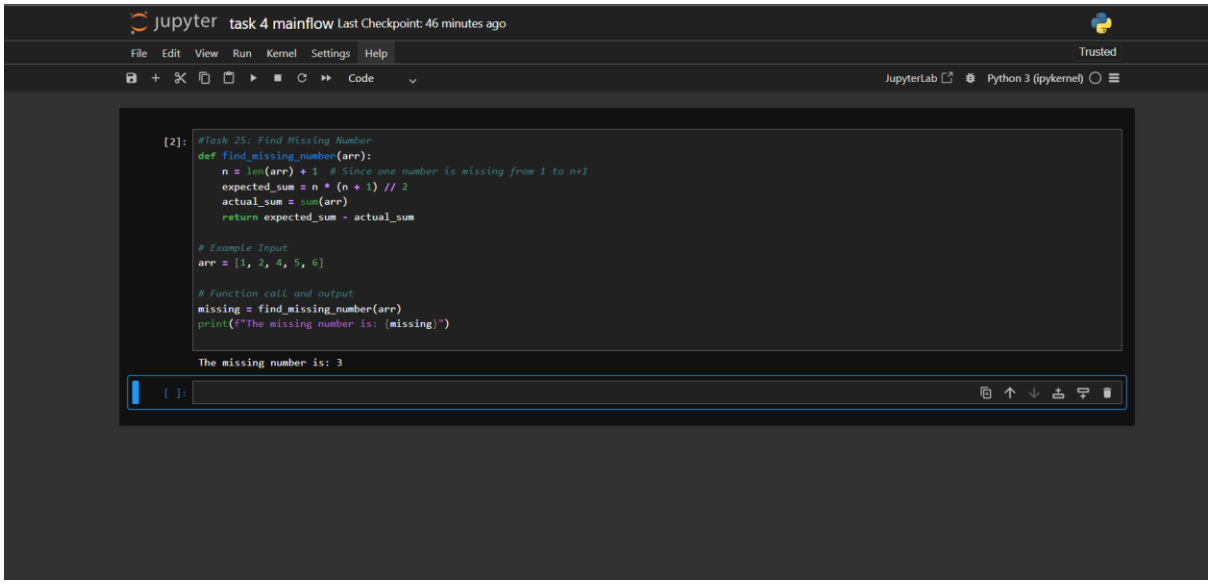
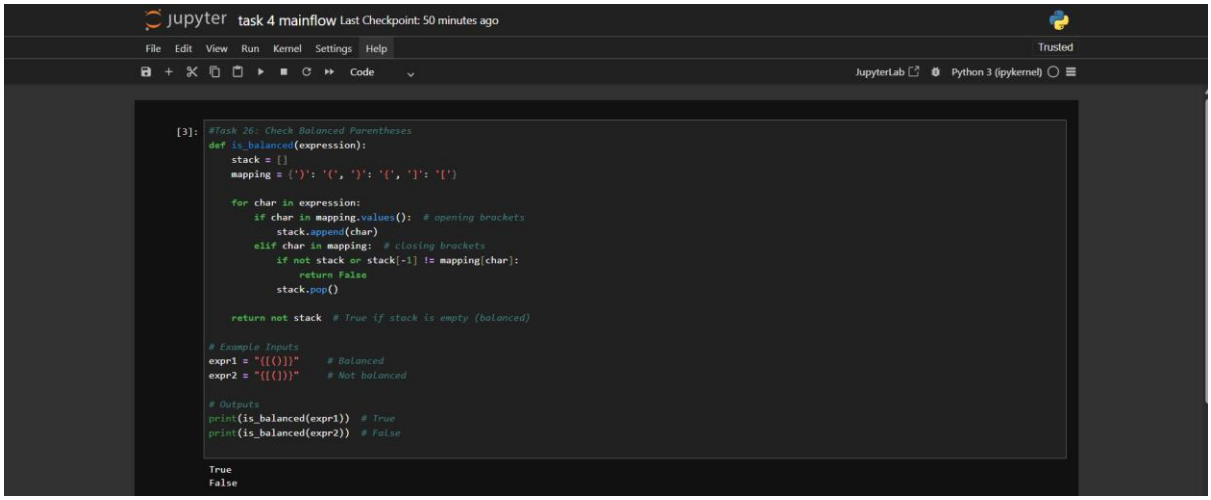
