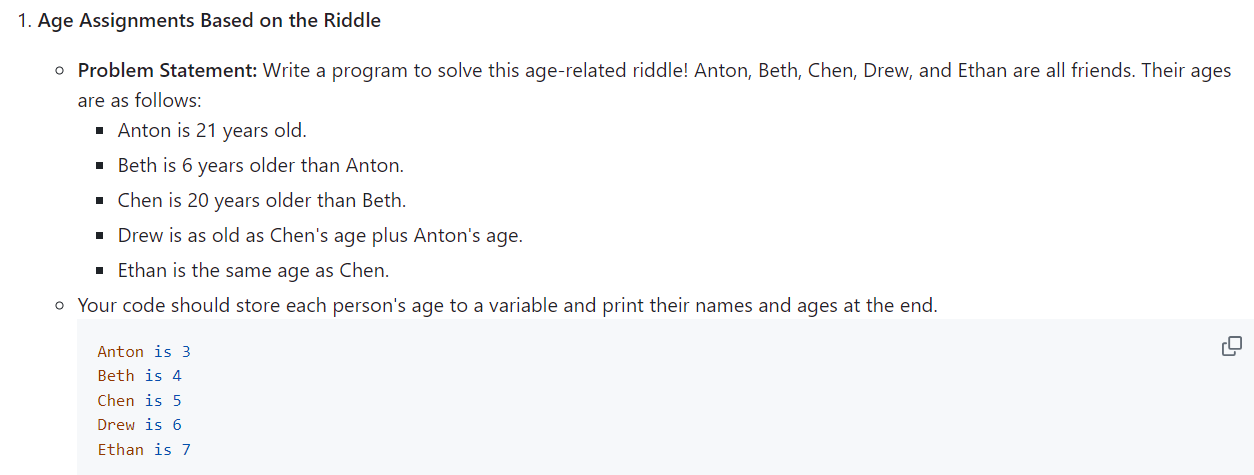
**Python Programming Assignment 01**



# Age Assignments Based on the Riddle

# Given age of Anton

anton\_age : int = 21

# Calculate ages based on the given relationships

beth\_age : int = anton\_age + 6

chen\_age : int = beth\_age + 20

drew\_age : int = chen\_age + anton\_age

ethan\_age : int = chen\_age

# Output the results

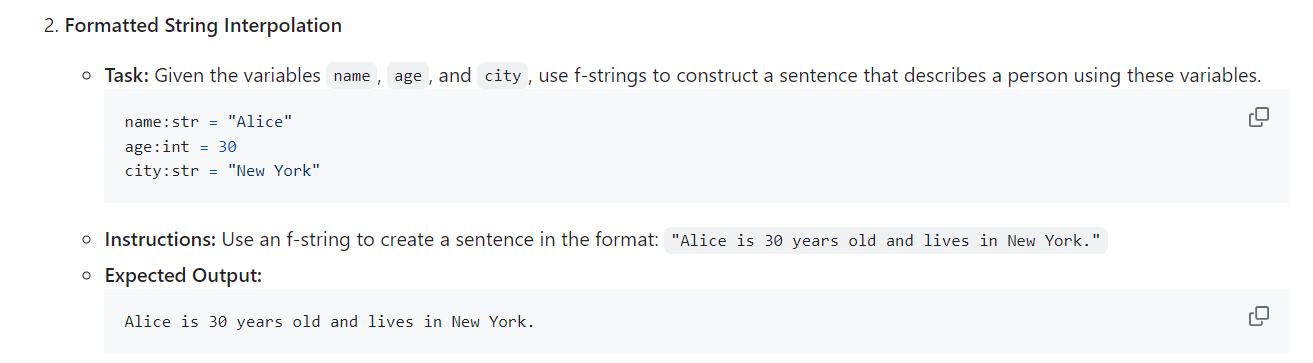
print(f"Anton is {anton\_age}")

print(f"Beth is {beth\_age}")

print(f"Chen is {chen\_age}")

print(f"Drew is {drew\_age}")

print(f"Ethan is {ethan\_age}")



