



الجمهورية العربية السورية

جامعة تشرين

كلية الهندسة الميكانيكية والكهربائية

قسم هندسة الاتصالات والإلكترونيات

السنة الخامسة: وظيفة 1 برمجة الشبكات

الاسم: زينب أحمد سعود

الرقم الجامعي : 2890

بإشراف: د. مهند عيسى

**Name: Zeinab Saoud, Number:2890, Submitted to
GitHub: zeinab122**

Quistion1:

A-

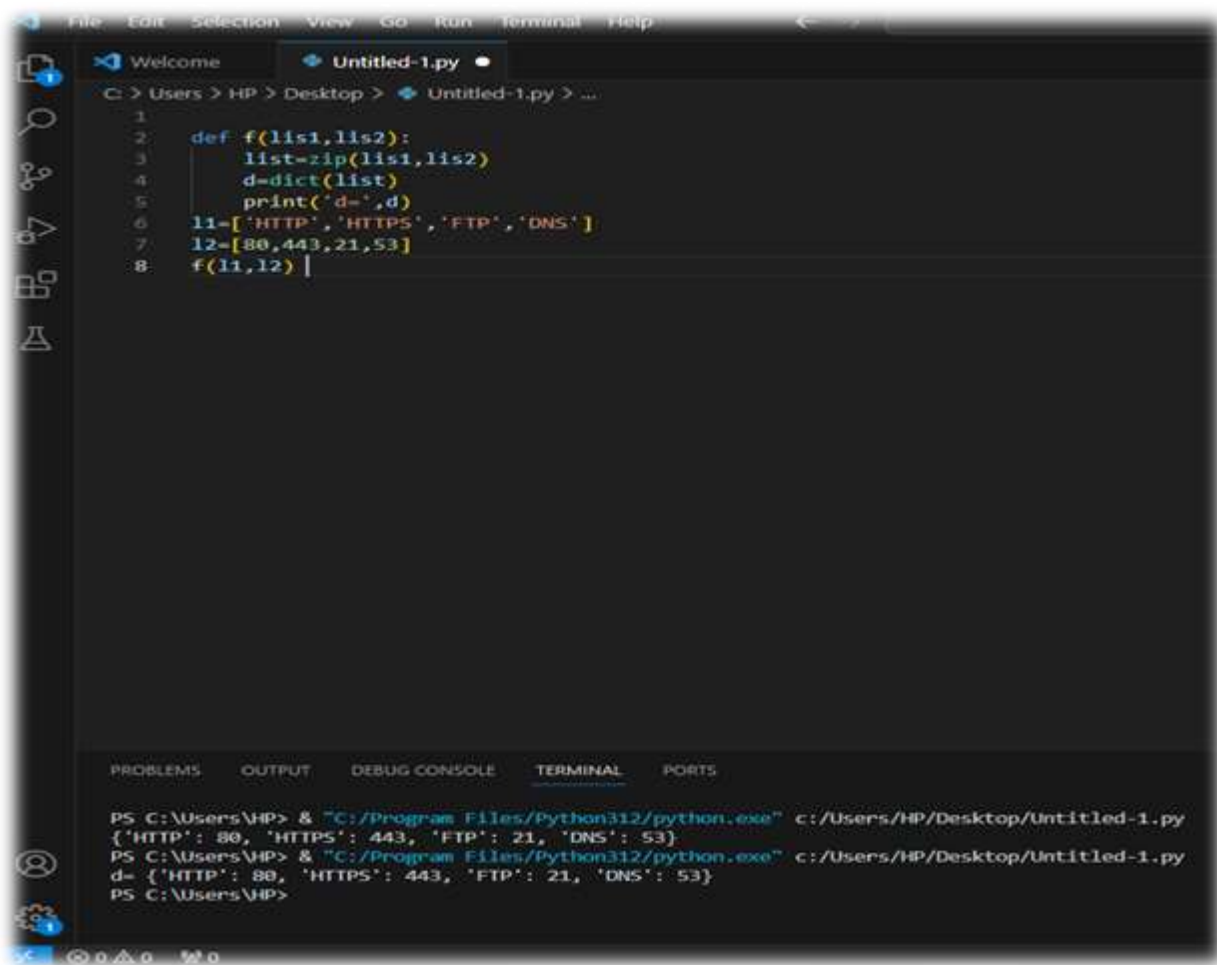
If you have two lists, L1=['HTTP','HTTPS','FTP','DNS']
L2=[80,443,21,53], convert it to generate this
dictionary d={'HTTP':80,'HTTPS':443,'FTP':21,'DNS':53 }