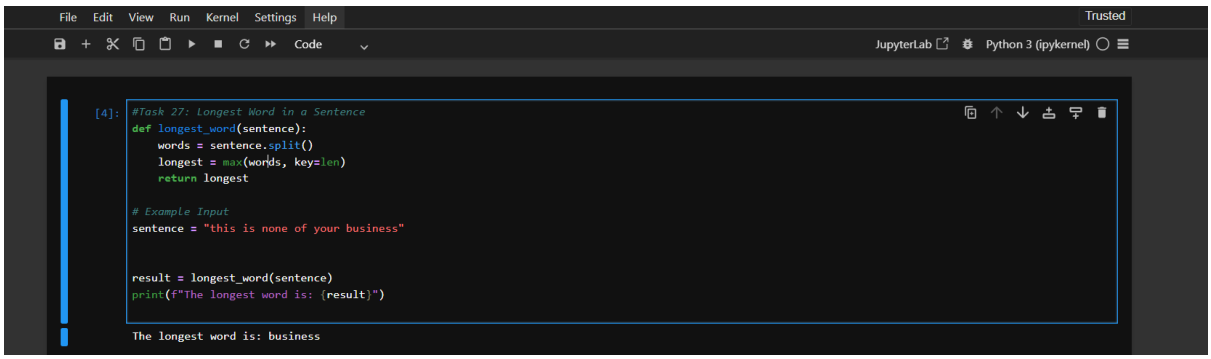
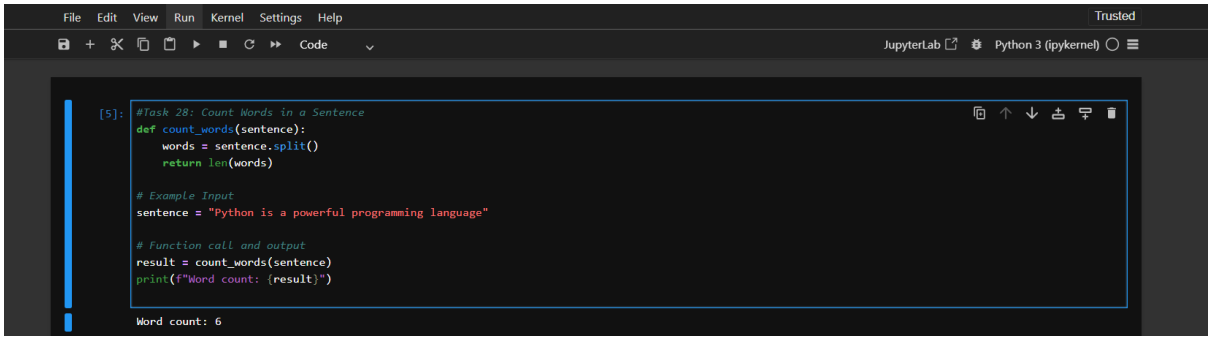
