +1,26 @@
@@ -1,21 +1,26 @@
@@ -1,21 +1,26 @@
 >**Note**: Please **fork** the current Udacity repository so that you will have a **remote** repository in **your** Github account. Clone the remote repository to your local machine. Later, as a part of the project "Post your Work on Github", you will push your proposed changes to the remote repository in your Github account.

Date created
-The project was created by Nawaf on moday 2022
+The project was created by Nawaf on moday 12/2022

Project Title

-Exploring data to take action
+Explore US Bikeshare Data

 ### Description
-Exploring data by python lang
+In this project, you will make use of Python to explore data related to bike share systems for three major cities in the United States—Chicago, New York City, and Washington. You will write code to import the data and answer interesting questions about it by computing descriptive statistics. You will also write a script that takes in raw input to create an interactive experience in the terminal to present these statistics.

 ### Files used
 i was used the jupyter nootbook, with data file it was chicago,new_york_city,washington, csv

 ### Credits
-will unfortunately until now there is. no credits
+will unfortunately until now there is. no credits.
+
+## Useful website:
+https://stackoverflow.com/
+https://www.python.org/shell/

 ## Conclusion
-The project was very useful and I learned a lot from it about how data analysis can help us make the right decisions
+The project was very useful and I learned a lot from it about how data analysis can help us make the right decisions.
+

(END)

C.	After each change, stage and commit your changes. When you commit your work, you should use a descriptive message of the changes made. Your changes should be small and aligned with your commit message.	$ git commit -m "docs: I was adding a new part which was the conclusion of the project"
		
		$ git commit -m "docs: I was adding a new part which was the useful website of the project, and updated the Date created, Project Title, Description, Conclusion"

D.	Push your changes to the remote repository branch.	$ git push origin documentation
E.	Switch back to the *master* branch.	$ git checkout master
F.	Check the local repository log to see how *all the branches* have changed.	**$ git log --oneline --graph --all**
G.	Go to Github. Notice that you now have two branches available for your project, and when you change branches the README changes.	**No git command needed**

4. Refactor Code

Now you will be working in your local repository, on the code in your BikeShare python file to make improvements to its efficiency and readability. You should repeat steps C through E three times to make at least three commits as you refactor.

	Tasks	Git Commands
A.	Create a branch named *refactoring* on your local repository.	$ git branch refactoring
B.	Switch to the *refactoring* branch.	$ git checkout refactoring
C.	Similar to the process you used in making the documentation changes, make 2 or more changes in refactoring your code.	**No git command needed (edit the code in your python file)**

D.	*For each change,* stage and commit your work with a descriptive message of the changes made.	$ git status,
		& git add .
		$ git commit -m "refactor: I was updating the get_filters function by adding title method to makes all characters lower case except the first one"
		
		$ git status,
		& git add .
		$ git commit -m "refactor: I was updating the display_data function by adding new while loops "

E.	Push your commits to your remote repository branch.	$ git push origin refactoring
F.	Switch back to the *master* branch.	$ git checkout master
G.	Check the local repository log to see how *all the branches* have changed.	$ **git log --oneline --graph --all**

H.	Go to GitHub. Notice that you now have 3 branches. Notice how the files change as you move through the branches.	**No git command needed**

5. Merge Branches

	Tasks	Git Commands
A.	Switch to the *master* branch.	$ git checkout master
B.	Pull the changes you and your coworkers might have made in the passing days (in this case, you won't have any updates, but pulling changes is often the first thing you do each day).	$ git pull origin
C.	Since your changes are all ready to go, merge all the branches into the master. Address any merge conflicts. If you split up your work among your branches correctly, you should have no merge conflicts.	git merge documentation git merge refactoring
D.	You should see a message that shows the changes to the files, insertions, and deletions.	**No git command needed**
E.	Push the repository to your remote repository.	$ git push

F.	Go to GitHub. Notice that your master branch has all of the changes.	**No git command needed**

Submission:

This concludes the project.
- Please review this document to make sure you entered all the required response fields in all four sections.
- Download this document as a PDF file.
- Submit the PDF file on the Project Submission page within the Udacity Classroom.

Resources:
- **https://www.atlassian.com/git/glossary**
- **https://confluence.atlassian.com/bitbucketserver/basic-git-commands-776639767.html**
- **https://youtu.be/XTolZqmZq6s**
- **https://youtu.be/ytSoabxSQ6E**