نريد بناء قاموس مفاتيحه عناصر القائمة الاولى L1 و قيمه عناصر القاصمة الثانية L2 : تم بناء تابع ببارامترين (قائمتين) يحوي على تابع zip الذي يجمع عناصر اكثر من قائمة و ذلك حسب ال index أي اول عنصر في القائمة الأولى مع اول عنصر في القائمة الثانية و هكذا... ثم استخدمت dict للتحويل الى قاموس بالشكل المطلوب و تم الانتهاء من بناء التابع

أخيرا تم تعريف القاصمتين المطلوبة و استدعاء التابع المبني سابقا للتنفذ و الطباعة

```
def f(lis1,lis2):  
    list=zip(lis1,lis2)  
    d=dict(list)  
    print('d=',d)  
l1=[ 'HTTP', 'HTTPS', 'FTP', 'DNS' ]  
l2=[ 80,443,21,53 ]  
f(l1,l2)
```

الخرج موضح في السكرين:



```
File Edit Selection View Go Run Terminal Help
Welcome
Untitled-1.py
C:\Users\HP\Desktop> Untitled-1.py
1
2 def f(lis1,lis2):
3     list=zip(lis1,lis2)
4     d=dict(list)
5     print('d=',d)
6 lis1=['HTTP','HTTPS','FTP','DNS']
7 lis2=[80,443,21,53]
8 f(lis1,lis2)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/Untitled-1.py
{'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/Untitled-1.py
d= {'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
PS C:\Users\HP>
```

B-

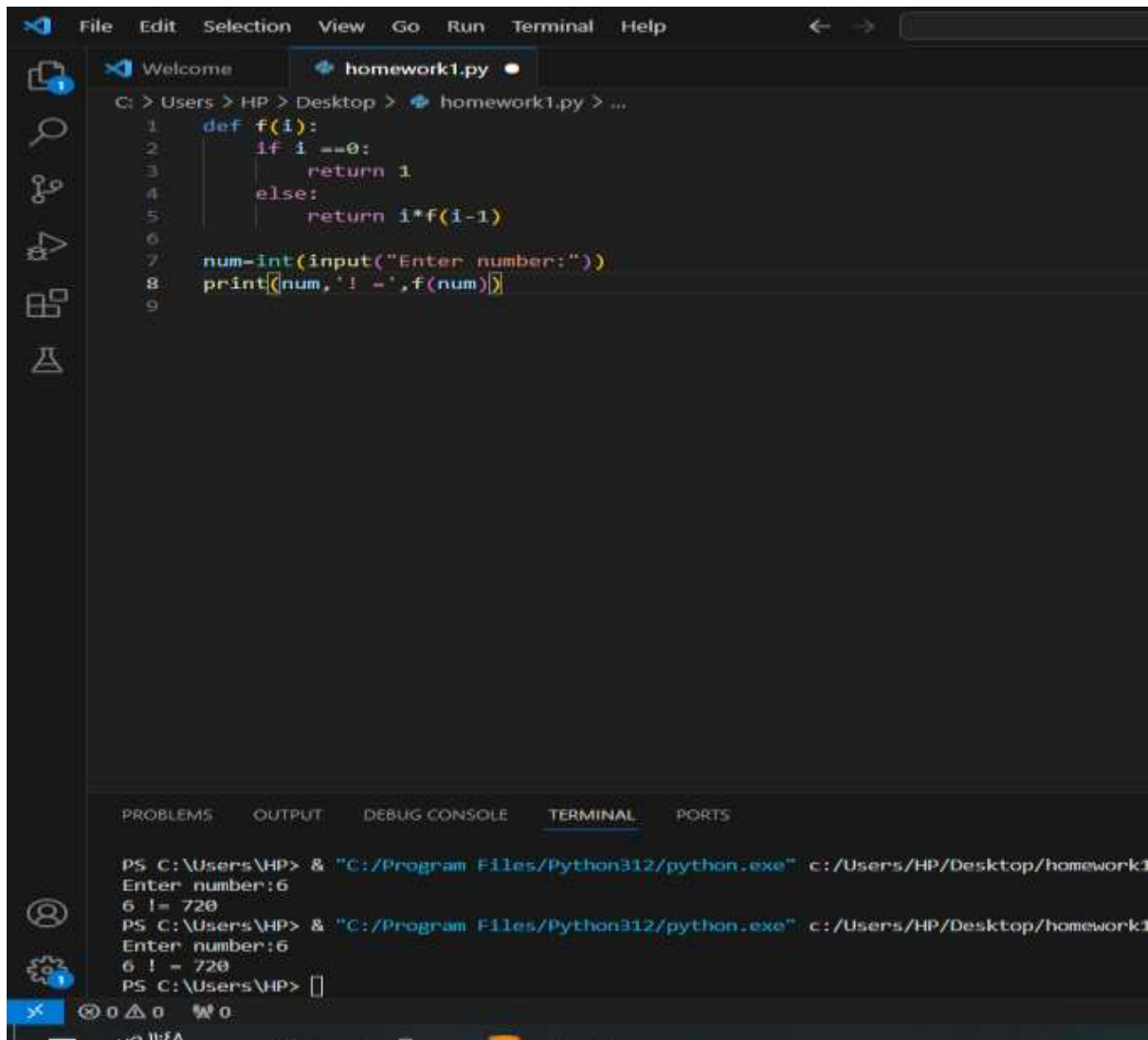
Write a Python program that calculates the factorial of a given number entered by user.