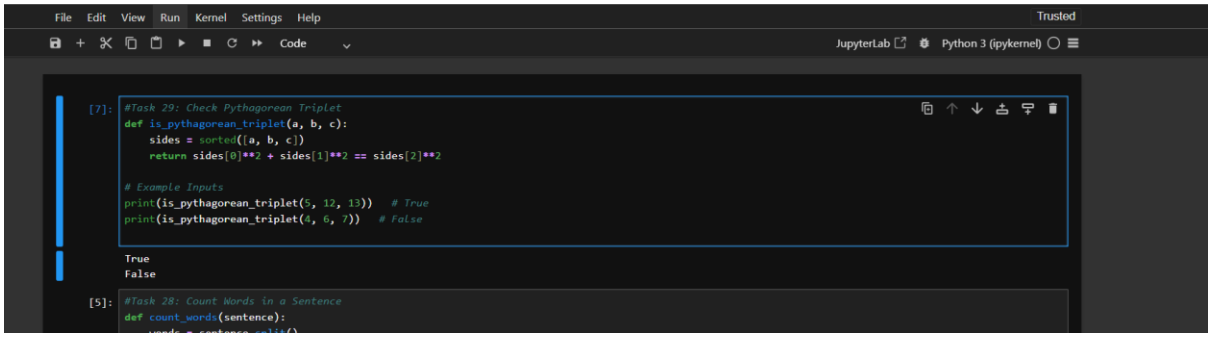
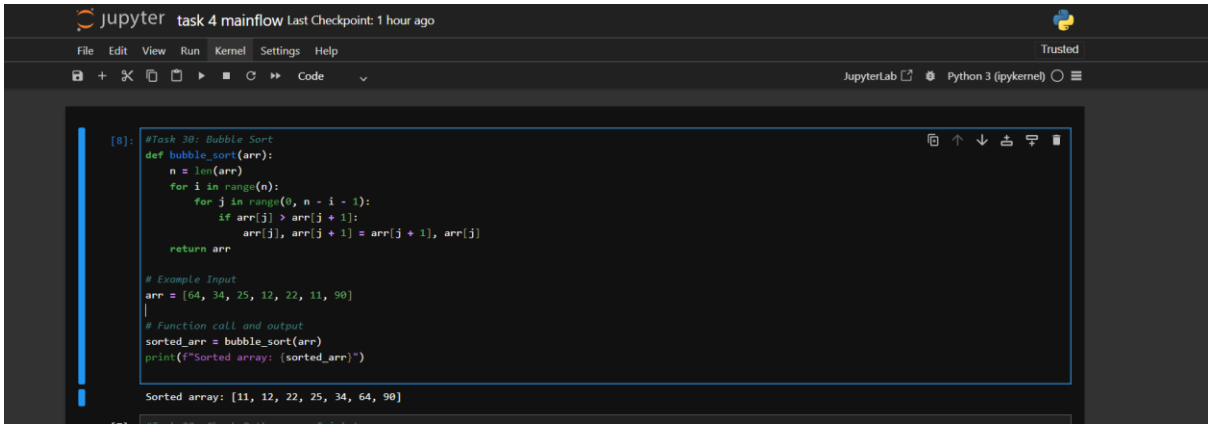

