# Mock Test 1 - Data Science - PPT - PW Skills

# 17. Write a function that takes a list of numbers as input and returns a new list containing only the even numbers from the input list. Use list comprehension to solve this problem.

# Example:

# Input: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

# Output: [2, 4, 6, 8, 10]

def get\_even\_numbers(input\_list):

return [num for num in input\_list if num % 2 == 0]

This function as follows:

input\_list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

output\_list = get\_even\_numbers(input\_list)

print(output\_list)

Output:

[2, 4, 6, 8, 10]

# 18. Implement a decorator function called ‘timer’ that measures the execution time of a function. The ‘timer’ decorator should print the time taken by the decorated function to execute. Use the ‘time’ module in Python to calculate the execution time. Example: import time @timer def my\_function():     # Function code goes here     time.sleep(2) my\_function() Output: "Execution time: 2.00123 seconds"

import time

from functools import wraps

def timer(func):

@wraps(func)

def wrapper(\*args, \*\*kwargs):

start\_time = time.time()

result = func(\*args, \*\*kwargs)

end\_time = time.time()

execution\_time = end\_time - start\_time

print(f"Execution time: {execution\_time:.5f} seconds")

return result

return wrapper

The @timer decorator on your function to measure its execution time:

@timer

def my\_function():

# Function code goes here

time.sleep(2)

my\_function()

my\_function(), it will print the execution time in seconds, like this:

Execution time: 2.00123 seconds

# 19. Write a function called ‘calculate\_mean’ that takes a list of numbers as input and returns the mean (average) of the numbers. The function should calculate the mean using the sum of the numbers divided by the total count. Example: def calculate\_mean(numbers):     total = sum(numbers)     count = len(numbers)     mean = total / count     return mean data = [10, 15, 20, 25, 30] mean\_value = calculate\_mean(data) print("Mean:", mean\_value) Output: Mean: 20.0

The implementation of the calculate\_mean function:

def calculate\_mean(numbers):

total = sum(numbers)

count = len(numbers)

mean = total / count

return mean

data = [10, 15, 20, 25, 30]

mean\_value = calculate\_mean(data)

print("Mean:", mean\_value)

Output:

Mean: 20.0

The function takes a list of numbers as input, calculates the sum of the numbers using the sum() function, counts the total number of elements in the list using the len() function, and then divides the sum by the count to calculate the mean. The mean value is returned as the result. In the example, the list [10, 15, 20, 25, 30] is passed to the function, and the calculated mean value of 20.0 is printed.

# 20. Write a function called ‘perform\_hypothesis\_test’ that takes two lists of numbers as input, representing two samples. The function should perform a two-sample t-test and return the p-value. Use the ‘scipy.stats’ module in Python to calculate the t-test and p-value. Example: from scipy import stats def perform\_hypothesis\_test(sample1, sample2):     t\_statistic, p\_value = stats.ttest\_ind(sample1, sample2)     return p\_value sample1 = [5, 10, 15, 20, 25] sample2 = [10, 20, 30, 40, 50] p\_value = perform\_hypothesis\_test(sample1, sample2) print("P-value:", p\_value) Output: P-value: 0.1064706396450037

from scipy import stats

def perform\_hypothesis\_test(sample1, sample2):

t\_statistic, p\_value = stats.ttest\_ind(sample1, sample2)

return p\_value

This function by providing two lists of numbers as input samples, just like in the example you provided:

sample1 = [5, 10, 15, 20, 25]

sample2 = [10, 20, 30, 40, 50]

p\_value = perform\_hypothesis\_test(sample1, sample2)

print("P-value:", p\_value)

The Output:

P-value: 0.1064706396450037

The p-value represents the probability of observing the data under the null hypothesis that the two samples have the same mean. In this example, the p-value is approximately 0.1065, which is greater than the typical significance level of 0.05. Therefore, we do not have enough evidence to reject the null hypothesis and conclude that the two samples have significantly different means.