# Formatted String Interpolation

name : str = "Alice"

age : int = 30

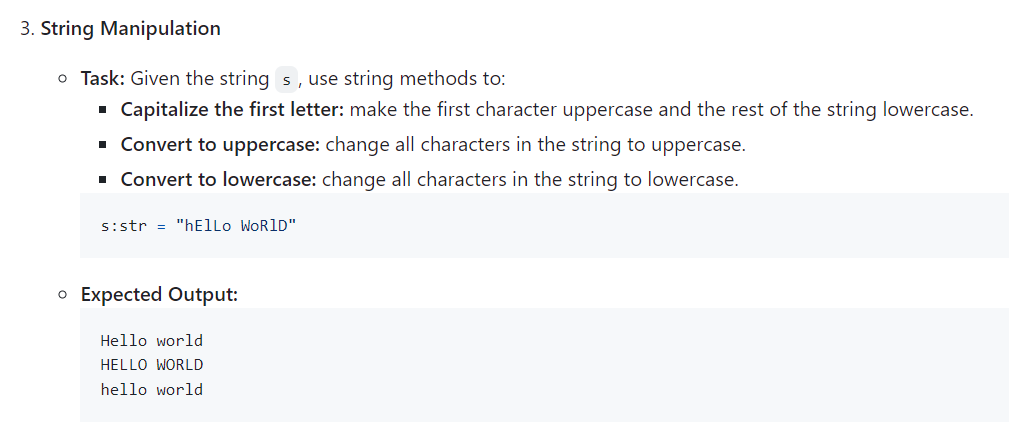
city : str = "New York"

# Construct the sentence using an f-string

sentence : str = f"{name} is {age} years old and lives in {city}."

# Output the result

print(sentence)



# String Manipulation

s:str = "hElLo WoRlD"

# Capitalize the first letter (first character uppercase, rest lowercase)

capitalized = s.capitalize()

# Convert the entire string to uppercase

uppercase = s.upper()

# Convert the entire string to lowercase

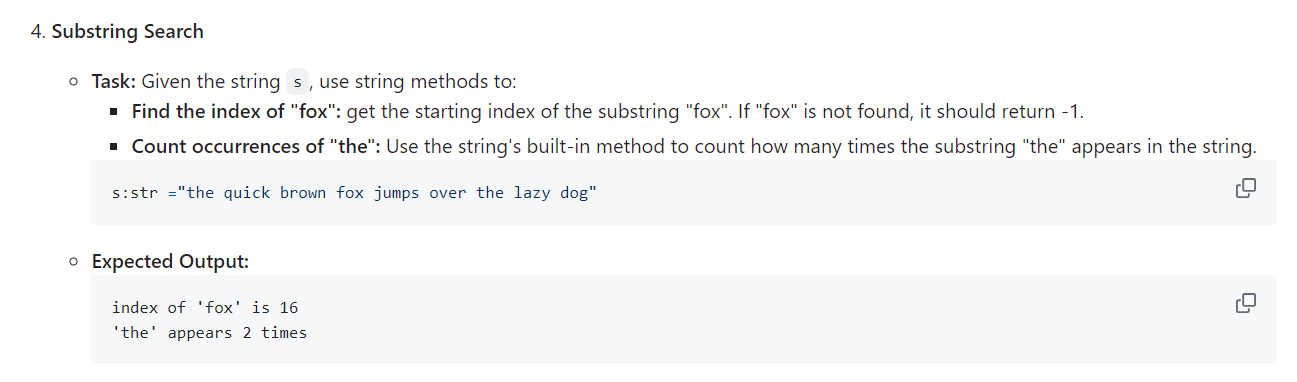
lowercase = s.lower()

# Output the results

print("Capitalized: ", capitalized)

print("Uppercase: ", uppercase)

print("Lowercase: ", lowercase)



# Substring Search

s:str ="the quick brown fox jumps over the lazy dog"

# Find the index of "fox"

index\_of\_fox : int = s.find("fox")