jupyter task 4 mainflow Last Checkpoint: 46 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[2]:
#Task 25: Find Missing Number
def find_missing_number(arr):
n = len(arr) + 1 # Since one number is missing from 1 to n+1
expected_sum = n * (n + 1) // 2
actual_sum = sum(arr)
return expected_sum - actual_sum
# Example Input
arr = [1, 2, 4, 5, 6]
# Function call and output
missing = find_missing_number(arr)
print(f"The missing number is: {missing}")
The missing number is: 3
[ ]:

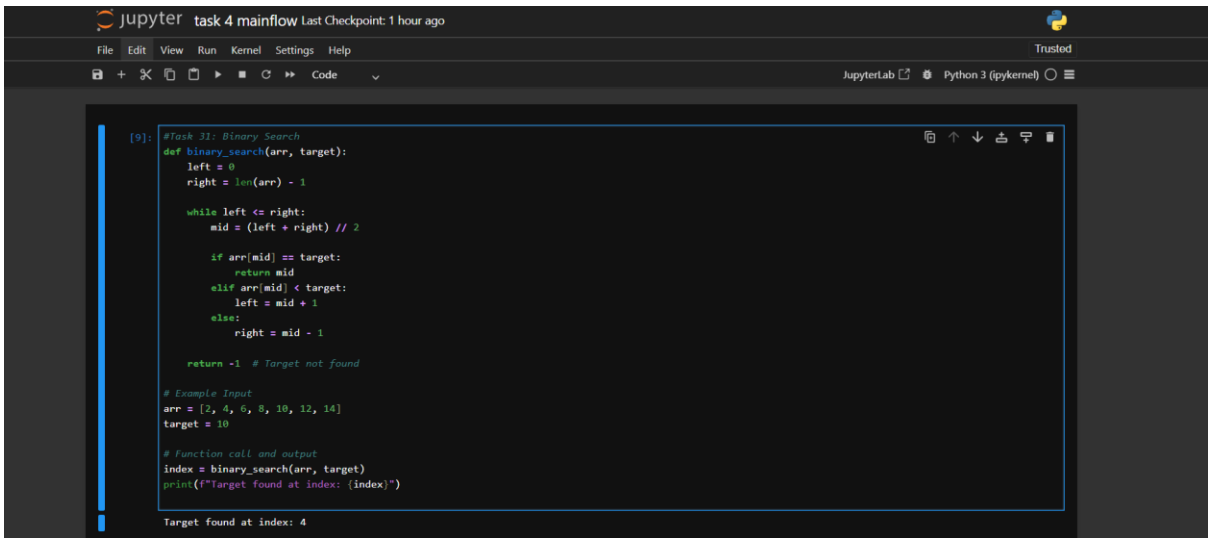

jupyter task 4 mainflow Last Checkpoint: 50 minutes ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[3]:
#Task 26: Check Balanced Parentheses
def is_balanced(expression):
stack = []
mapping = {')': '(', '}': '{', ']': '['}
for char in expression:
if char in mapping.values(): # opening brackets
stack.append(char)
elif char in mapping: # closing brackets
if not stack or stack[-1] != mapping[char]:
return False
stack.pop()
return not stack # True if stack is empty (balanced)
# Example Inputs
expr1 = "{[()]}" # Balanced
expr2 = "{[())}" # Not balanced
# Outputs
print(is_balanced(expr1)) # True
print(is_balanced(expr2)) # False
True
False

