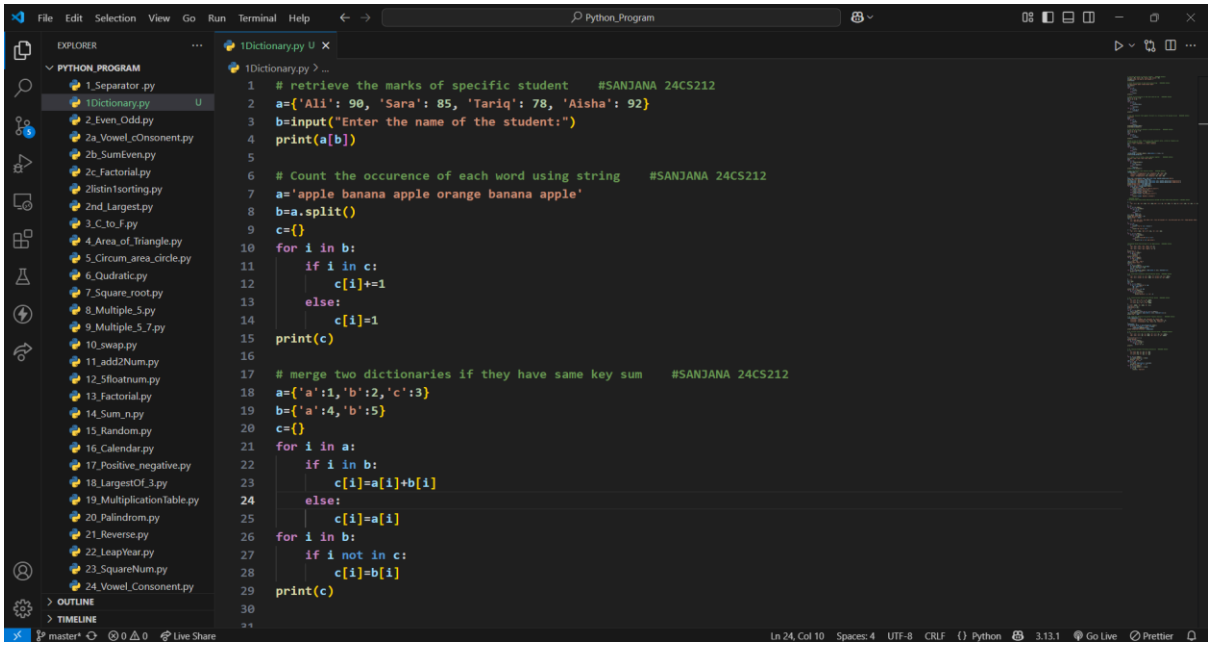
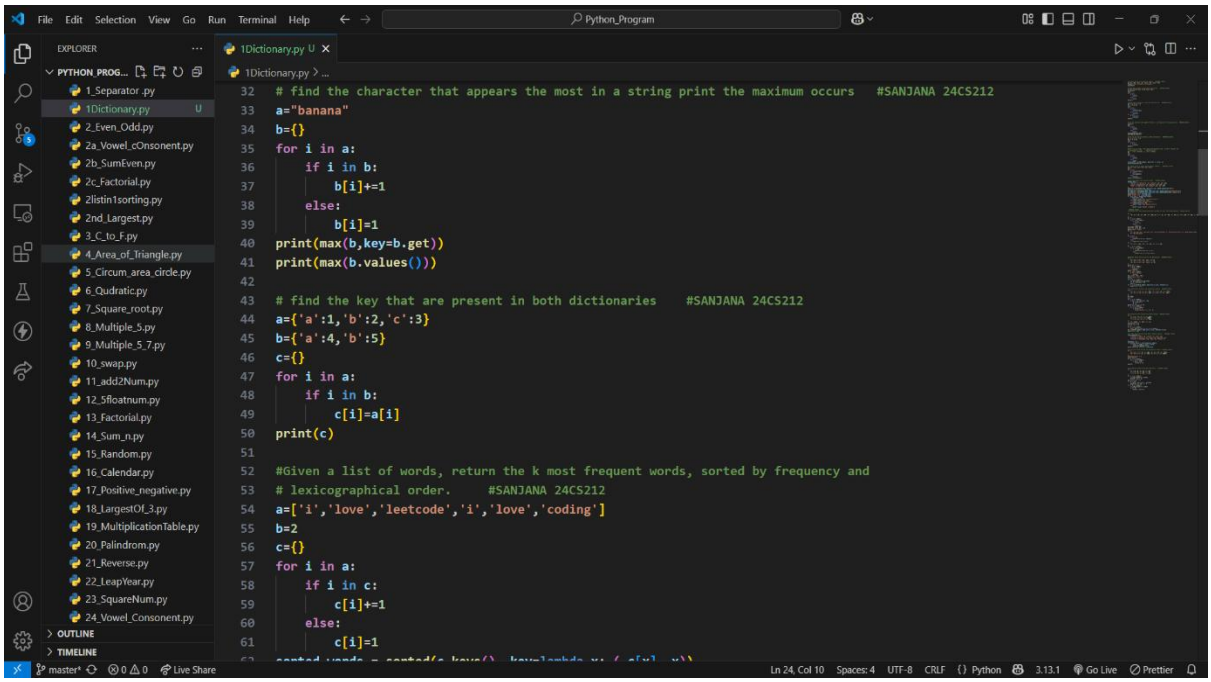
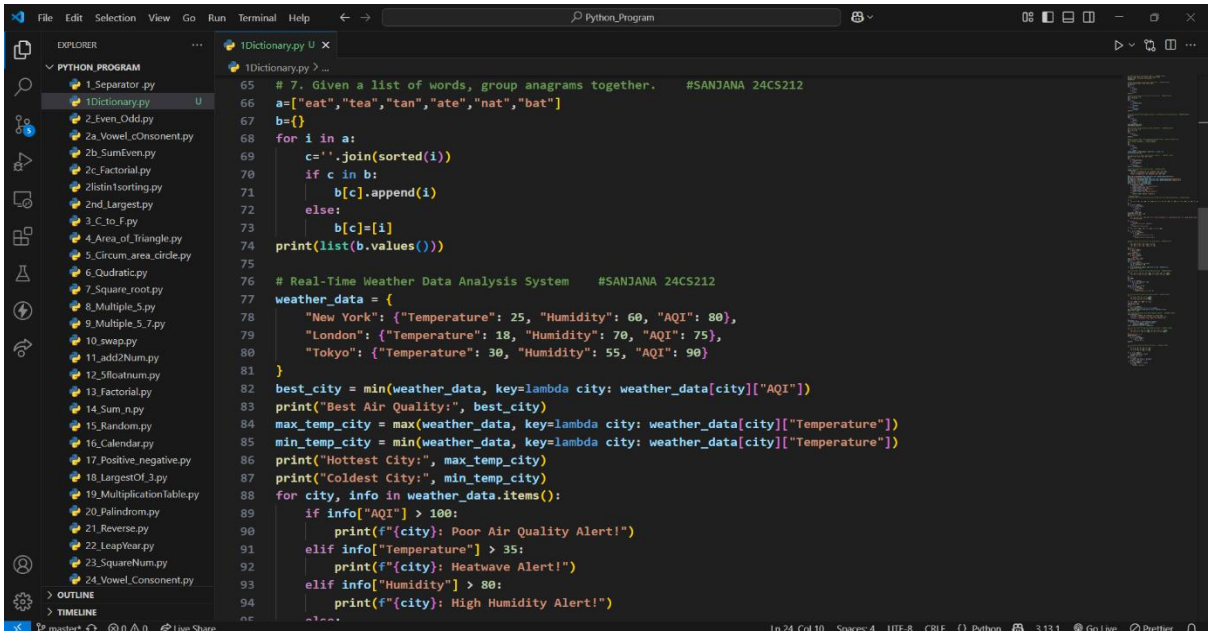
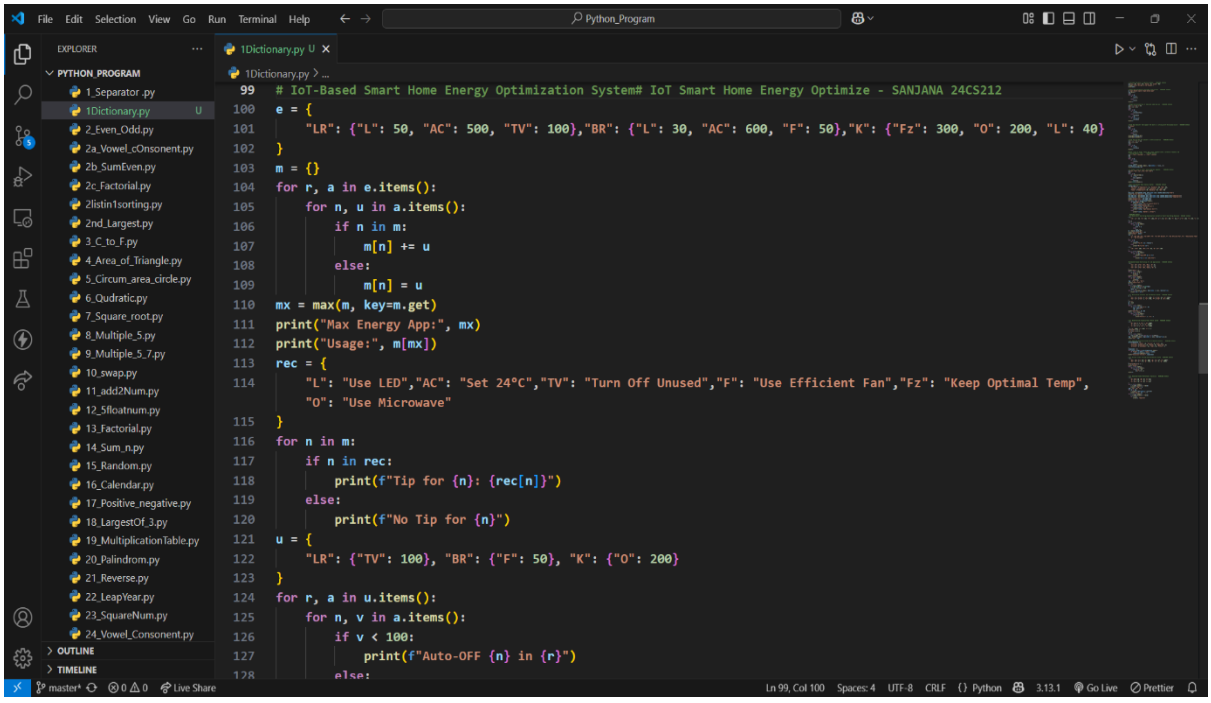

File Edit Selection View Go Run Terminal Help
Python_Program
EXPLORER
PYTHON_PROGRAM
1Dictionary.py U
1_Separator .py
1Dictionary.py U
2_Even_Odd.py
2a_Vowel_cOnsonent.py
2b_SumEven.py
2c_Factorial.py
2listin1sorting.py
2nd_Largest.py
3_C_to_F.py
4_Area_of_Triangle.py
5_Circum_area_circle.py
6_Qudratic.py
7_Square_root.py
8_Multiple_5.py
9_Multiple_5_7.py
10_swap.py
11_add2Num.py
12_5floatnum.py
13_Factorial.py
14_Sum_n.py
15_Random.py
16_Calendar.py
17_Positive_negative.py
18_LargestOf_3.py
19_MultiplicationTable.py
20_Palindrom.py
21_Reverse.py
22_LeapYear.py
23_SquareNum.py
24_Vowel_Consonent.py
OUTLINE
TIMELINE
1Dictionary.py U
1Dictionary.py > ...
