

**ANL252 (Online)**

**Python for Data Analytics**

# **Tutor-Marked Assignment**

**July 2023 Presentation**

**Submitted by:**

|  |  |
| --- | --- |
| **Name** | **PI No.** |
| **Nur Fazillah Binte Abdul Rahman** | **K1870536** |

**Tutorial Group: ­­­­­­­­­­­­ TV 03**

**Instructor’s Name: Mr. Munish Kumar**

**Submission Date: 15/09/2023**

**Question 1**

1. Reasons for plagiarism in coding:

* It is possible that students have difficulty with their assignment, or they are enticed to use open-source software for their tasks.
* Students may not be aware of what programming plagiarism is.
* Students want to attain a good grade for the coursework.
* Lack of interest in the subject matter, hence, plagiarise to get assignment done.
* Copying other’s developer code is common as this save a lot of time and effort to create the same coding repeatedly.

Ways to avoid plagiarism in coding:

* Reference your sources and cite when required.
* In your code, add comments and cite the original source you copied or borrow.
* Start your code a fresh without any external sources.
* Use a plagiarism checker to check your code and confirm that it is original.
* Avoid using code that is similar to the original source or those that only have minor changes such as renaming variable or changing a number.

import random

number = random.randint(1, 100)

while True:

guess = int(input("Guess a number between 1 and 100: "))

if guess == number:

print("Congratulations! You guessed the number!")

break

elif guess < number:

print("The number is higher than your guess. Try again.")

else:

print("The number is lower than your guess. Try again.")

This code imports the random module and generates a random integer between 1 and 100 using the randint() function(Python Examples | Programiz, n.d.).It then enters a while loop that keeps running until the user guesses the correct number. Inside the loop, it prompts the user to enter a number using the input() function and converts it to an integer using int()(Navone, 2023). If the user’s guess is correct, it prints a congratulatory message and exits the loop using break. If the guess is incorrect, it prints a message indicating whether the actual number is higher or lower than the guess and continues with another iteration of the loop (Navone, 2023).

import random

number = random.randint(1, 100)

for i in range(10):

guess = int(input("Guess a number between 1 and 100: "))

if guess == number:

print("Congratulations! You guessed the number!")

break

elif guess < number:

print("The number is higher than your guess. Try again.")

else:

print("The number is lower than your guess. Try again.")

* Improved code quality: By making the codebase more efficient, scalable, and maintainable, rewriting the code can raise its quality. (Sharovar, 2022)
* Better alignment with business goals: Coding changes can assist the codebase be more in line with the aims of the present company. (Rewrite (Programming), 2023)
* Reduced technical debt: By removing outdated or useless legacy code, rewriting code can assist in reducing technical debt. (When to Rewrite a Code Base From Scratch, n.d.)

**Question 2**

* Use f-strings for input prompts: You can directly format the prompt using f-strings rather than converting it to a string using the str() method. As an illustration, item = input(f"Hello! What would you like to purchase?) (Python, 2023).
* Use a while loop instead of a for loop: A while loop, which runs until the user tells it to stop, can be used in place of a for loop with a set number of iterations. This enables greater flexibility and an improved user experience. (Python, 2023).
* Add error handling: Instead of a for loop with a predetermined number of iterations, a while loop can be used, which continues to run until the user instructs it to stop. Greater flexibility and a better user experience are made possible as a result (Hasan, 2021).

Improved code

products = ['laptop', 'mouse', 'webcam', 'keyboard', 'speaker']

updated\_items = []

print(f'We have a list of products here: {products}.')

while True:

item = input(f"Hello! What do you want to buy? ")

if item not in products:

print(f'Wrong product! Please try again.')

continue

try:

price\_of\_item = float(input("How much is it (in SGD)? "))

except ValueError:

print("Invalid input. Please enter a number.")

continue

entered\_input = [item, price\_of\_item]

updated\_items.append(entered\_input)

query = input("Would you like to continue? (yes/no) ")

if query.lower() != 'yes':

break

print(f'This is our updated shopping list: {updated\_items}')

**References**

Python Examples | Programiz. (n.d.). <https://www.programiz.com/python-programming/examples>

Navone, E. C. (2023, July 7). Python Code Example Handbook – Sample Script Coding Tutorial for Beginners. freeCodeCamp.org. <https://www.freecodecamp.org/news/python-code-examples-sample-script-coding-tutorial-for-beginners/>

Sharovar, E. (2022, December 17). *Code Rewriting: When and Why*. Waverley. <https://waverleysoftware.com/blog/code-rewriting-when-and-why/>

*Rewrite (programming)*. (2023, September 11). Wikipedia. <https://en.wikipedia.org/wiki/Rewrite_%28programming%29>

*When to rewrite a code base from scratch*. (n.d.). Stack Overflow. <https://stackoverflow.com/questions/1064403/when-to-rewrite-a-code-base-from-scratch/>

Python, R. (2023, January 25). *Python Code Quality: Tools & Best Practices*. realpython.com. <https://realpython.com/python-code-quality/>

Hasan, M. (2021, July 26). *The 10 Best and Useful Tips To Speed Up Your Python Code*. UbuntuPIT. <https://www.ubuntupit.com/best-and-useful-tips-to-speed-up-your-python-code/>