



College of
Science and
Technology

Assignment of CSF101

Submitted: Sonam Yeshi

Index No: 02240324

Introduction

First code, is a versatile utility tool that performs various tasks, including prime number summation, unit conversion, text analysis, and file processing. It features a user-friendly menu, allowing users to choose functions like palindrome checking, consonant counting, and finding the min/max in a list. With clear prompts and error handling, it provides a seamless experience for everyday computations.

Second code is a fun and playful interactive program that offers two simple yet engaging games—Guess the Number and Rock, Paper, Scissors—all while maintaining a lighthearted and casual tone. The user is welcomed with affectionate phrases like "Honney" and "Babe," which gives the entire experience a sweet, charming, and fun atmosphere. It's not just about playing games; it's about creating playful interaction with the user.

The Guess the Number game is designed to be a guessing challenge where the user must guess a number between 15 and 150. The computer will keep track of the correct number, and the user is given hints whether their guess is too high or too low. It continues until the user successfully guesses the right number, with playful feedback along the way. If the user enters an invalid input, such as a non-numeric entry, the program gently prompts them to enter a valid number.

The Rock, Paper, Scissors game invites the user to choose one of three options—rock, paper, or scissors—and then compares their choice to the computer's randomly selected choice. The game proceeds with a comparison of the two options and declares a winner (or a tie if both choices are the same). As with the first game, there's playful feedback after each round, and the user is given the option to play again or exit.

The entire experience is controlled through an intuitive game menu that presents the user with a choice of games or the option to exit. The menu guides the user to select one of the options, and the games continue until the user decides they want to stop playing.

If the user makes an invalid choice or enters an incorrect input, the script handles the error gracefully by reminding them with gentle and humorous prompts to make a valid selection. The program uses loops to ensure that users are continually prompted until they provide a valid input or choose to exit.

Code 1:

```
def is_prime(n):
    if n < 2:
        return False
    for i in range(2, int(n ** 0.5) + 1):
        if n % i == 0:
            return False
    return True

def sum_primes(start, end):
    return sum(n for n in range(start, end + 1) if is_prime(n))

def length_converter(value, unit):
    if unit.lower() == "feet":
        return value * 0.3048
    elif unit.lower() == "meter":
        return value / 0.3048
    else:
        return None

def count_consonants(string):
    vowels = "aeiouAEIOU"
    return sum(1 for char in string if char.isalpha() and char not in vowels)

def is_palindrome(string):
    cleaned = "".join(char.lower() for char in string if char.isalnum())
    return cleaned == cleaned[::-1]

def find_min_max(numbers):
    return min(numbers), max(numbers)
```

Cont....

```
def count_words_in_file(filename):
    try:
        with open(filename, 'r', encoding='utf-8') as file:
            return len(file.read().split())
    except FileNotFoundError:
        return "File not found."

def main():
    while True:
        print("\nMenu:")
        print("1. Sum of prime numbers in a range")
        print("2. Length unit converter (Feet, Meter)")
        print("3. Consonant counter")
        print("4. Palindrome checker")
        print("5. Find Min and Max number in a list")
        print("6. Word counter in a text file")
        print("7. Exit")

        choice = input("Enter your choice (1-7): ")

        if choice == "1":
            start = int(input("Enter start range: "))
            end = int(input("Enter end range: "))
            print("Sum of prime numbers:", sum_primes(start, end))

        elif choice == "2":
            value = float(input("Enter value: "))
            unit = input("Enter unit (Feet/Meter): ")
            converted = length_converter(value, unit)
            if converted is not None:
                print(f"Converted value: {converted:.2f}")
            else:
                print("Invalid unit. Please enter Feet or Meter.")
```

Cont....

```
elif choice == "3":  
    string = input("Enter a string: ")  
    print("Number of consonants:", count_consonants(string))  
  
elif choice == "4":  
    string = input("Enter a string: ")  
    if is_palindrome(string):  
        print("The string is a palindrome.")  
    else:  
        print("The string is not a palindrome.")  
  
elif choice == "5":  
    numbers = list(map(int, input("Enter numbers separated by space: ").split()))  
    min_num, max_num = find_min_max(numbers)  
    print(f"Minimum number: {min_num}, Maximum number: {max_num}")  
  
elif choice == "6":  
    filename = input("Enter the filename: ")  
    print("Word count:", count_words_in_file(filename))  
  
elif choice == "7":  
    print("Exiting the program.")  
    break  
  
else:  
    print("Invalid choice. Please enter a number between 1 and 7.")  
  
if __name__ == "__main__":  
    main()  
    I
```