1 # retrieve the marks of specific student #SANJANA 24CS212
2 a={'Ali': 90, 'Sara': 85, 'Tariq': 78, 'Aisha': 92}
3 b=input("Enter the name of the student:")
4 print(a[b])
5
6 # Count the occurence of each word using string #SANJANA 24CS212
7 a='apple banana apple orange banana apple'
8 b=a.split()
9 c={}
10 for i in b:
11 if i in c:
12 c[i]+=1
13 else:
14 c[i]=1
15 print(c)
16
17 # merge two dictionaries if they have same key sum #SANJANA 24CS212
18 a={'a':1,'b':2,'c':3}
19 b={'a':4,'b':5}
20 c={}
21 for i in a:
22 if i in b:
23 c[i]=a[i]+b[i]
24 else:
25 c[i]=a[i]
26 for i in b:
27 if i not in c:
28 c[i]=b[i]
29 print(c)
30
master* 0 0 Live Share
Ln 24, Col 10 Spaces: 4 UTF-8 CRLF {} Python 3.13.1 Go Live Prettier

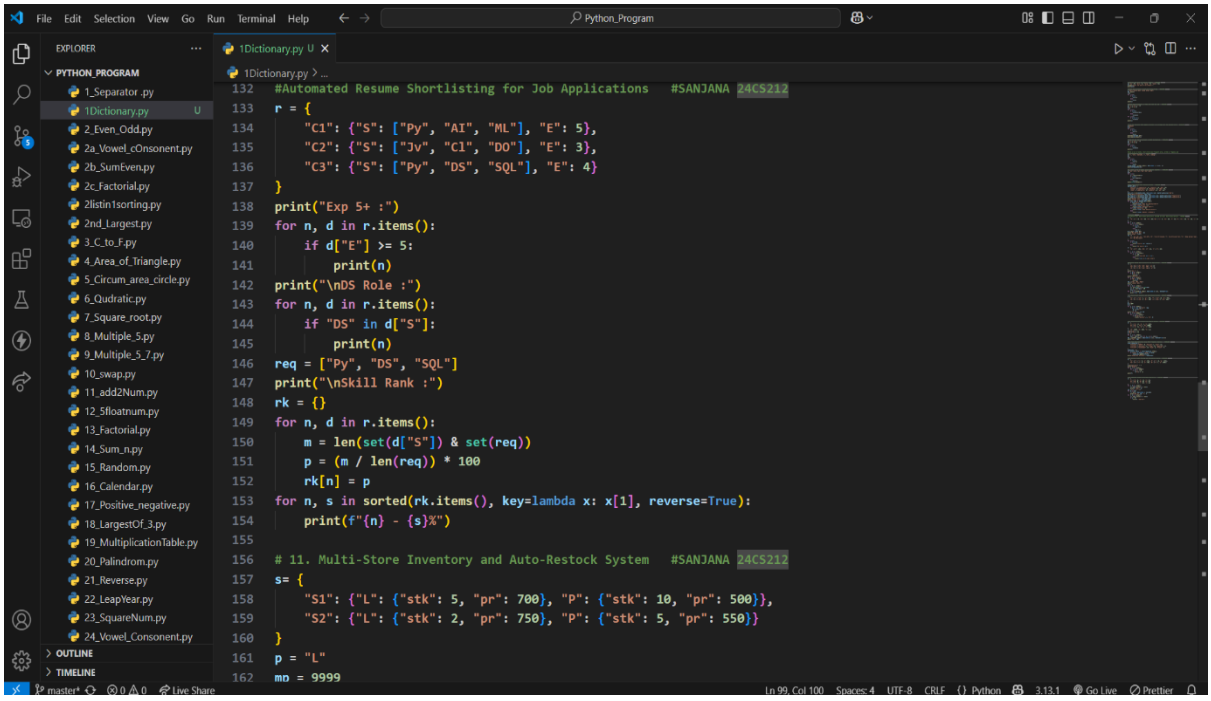

File Edit Selection View Go Run Terminal Help
Python_Program
EXPLORER
PYTHON_PROG...