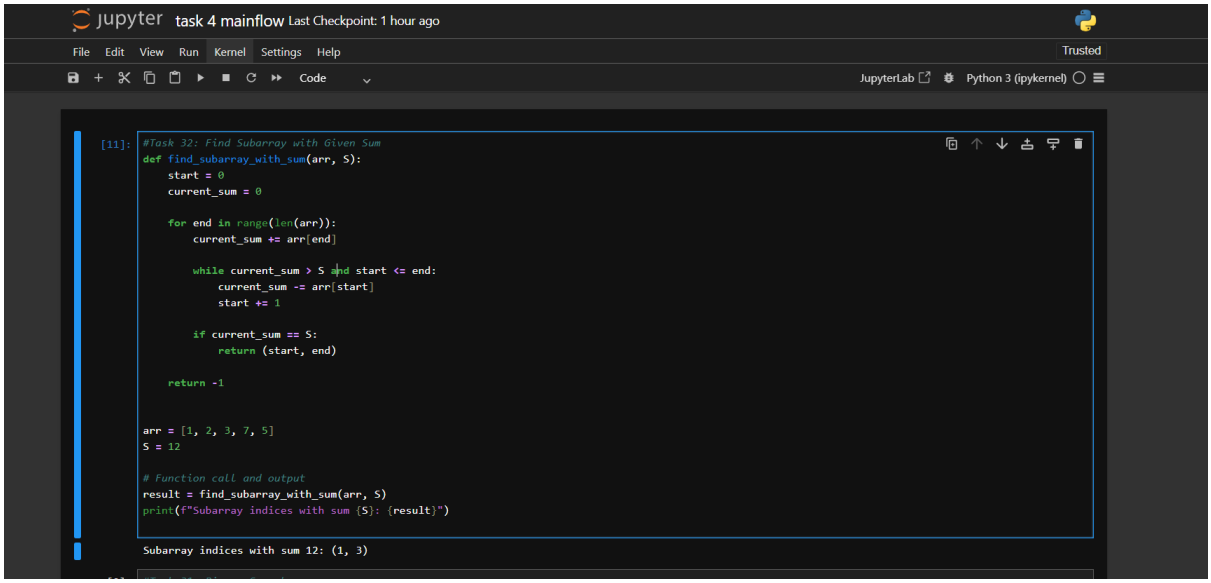

File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[4]:
#Task 27: Longest Word in a Sentence
def longest_word(sentence):
words = sentence.split()
longest = max(words, key=len)
return longest
# Example Input
sentence = "this is none of your business"
result = longest_word(sentence)
print(f"The longest word is: {result}")
The longest word is: business

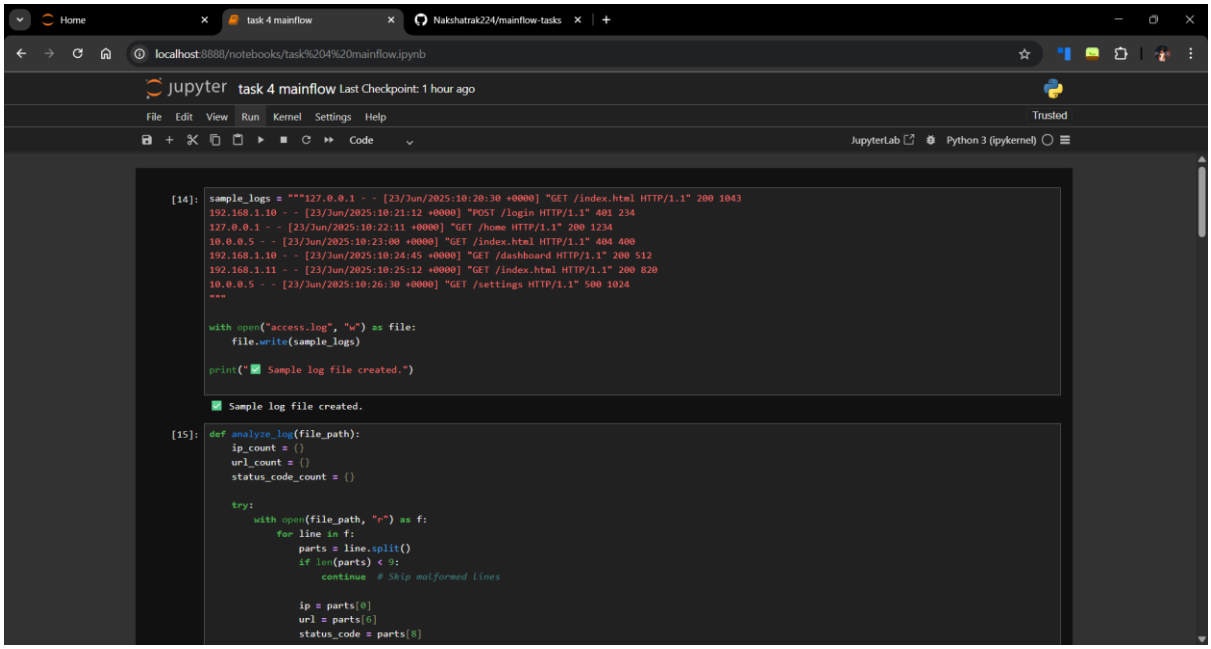

File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[5]:
#Task 28: Count Words in a Sentence
def count_words(sentence):
words = sentence.split()
return len(words)
# Example Input
sentence = "Python is a powerful programming language"
# Function call and output
result = count_words(sentence)
print(f"Word count: {result}")
Word count: 6

