Python in QGIS & ArcGIS

Exercise 2

2.1 – My first Script

Goal of this Exercise
You will use this exercise to get used to the basic syntax and classes in python, gain some experience with coding and synchronize your data with your GitHub account.

When writing python code, please provide meaningful comments.

Materials
No (geo)data materials are available for this task.

Tasks (10 Points)
In a first step write three simple functions in one Python file (module) (6 points) that are then called in the 'main' part of this module (2 points). You can copy the 'main' part from below and just add the function definitions. Please make sure that the code works without exceptions for the parameters provided in the example file. Upload your solution to GitHub (2 points).

Below you'll find the basic structure of the functions.

1st function:

```
# donuts
# Given an integer count of a number of donuts, return a string
# of the form 'Number of donuts: <count>', where <count> is the number
# passed in. However, if the count is 10 or more, then use the word
'many'
# instead of the actual count.
# So donuts(5) returns 'Number of donuts: 5'
# and donuts(23) returns 'Number of donuts: many'
def donuts(count):
    # +++ your code here +++
    return
```

2nd function

```
# verbing
# Given a string, if its length is at least 3,
# add 'ing' to its end.
# Unless it already ends in 'ing', in which case
# add 'ly' instead.
# If the string length is less than 3, leave it unchanged.
# Return the resulting string.
def verbing(s):
  # +++your code here+++
  return
```

3rd function

```
# Remove adjacent
# Given a list of numbers, return a list where
# all adjacent == elements have been reduced to a single element,
# so [1, 2, 2, 3] returns [1, 2, 3]. You may create a new list or
# modify the passed in list.
def remove_adjacent(nums):
  # +++your code here+++
  return
```

Please use the following code to test your functions. The main() function contains some scenarios for each function. As already said, please make sure that the code works without exceptions for the parameters provided in the example file. **Please remember to comment your functions accordingly!**

```
def donuts(count):
    # +++your code here+++
    return

def verbing(s):
    # +++your code here+++
    return

def remove_adjacent(nums):
    # +++your code here+++
    return


def main():
    print('donuts')

    print(donuts(4))
    print(donuts(9))
    print(donuts(10))
    print(donuts('twentyone'))

    print('verbing')
    print(verbing('hail'))
    print(verbing('swiming'))
    print(verbing('do'))

    print('remove_adjacent')
    print(remove_adjacent([1, 2, 2, 3]))
    print(remove_adjacent([2, 2, 3, 3, 3]))
    print(remove_adjacent([]))


# Standard boilerplate to call the main() function.
if __name__ == '__main__':
  main()
```

In a final step please log in to your github account, create a new repository named **Python Course ifgi**. Create a Subfolder **exercise_2** and push your commented solution there.

Invite Sven and me as contributors to your repo using the following email addresses:

- s.harpering@gmail.com
- p.rieffel+github@gmail.com

Please use the Template **Exercise Sheet Template Solution** in Learnweb to provide a URL to your repository. Please upload the file as a Word document.

General information
The deadline is announced on Learnweb.