1_Separator .py
1Dictionary.py U
2_Even_Odd.py
2a_Vowel_cOnsonent.py
2b_SumEven.py
2c_Factorial.py
2listin1sorting.py
2nd_Largest.py
3_C_to_F.py
4_Area_of_Triangle.py
5_Circum_area_circle.py
6_Qudratic.py
7_Square_root.py
8_Multiple_5.py
9_Multiple_5_7.py
10_swap.py
11_add2Num.py
12_5floatnum.py
13_Factorial.py
14_Sum_n.py
15_Random.py
16_Calendar.py
17_Positive_negative.py
18_LargestOf_3.py
19_MultiplicationTable.py
20_Palindrom.py
21_Reverse.py
22_LeapYear.py
23_SquareNum.py
24_Vowel_Consonent.py
OUTLINE
TIMELINE
1Dictionary.py U
1Dictionary.py > ...
32 # find the character that appears the most in a string print the maximum occurs #SANJANA 24CS212
33 a="banana"
34 b={}
35 for i in a:
36 if i in b:
37 b[i]+=1
38 else:
39 b[i]=1
40 print(max(b,key=b.get))
41 print(max(b.values()))
42
43 # find the key that are present in both dictionaries #SANJANA 24CS212
44 a={'a':1,'b':2,'c':3}
45 b={'a':4,'b':5}
46 c={}
47 for i in a:
48 if i in b:
49 c[i]=a[i]
50 print(c)
51
52 #Given a list of words, return the k most frequent words, sorted by frequency and
53 # lexicographical order. #SANJANA 24CS212
54 a=['i','love','leetcode','i','love','coding']
55 b=2
56 c={}
57 for i in a:
58 if i in c:
59 c[i]+=1
60 else:
61 c[i]=1
master* 0 0 Live Share
Ln 24, Col 10 Spaces: 4 UTF-8 CRLF {} Python 3.13.1 Go Live Prettier



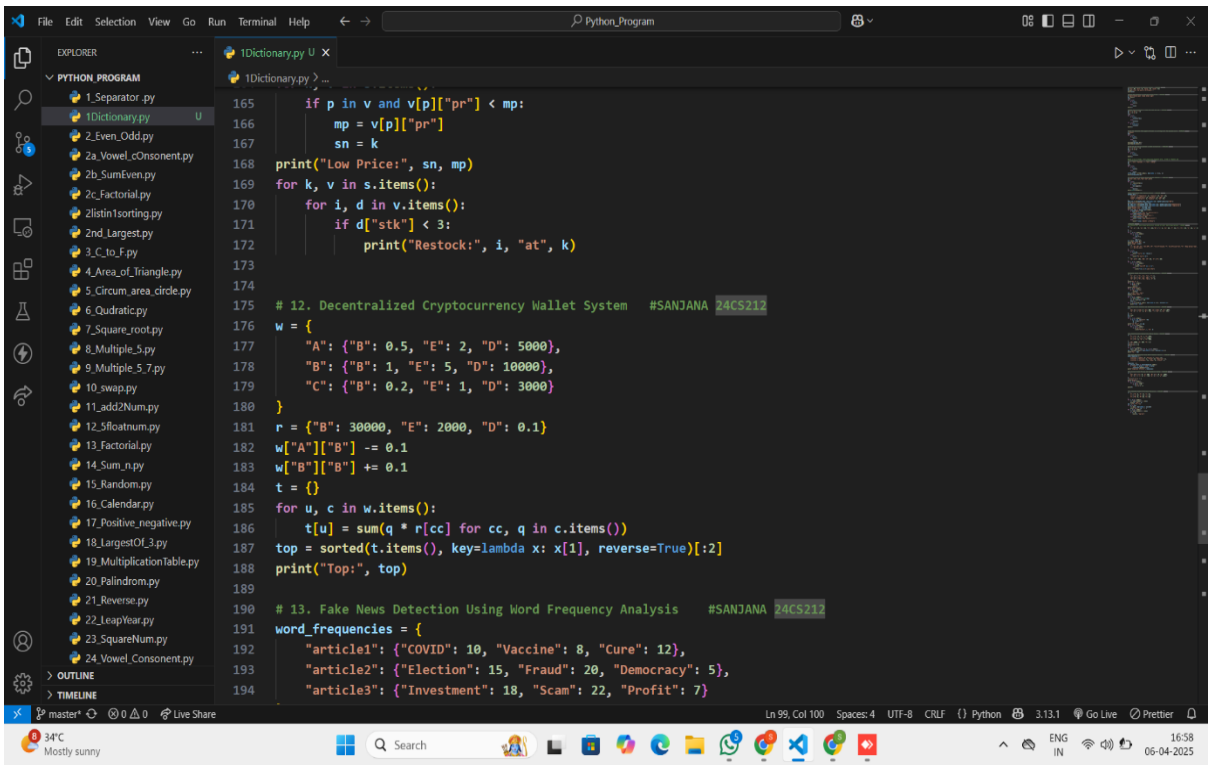
```python
# 7. Given a list of words, group anagrams together.    #SANJANA 24CS212
a=["eat","tea","tan","ate","nat","bat"]
b={}
for i in a:
    c=''.join(sorted(i))
    if c in b:
        b[c].append(i)
    else:
        b[c]=[i]
print(list(b.values()))

# Real-Time Weather Data Analysis System    #SANJANA 24CS212
weather_data = {
    "New York": {"Temperature": 25, "Humidity": 60, "AQI": 80},
