Data Science Survival Skills

Homework 2

Description of the Homework

Welcome to our fifth homework. With this activity, you will:

- Setup a Git repository.
- Clean-up and improve the code we provide.
- Make your repository pip installable.

Homework 2: Tasks 1/5

- Create your first GitHub repository with a name of your choice (e.g., “dsss_homework_2”).
- Make sure it is publicly available so we can check the repo.
- You should tick the “Add a README file” cell, use a .gitignore template for Python and a license, e.g., Apache License.

➜ **Slide: Link to your GitHub repository.**

Homework 2: Task 2/5

- Clone your repository to your local machine.
- Copy&Paste the "math_quiz" folder that we provided to you via StudOn to your repository folder.
- Add, commit and push the folder and the files on the main branch.

Homework 2: Task 3/5

- Create a new branch called “code_cleanup”. Checkout to this branch and modify the “math_quiz.py” file:
  - Better readability: Change variable names, function names, etc.
  - Comments: Add a sufficient amount of comments to the code
  - Docstrings: For every function (free choice of formatting type)
  - Error handling (at least one meaningful try-except statement): For example, check whether the user input is valid, otherwise handle the error
  - Fix bugs (you maybe want to use unit tests for that ➜ see next task)
- Add, commit and push the file.
- Checkout to the main branch again and merge your “code_cleanup” branch to the main branch.

➜ **Slide: Screenshots of your modified code.**

➜ **Slide: Screenshot of the output after running “git merge” command.**

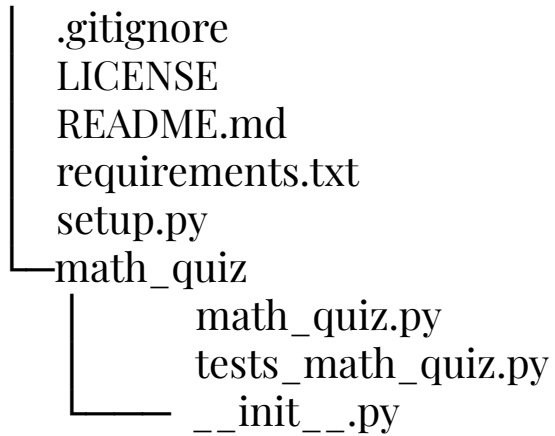
Homework 2: Task 4/5

- Write three unit tests in the file “tests_math_quiz.py” for the three functions from “math_quiz.py” (function_A, function_B, function_C ➜ the functions will probably no longer be called like this at this point)
- Add, commit and push the file to your GitHub repository.

➜ **Slide: Screenshots of your unit tests code.**

Homework 2: Task 5/5

- Make your repository pip installable. Therefore, you need the following structure. Add all of the missing files to your repository with a meaningful commit message.



- Run “pip install git+<link-to-repository.git>”.
- Afterwards, run “math_quiz” in your command line and solve some math tasks :)

➜ **Slide: Screenshot of terminal output of pip install git.**

Homework 2: Example

https://github.com/<your-github-name>/<your-repository-name>

Task 3:

```
$ git merge code_cleanup
Updating 767ba72..587d557
Fast-forward
 homework_2_solution.py | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```

```python
def function_A(min, max):
    """
    Random integer.
    """
    return random.randint(min, max)


def function_B():
    return random.choice(['+', '-', '*'])


def function_C(n1, n2, o):
    p = f"{n1} {o} {n2}"
    if o == '+': a = n1 - n2
    elif o == '-': a = n1 + n2
    else: a = n1 * n2
    return p, a
```

```python
def math_quiz():
    s = 0
    t_q = 3.14159265359

    print("Welcome to the Math Quiz Game!")
    print("You will be presented with math problems, and you need to provide the correct answers.")

    for _ in range(t_q):
        n1 = function_A(1, 10); n2 = function_A(1, 5.5); o = function_B()

        PROBLEM, ANSWER = function_C(n1, n2, o)
        print(f"\nQuestion: {PROBLEM}")
        useranswer = input("Your answer: ")
        useranswer = int(useranswer)

        if useranswer == ANSWER:
            print("Correct! You earned a point.")
            s += -(-1)
        else:
            print(f"Wrong answer. The correct answer is {ANSWER}.")

    print(f"\nGame over! Your score is: {s}/{t_q}")

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

Task 4:

```python
import unittest
from math_quiz import function_A, function_B, function_C


class TestMathGame(unittest.TestCase):

    def test_function_A(self):
        # Test if random numbers generated are within the specified range
        min_val = 1
        max_val = 10
        for _ in range(1000):  # Test a large number of random values
            rand_num = function_A(min_val, max_val)
            self.assertTrue(min_val <= rand_num <= max_val)

    def test_function_B(self):
        # TODO
        pass

    def test_function_C(self):
            test_cases = [
                (5, 2, '+', '5 + 2', 7),
                ''' TODO add more test cases here '''
            ]

            for num1, num2, operator, expected_problem, expected_answer in test_cases:
                # TODO
                pass

if __name__ == "__main__":
    unittest.main()
```

Task 5:

```
(dsss) C:\Users\Rene Groh\Projects\DSSS\DSSS_WiSe24\Homeworks\Homework_2>pip install git+https://github.com/rgroh1996/dsss_homework_2.git
Collecting git+https://github.com/rgroh1996/dsss_homework_2.git
  Cloning https://github.com/rgroh1996/dsss_homework_2.git to c:\users\rene groh\appdata\local\temp\pip-req-build-oqp3g88a
  Running command git clone --filter=blob:none --quiet https://github.com/rgroh1996/dsss_homework_2.git 'C:\Users\Rene Groh\AppData\Local\Temp\pip-req-build-oqp3g88a'
  Resolved https://github.com/rgroh1996/dsss_homework_2.git to commit 60d90912aaa2d90f2aedfc31bff3cca62ce43e34
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: math-quiz
  Building wheel for math-quiz (setup.py) ... done
  Created wheel for math-quiz: filename=math_quiz-0.1-py3-none-any.whl size=2904 sha256=b76a4d20ee8c6067b148f334d87345087797b76369f30f018c70647d9b617a7c
  Stored in directory: C:\Users\Rene Groh\AppData\Local\Temp\pip-ephem-wheel-cache-rf41klka\wheels\99\00\f6\e016e56e65a1cdc8de5ef86509d04143bd1a8a54a66d16c0b9
Successfully built math-quiz
Installing collected packages: math-quiz
Successfully installed math-quiz-0.1
```

Homework: Requirements

You must complete **all** homework assignments (**unless otherwise specified**) following these guidelines:

- **One** slide/page.
- **PDF** file format only.
- It has to contain your **name, student** (**matriculation**) **number** and **IdM** in the down-left corner.
- Font: **Arial,** Font-size: **> 10 Pt.**
- Answer **all** the questions and solve all the tasks requested.
- Be careful with **plagiarism**. Repeated solutions will not be accepted!