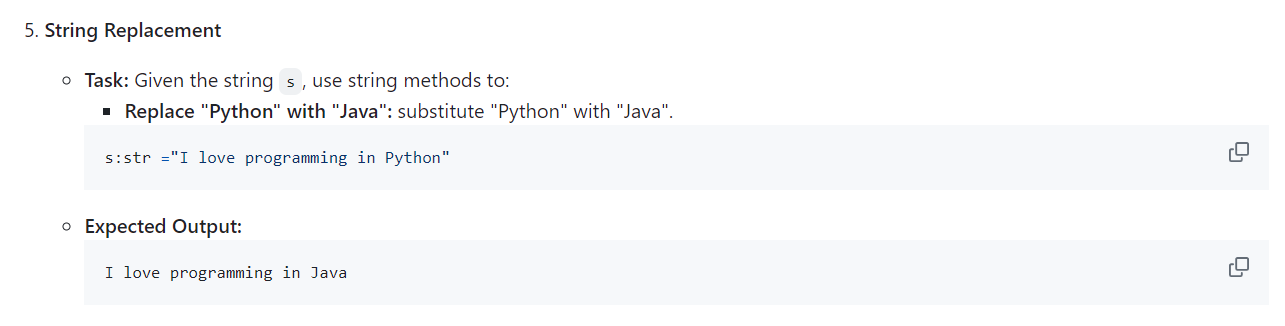
# Count occurrences of "the"

count\_of\_the : int = s.count("the")

# Output the results

print(f"index of 'fox' is {index\_of\_fox}")

print(f"'the' appears {count\_of\_the} times")



# String Replacement

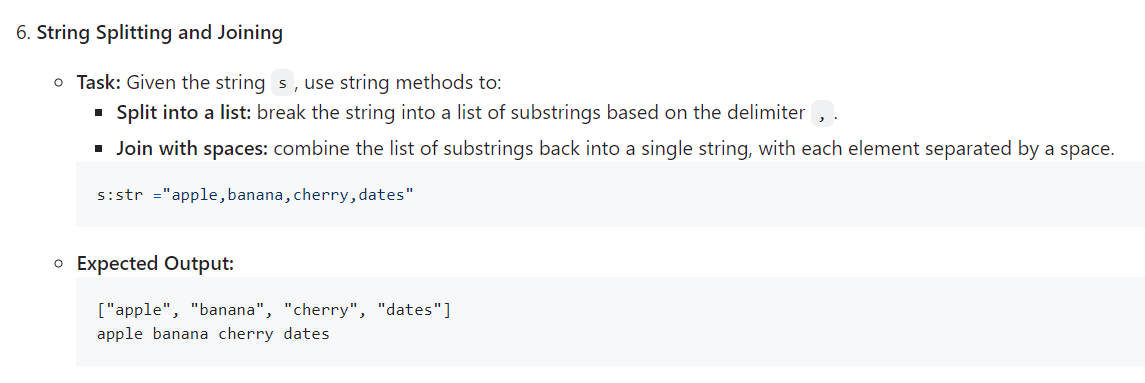
s:str ="I love programming in Python"

# Replace "Python" with "Java"

substitute\_sentence : str = s.replace("Python", "Java")

# Output the result

print(substitute\_sentence)



# String Splitting and Joining

s:str ="apple,banana,cherry,dates"

# break the string into a list of substrings based on the delimiter ,

s\_list : list[str] = s.split(",")

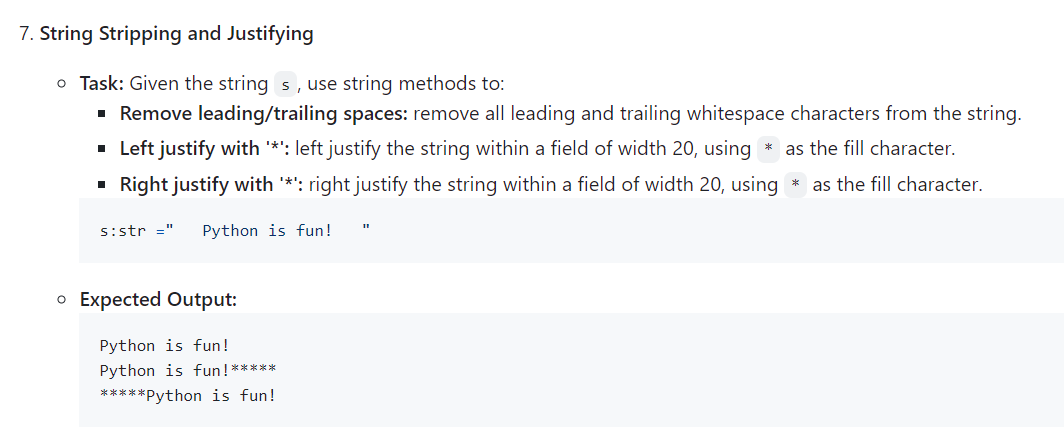
# combine the list of substrings back into a single string, with each element separated by a space

s\_joined : str = " ".join(s\_list)