    "London": {"Temperature": 18, "Humidity": 70, "AQI": 75},
    "Tokyo": {"Temperature": 30, "Humidity": 55, "AQI": 90}
}
best_city = min(weather_data, key=lambda city: weather_data[city]["AQI"])
print("Best Air Quality:", best_city)
max_temp_city = max(weather_data, key=lambda city: weather_data[city]["Temperature"])
min_temp_city = min(weather_data, key=lambda city: weather_data[city]["Temperature"])
print("Hottest City:", max_temp_city)
print("Coldest City:", min_temp_city)
for city, info in weather_data.items():
    if info["AQI"] > 100:
        print(f"{city}: Poor Air Quality Alert!")
    elif info["Temperature"] > 35:
        print(f"{city}: Heatwave Alert!")
    elif info["Humidity"] > 80:
        print(f"{city}: High Humidity Alert!")
```



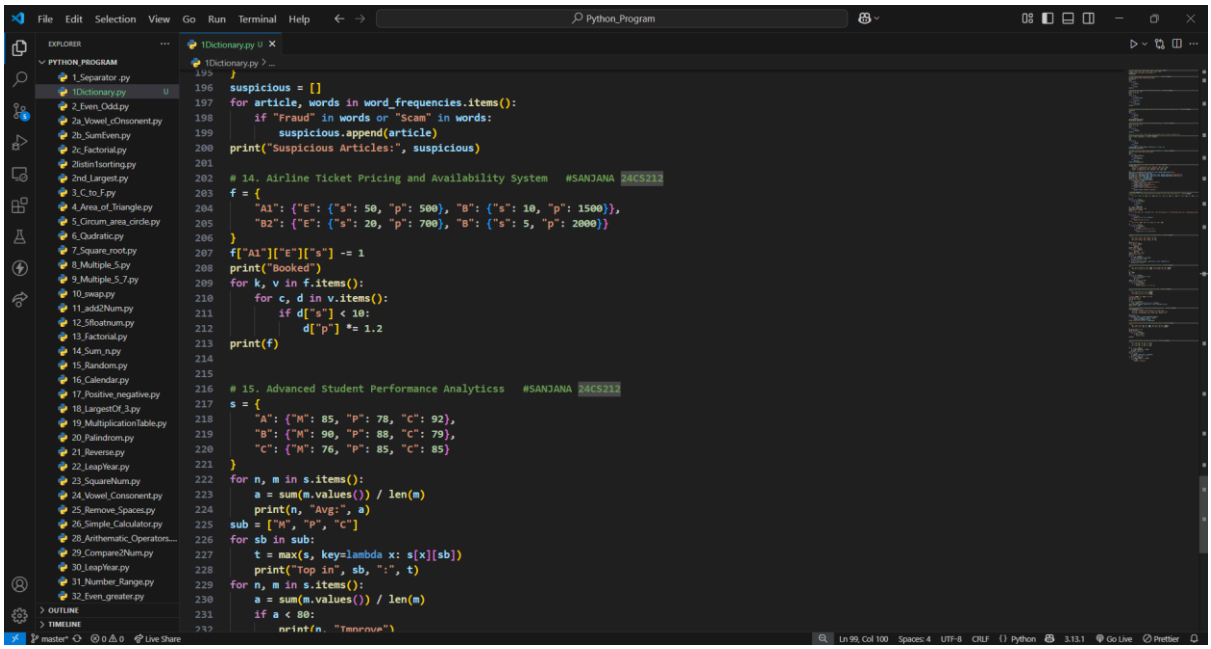
```python
# IoT-Based Smart Home Energy Optimization System# IoT Smart Home Energy Optimize - SANJANA 24CS212
e = {
    "LR": {"L": 50, "AC": 500, "TV": 100},"BR": {"L": 30, "AC": 600, "F": 50},"K": {"Fz": 300, "O": 200, "L": 40}
}
m = {}
for r, a in e.items():
    for n, u in a.items():
        if n in m:
            m[n] += u
        else:
            m[n] = u
mx = max(m, key=m.get)
print("Max Energy App:", mx)
print("Usage:", m[mx])
rec = {
    "L": "Use LED","AC": "Set 24°C","TV": "Turn Off Unused","F": "Use Efficient Fan","Fz": "Keep Optimal Temp",
    "O": "Use Microwave"
}
for n in m:
    if n in rec:
        print(f"Tip for {n}: {rec[n]}")
    else:
        print(f"No Tip for {n}")
u = {
    "LR": {"TV": 100}, "BR": {"F": 50}, "K": {"O": 200}
}
for r, a in u.items():
    for n, v in a.items():
        if v < 100:
            print(f"Auto-OFF {n} in {r}")
        else:
```



```python
#Automated Resume Shortlisting for Job Applications   #SANJANA 24CS212