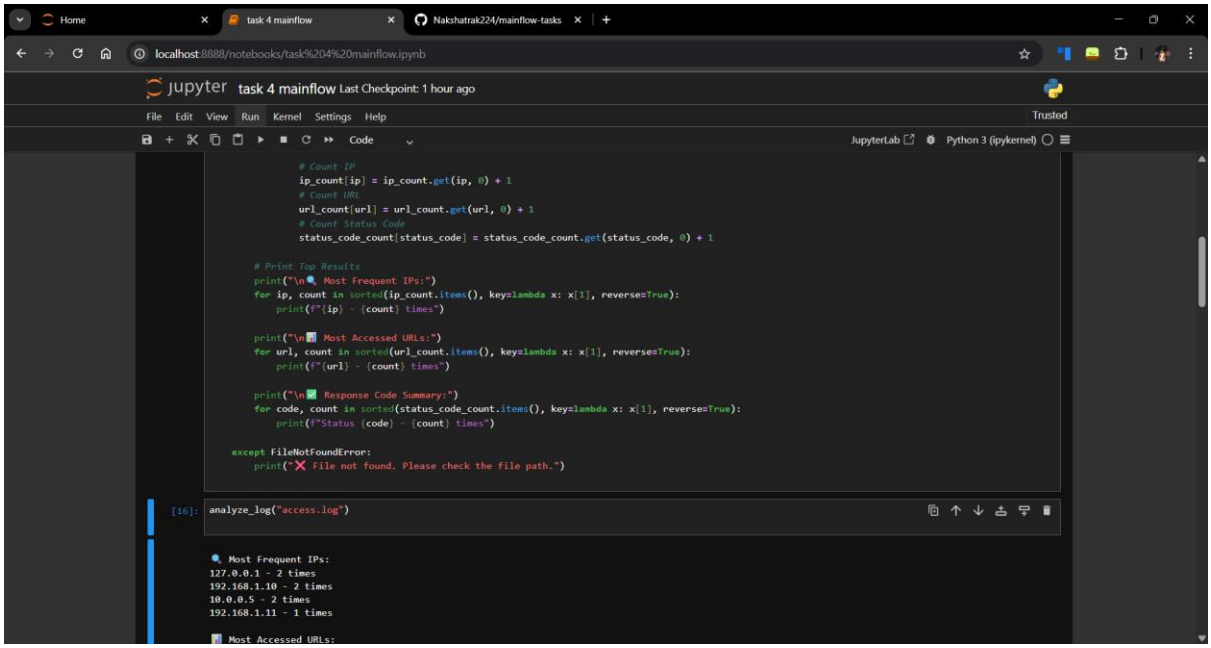

File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[7]:
#Task 29: Check Pythagorean Triplet
def is_pythagorean_triplet(a, b, c):
sides = sorted([a, b, c])
return sides[0]**2 + sides[1]**2 == sides[2]**2
Example Inputs
print(is_pythagorean_triplet(5, 12, 13)) # True
print(is_pythagorean_triplet(4, 6, 7)) # False
True
False
[5]:
#Task 28: Count Words in a Sentence
def count_words(sentence):