# Output the results

print(s\_list)

print(s\_joined)



# String Stripping and Justifying

s:str ="   Python is fun!   "

# Remove leading and trailing spaces

s\_trimmed : str = s.strip()

# left justify the string within a field of width 20, using \* as the fill character

s\_left\_justified : str = s\_trimmed.ljust(20, '\*')

# right justify the string within a field of width 20, using \* as the fill character

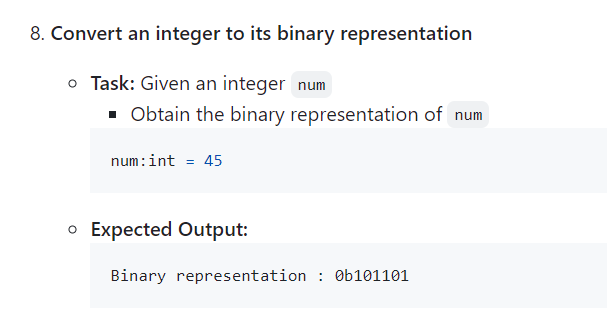
s\_right\_justified : str = s\_trimmed.rjust(20, '\*')

# Output the results

print(s\_trimmed)

print(s\_left\_justified)

print(s\_right\_justified)



# Convert an integer to its binary representation

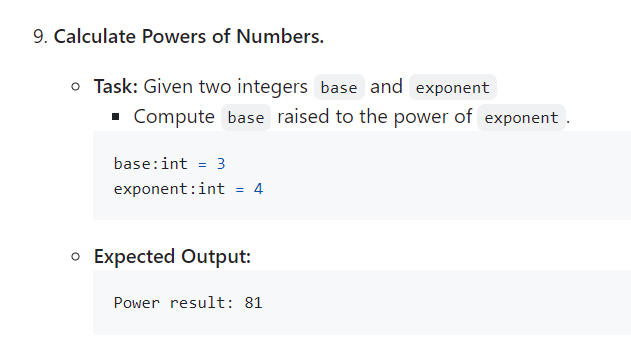
num : int = 45

# Get the binary representation of the number

binary\_representation : str = bin(num)

# Output the result

print(f"Binary representation: {binary\_representation}")



# Calculate Powers of Numbers.

base : int = 3

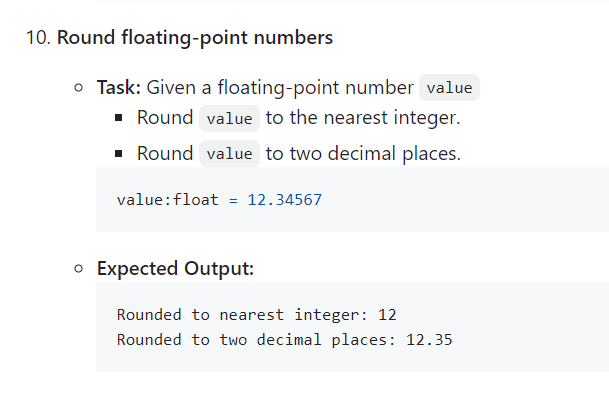
exponent : int = 4

# Calculate the power

power\_result : int = base \*\* exponent

# Output the result

print(f"Power result: {power\_result}")



# Round floating-point numbers

value : float = 12.34567

# Round to the nearest integer

rounded\_integer : int = round(value)

# Round to two decimal places

rounded\_two\_decimals : float = round(value, 2)

# Output the results

print(f"Rounded to nearest integer: {rounded\_integer}")

print(f"Rounded to two decimal places: {rounded\_two\_decimals}")