r = {
    "C1": {"S": ["Py", "AI", "ML"], "E": 5},
    "C2": {"S": ["Jv", "Cl", "DO"], "E": 3},
    "C3": {"S": ["Py", "DS", "SQL"], "E": 4}
}
print("Exp 5+ :")
for n, d in r.items():
    if d["E"] >= 5:
        print(n)
print("\nDS Role :")
for n, d in r.items():
    if "DS" in d["S"]:
        print(n)
req = ["Py", "DS", "SQL"]
print("\nSkill Rank :")
rk = {}
for n, d in r.items():
    m = len(set(d["S"]) & set(req))
    p = (m / len(req)) * 100
    rk[n] = p
for n, s in sorted(rk.items(), key=lambda x: x[1], reverse=True):
    print(f"{n} - {s}%")

# 11. Multi-Store Inventory and Auto-Restock System   #SANJANA 24CS212
s= {
    "S1": {"L": {"stk": 5, "pr": 700}, "P": {"stk": 10, "pr": 500}},
    "S2": {"L": {"stk": 2, "pr": 750}, "P": {"stk": 5, "pr": 550}}
}
p = "L"
mp = 9999
```



```python
    if p in v and v[p]["pr"] < mp:
        mp = v[p]["pr"]
        sn = k
print("Low Price:", sn, mp)
for k, v in s.items():
    for i, d in v.items():
        if d["stk"] < 3:
            print("Restock:", i, "at", k)


# 12. Decentralized Cryptocurrency Wallet System   #SANJANA 24CS212
w = {
    "A": {"B": 0.5, "E": 2, "D": 5000},
    "B": {"B": 1, "E": 5, "D": 10000},
    "C": {"B": 0.2, "E": 1, "D": 3000}
}
r = {"B": 30000, "E": 2000, "D": 0.1}
w["A"]["B"] -= 0.1
w["B"]["B"] += 0.1
t = {}
for u, c in w.items():
    t[u] = sum(q * r[cc] for cc, q in c.items())
top = sorted(t.items(), key=lambda x: x[1], reverse=True)[:2]
print("Top:", top)

# 13. Fake News Detection Using Word Frequency Analysis    #SANJANA 24CS212
word_frequencies = {
    "article1": {"COVID": 10, "Vaccine": 8, "Cure": 12},
    "article2": {"Election": 15, "Fraud": 20, "Democracy": 5},
    "article3": {"Investment": 18, "Scam": 22, "Profit": 7}
```