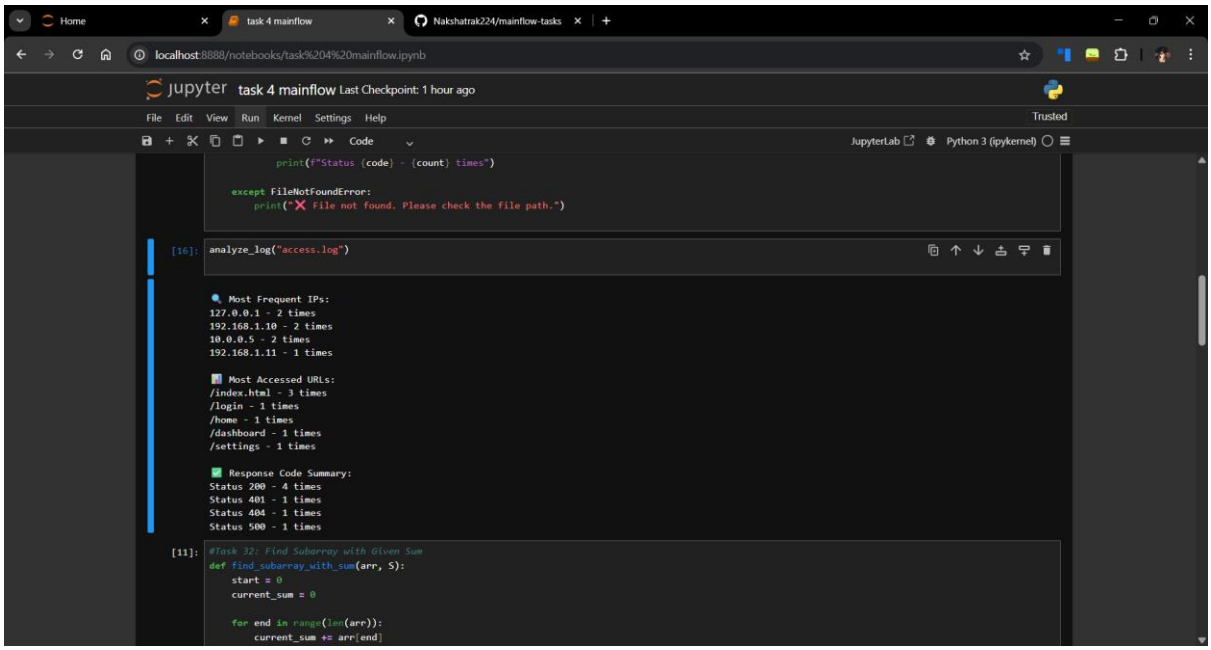

jupyter task 4 mainflow Last Checkpoint: 1 hour ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[8]:
#Task 30: Bubble Sort
def bubble_sort(arr):
n = len(arr)
for i in range(n):
for j in range(0, n - i - 1):
if arr[j] > arr[j + 1]:
arr[j], arr[j + 1] = arr[j + 1], arr[j]
return arr
# Example Input
arr = [64, 34, 25, 12, 22, 11, 90]
# Function call and output
sorted_arr = bubble_sort(arr)
print(f"Sorted array: {sorted_arr}")
Sorted array: [11, 12, 22, 25, 34, 64, 90]


jupyter task 4 mainflow Last Checkpoint: 1 hour ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[9]:
#Task 31: Binary Search
def binary_search(arr, target):
left = 0
right = len(arr) - 1
while left <= right:
mid = (left + right) // 2
if arr[mid] == target:
return mid
elif arr[mid] < target:
left = mid + 1
else:
right = mid - 1
return -1 # Target not found
# Example Input
arr = [2, 4, 6, 8, 10, 12, 14]
target = 10
# Function call and output
index = binary_search(arr, target)
print(f"Target found at index: {index}")
Target found at index: 4


jupyter task 4 mainflow Last Checkpoint: 1 hour ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[11]:
#Task 32: Find Subarray with Given Sum
def find_subarray_with_sum(arr, S):
start = 0
current_sum = 0
for end in range(len(arr)):
current_sum += arr[end]
while current_sum > S and start <= end:
current_sum -= arr[start]
start += 1
if current_sum == S:
return (start, end)
return -1
arr = [1, 2, 3, 7, 5]
S = 12
# Function call and output
result = find_subarray_with_sum(arr, S)
print(f"Subarray indices with sum {S}: {result}")
Subarray indices with sum 12: (1, 3)