Output:

Menu:

- 1. Sum of prime numbers in a range**
- 2. Length unit converter (Feet, Meter)**
- 3. Consonant counter**
- 4. Palindrome checker**
- 5. Find Min and Max number in a list**
- 6. Word counter in a text file**
- 7. Exit**

Enter your choice (1-7):

Code 2:

```
1 import random
2
3 def guess_number():
4     number = random.randint(15, 150)
5     while True:
6         try:
7             love = int(input("Guess the number (15-150): "))
8             if love < number:
9                 print("I wish we were right, it's too low.")
10            elif love > number:
11                print("Will you try again for us, it's too high.")
12            else:
13                print("WOW! You won my heart. You win.")
14                break
15        except ValueError:
16            print("Don't be silly Honney.")
17
18
19 def rock_paper_scissors():
20     choices = ["rock", "paper", "scissors"]
21     while True:
22         user_choice = input("Choose rock, paper, or scissors: ").lower()
23         if user_choice not in choices:
24             print("Don't be silly Honney. Make it once.")
25
26         computer= random.choice(choices)
27         print(f"Computer chose: {computer}")
28
29         if user_choice == computer:
30             print("Have you read my mind Honney. It's a tie!")
31         elif (user_choice == "rock" and computer == "scissors") or \
32             (user_choice == "scissors" and computer == "paper") or \
33             (user_choice == "paper" and computer == "rock"):
34             print("Opz! Babe you are blushing me. You win!")
35         else:
36             print("You lose! Babe catch me now.")
```

Cont....

```
if input("DO u want to play again? (y/n): ").lower() != 'y':
    break

def game_menu():
    while True:
        print("\nGame Choice is on you Babe:")
        print("1. Guess Number Game")
        print("2. Rock, Paper, Scissors")
        print("3. Exit")

        choice = input("Enter Your Best One Honney: ")
        if choice == '1':
            guess_number()
        elif choice == '2':
            rock_paper_scissors()
        elif choice == '3':
            break
        else:
            print("Don't be silly Honney, enter a number(1/2/3).")

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

Output:

Game Choice is on you Babe:

- 1. Guess Number Game**
- 2. Rock, Paper, Scissors**
- 3. Exit**

Enter Your Best One Honney:

Out put:

```
Choose rock, paper, or scissors: paper
Computer chose: rock
Opz! Babe you are blushing me. You win!
DO u want to play again? (y/n): n
```

Game Choice is on you Babe:

- 1. Guess Number Game
- 2. Rock, Paper, Scissors
- 3. Exit

Enter Your Best One Honney: 1

Guess the number (15-150): 148

Will u try again for us, it's to high.

Guess the number (15-150): █

Conclusion

First code, program serves as a simple yet powerful tool for performing a variety of everyday computations, including numerical operations, text analysis, and file processing. With its user-friendly interface and robust functionality, it ensures a smooth experience for users of all levels. Whether you need to check prime numbers, convert measurements, or analyze text, this program provides a quick and efficient solution.

This fun and interactive Python program allows users to enjoy two simple yet engaging games: Guess the Number and Rock, Paper, Scissors. With a playful and affectionate tone, the game provides a delightful user experience while demonstrating fundamental programming concepts such as loops, conditionals, randomization, and user input handling.

Through this project, we explored:

- Random number generation for unpredictability
- Loop structures to keep the games running until the user chooses to exit
- Exception handling to manage invalid inputs gracefully
- User-friendly prompts for an engaging and fun experience

This game is a great way to practice Python basics while making the experience more enjoyable with a personalized touch. Future improvements could include score tracking, difficulty levels, or a graphical user interface (GUI) to make it even more interactive.

Repo link: https://github.com/Sonam12yeshi/SonamYeshi_02240324_A1.git