File Edit Selection View Go Run Terminal Help
Python_Program
EXPLORER
PYTHON_PROGRAM
1_Separator .py
1Dictionary.py
2_Even_Odd.py
2a_Vowel_cOnsonent.py
2b_SumEven.py
2c_Factorial.py
2listin1sorting.py
2nd_Largest.py
3_C_to_F.py
4_Area_of_Triangle.py
5_Circum_area_circle.py
6_Qudratic.py
7_Square_root.py
8_Multiple_5.py
9_Multiple_5_7.py
10_swap.py
11_add2Num.py
12_5floatnum.py
13_Factorial.py
14_Sum_n.py
15_Random.py
16_Calendar.py
17_Positive_negative.py
18_LargestOf_3.py
19_MultiplicationTable.py
20_Palindrom.py
21_Reverse.py
22_LeapYear.py
23_SquareNum.py
24_Vowel_Consonent.py
25_Remove_Spaces.py
26_Simple_Calculator.py
28_Arithematic_Operators....
29_Compare2Num.py
30_LeapYear.py
31_Number_Range.py
32_Even_greater.py
OUTLINE
TIMELINE
1Dictionary.py U
1Dictionary.py > ...
}
suspicious = []
for article, words in word_frequencies.items():
if "Fraud" in words or "Scam" in words:
suspicious.append(article)
print("Suspicious Articles:", suspicious)
# 14. Airline Ticket Pricing and Availability System #SANJANA 24CS212
f = {
"A1": {"E": {"s": 50, "p": 500}, "B": {"s": 10, "p": 1500}},
"B2": {"E": {"s": 20, "p": 700}, "B": {"s": 5, "p": 2000}}
}
f["A1"]["E"]["s"] -= 1
print("Booked")
for k, v in f.items():
for c, d in v.items():
if d["s"] < 10:
d["p"] *= 1.2
print(f)
# 15. Advanced Student Performance Analyticss #SANJANA 24CS212
s = {
"A": {"M": 85, "P": 78, "C": 92},
"B": {"M": 90, "P": 88, "C": 79},
"C": {"M": 76, "P": 85, "C": 85}
}
for n, m in s.items():
a = sum(m.values()) / len(m)
print(n, "Avg:", a)
sub = ["M", "P", "C"]
for sb in sub:
t = max(s, key=lambda x: s[x][sb])
print("Top in", sb, ":", t)
for n, m in s.items():
a = sum(m.values()) / len(m)
if a < 80:
master*
Live Share
Ln 99, Col 100
Spaces: 4
UTF-8
CRLF
Python
3.13.1
Go Live
Prettier