Home
task 4 mainflow
Nakshatrak224/mainflow-tasks
localhost:8888/notebooks/task%204%20mainflow.ipynb
jupyter task 4 mainflow Last Checkpoint: 1 hour ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
[14]:
sample_logs = """127.0.0.1 - - [23/Jun/2025:10:20:30 +0000] "GET /index.html HTTP/1.1" 200 1043
192.168.1.10 - - [23/Jun/2025:10:21:12 +0000] "POST /login HTTP/1.1" 401 234
127.0.0.1 - - [23/Jun/2025:10:22:11 +0000] "GET /home HTTP/1.1" 200 1234
10.0.0.5 - - [23/Jun/2025:10:23:00 +0000] "GET /index.html HTTP/1.1" 404 400
192.168.1.10 - - [23/Jun/2025:10:24:45 +0000] "GET /dashboard HTTP/1.1" 200 512
192.168.1.11 - - [23/Jun/2025:10:25:12 +0000] "GET /index.html HTTP/1.1" 200 820
10.0.0.5 - - [23/Jun/2025:10:26:30 +0000] "GET /settings HTTP/1.1" 500 1024
"""
with open("access.log", "w") as file:
file.write(sample_logs)
print("✅ Sample log file created.")
✅ Sample log file created.
[15]:
def analyze_log(file_path):
ip_count = {}
url_count = {}
status_code_count = {}
try:
with open(file_path, "r") as f:
for line in f:
parts = line.split()
if len(parts) < 9:
continue # Skip malformed lines
ip = parts[0]
url = parts[6]
status_code = parts[8]


Home
task 4 mainflow
Nakshatrak224/mainflow-tasks
localhost:8888/notebooks/task%204%20mainflow.ipynb
jupyter task 4 mainflow Last Checkpoint: 1 hour ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
# Count IP
ip_count[ip] = ip_count.get(ip, 0) + 1
# Count URL
url_count[url] = url_count.get(url, 0) + 1
# Count Status Code
status_code_count[status_code] = status_code_count.get(status_code, 0) + 1
# Print Top Results
print("\n🔍 Most Frequent IPs:")
for ip, count in sorted(ip_count.items(), key=lambda x: x[1], reverse=True):
print(f"{ip} - {count} times")
print("\n📊 Most Accessed URLs:")
for url, count in sorted(url_count.items(), key=lambda x: x[1], reverse=True):
print(f"{url} - {count} times")
print("\n✅ Response Code Summary:")
for code, count in sorted(status_code_count.items(), key=lambda x: x[1], reverse=True):
print(f"Status {code} - {count} times")
except FileNotFoundError:
print("❌ File not found. Please check the file path.")
[16]:
analyze_log("access.log")
🔍 Most Frequent IPs:
127.0.0.1 - 2 times
192.168.1.10 - 2 times
10.0.0.5 - 2 times
192.168.1.11 - 1 times
📊 Most Accessed URLs:


Home
task 4 mainflow
Nakshatrak224/mainflow-tasks
localhost:8888/notebooks/task%204%20mainflow.ipynb
jupyter task 4 mainflow Last Checkpoint: 1 hour ago
File Edit View Run Kernel Settings Help
Trusted
Code
JupyterLab
Python 3 (ipykernel)
print(f"Status {code} - {count} times")
except FileNotFoundError:
print("❌ File not found. Please check the file path.")
[16]:
analyze_log("access.log")
🔍 Most Frequent IPs:
127.0.0.1 - 2 times
192.168.1.10 - 2 times
10.0.0.5 - 2 times
192.168.1.11 - 1 times
📊 Most Accessed URLs:
/index.html - 3 times
/login - 1 times
/home - 1 times
/dashboard - 1 times
/settings - 1 times
✅ Response Code Summary:
Status 200 - 4 times
Status 401 - 1 times
Status 404 - 1 times
Status 500 - 1 times
[11]:
#Task 32: Find Subarray with Given Sum
def find_subarray_with_sum(arr, S):
start = 0
current_sum = 0
for end in range(len(arr)):
current_sum += arr[end]