لإيجاد عاملي عدد ما لا يساوي الصفر هوي جداء ضرب العدد بكل الاعداد السابقة حتى 1 اما عاملي العدد صفر هو 1 و بالتالي تم بناء تابع ببارامتر واحد و هو العدد المراد إيجاد عامله اذا كان 0! يعيد الجواب 1 و الا يعيد جداء ضرب بكل ما سبقه وصولا الى 1 بعد الانتهاء من بناء التابع يطلب من المستخدم إدخال رقم ثم يتم استدعاء التابع لتنفيذ حساب العاملي عليه و طباعة الجواب للمستخدم

```
def f(i):
    if i ==0:
        return 1
    else:
        return i*f(i-1)
```

```
num=int(input("Enter number:"))
print(num,'! =',f(num))
```

الخرج موضح في السكرين وتم إدخال الرقم 6 كمثال:



The screenshot shows a Python IDE with a file named 'homework1.py'. The code defines a recursive function 'f(i)' that calculates the factorial of 'i'. It then prompts the user to enter a number and prints the result. The terminal output shows the program being run, the user entering '6', and the output '6 ! = 720'.

```
1 def f(i):
2     if i == 0:
3         return 1
4     else:
5         return i*f(i-1)
6
7 num=int(input("Enter number:"))
8 print(num,'! =',f(num))
9
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/homework1.py
Enter number:6
6 ! = 720
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/homework1.py
Enter number:6
6 ! = 720
PS C:\Users\HP>
```

C-

L=['Network' , 'Bio' , 'Programming' , 'Physics' , 'Music']

In this exercise, you will implement a Python program that reads the items of the previous list and identifies

the items that starts with 'B' letter, then print it on screen.

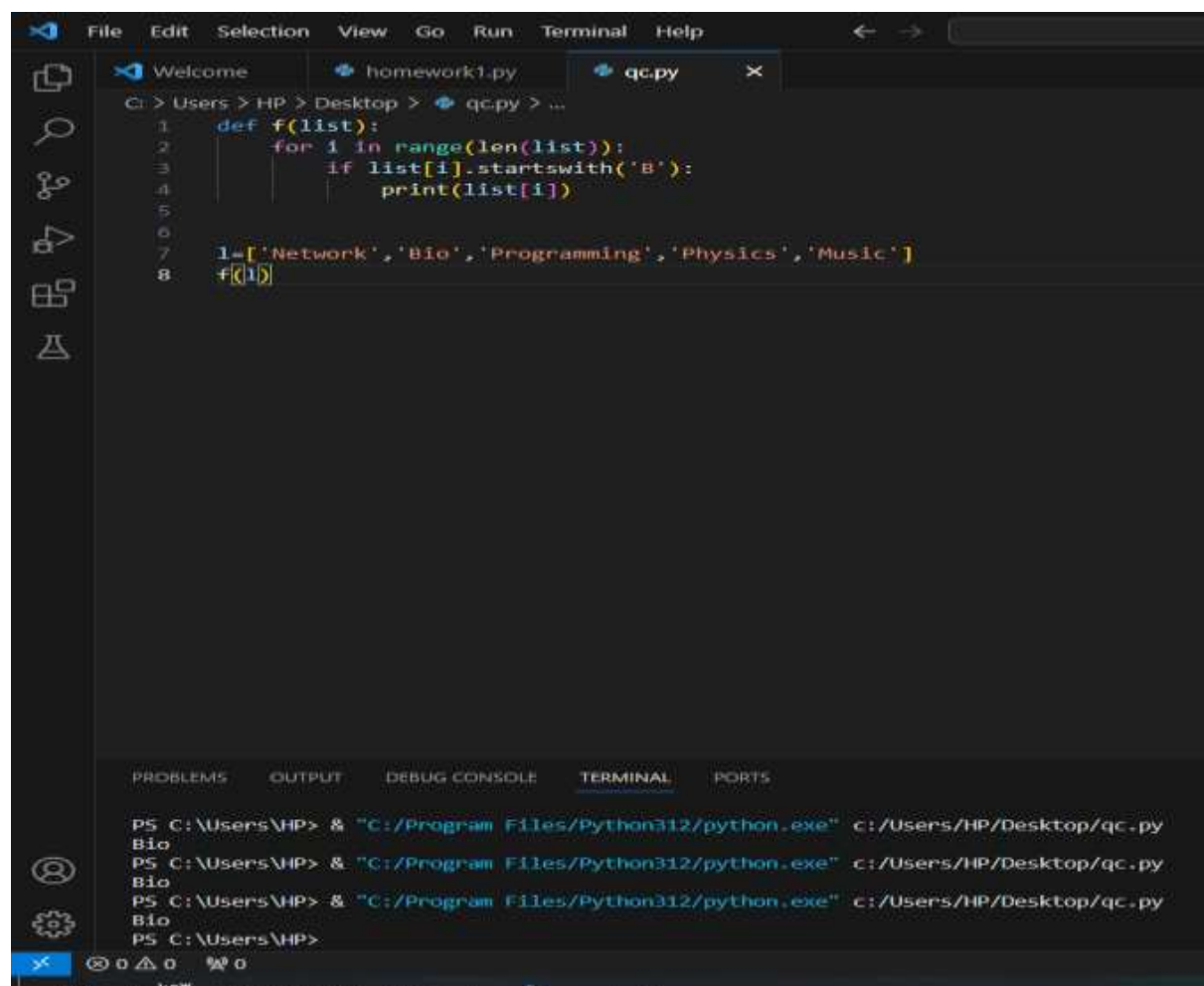
Tips: using loop, 'len ()' , startswith() methods.

تم بناء تابع ببارامتر واحد و هو القائمة يحوي حلقة for فيها متغير مجاله طول القائمة المستخدمة أي يمر على عناصرها بالترتيب اذا كان العنصر يبدأ بحرف B تتم طباعة العنصر بعد الانتهاء من بناء التابع يتم تعريف القائمة المطلوبة و استدعاء التابع الذي تم بناءه لتنفيذه و طباعة العنصر الذي يبدأ بحرف B و هو L[1]=Bio

```
def f(list):
    for i in range(len(list)):
        if list[i].startswith('B'):
            print(list[i])

l=['Network','Bio','Programming','Physics','Music']
f(l)
```

الخرج موضح في السكرين:

A screenshot of a Python IDE with a dark theme. The editor shows a file named 'qc.py' with the following code:

```
1 def f(list):
2     for i in range(len(list)):
3         if list[i].startswith('B'):
4             print(list[i])
5
6
7 l=['Network','Bio','Programming','Physics','Music']
8 f(l)
```

The bottom panel shows the 'TERMINAL' output with three identical lines of command and output:

```
P5 C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/qc.py
Bio
P5 C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/qc.py
Bio
P5 C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/qc.py
Bio
P5 C:\Users\HP>
```

D-

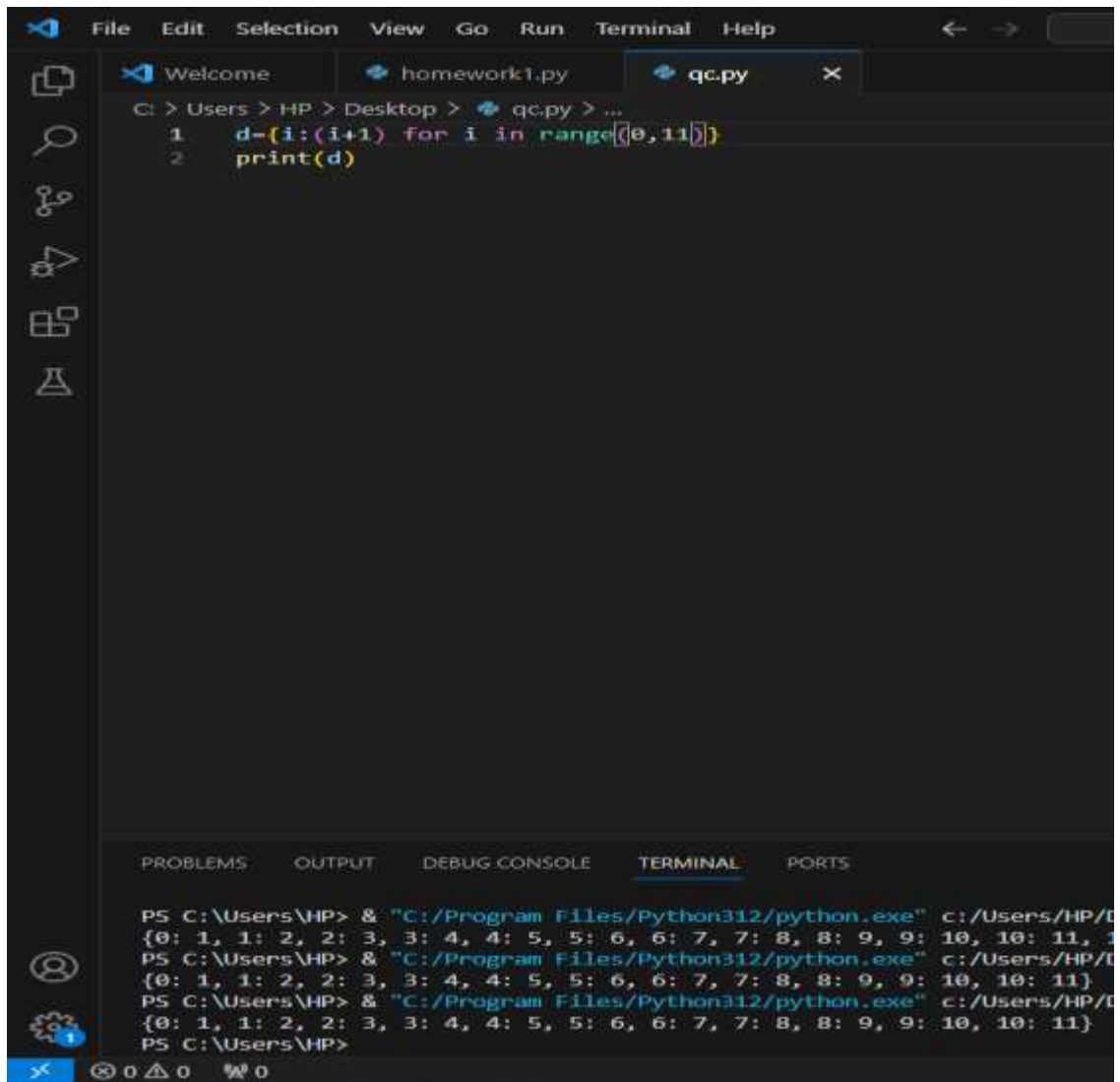
Using Dictionary comprehension, Generate this dictionary

`d={0:1,1:2,2:3,3:4,4:5,5:6,6:7,7:8,8:9,9:10,10:11}`

لبناء هذا القاموس نحتاج استخدام حلقة `for` بحيث يكون المفتاح رقم ما و تكون القيمة الرقم نفسه مضاف له واحد من اجل اختصار عملية استخدام تابع `tem` استخدام `dictionary comprehension` حيث `keys()=x` و `values()=x+1` وتم ذلك باستخدام الحلقة بمتغير يتم تنفيذه من 0 حتى 10 وفي النهاية طباعة القاموس

```
d={i:(i+1) for i in range(0,11)}  
print(d)
```

الخرج موضح في السكرين:



```
File Edit Selection View Go Run Terminal Help  
Welcome homework1.py qc.py  
C:\Users\HP\Desktop> qc.py  
1 d={i:(i+1) for i in range(0,11)}  
2 print(d)  
  
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS  
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/qc.py  
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}  
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/qc.py  
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}  
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/qc.py  
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}  
PS C:\Users\HP>
```

Question2:

Write a Python program that **converts a Binary number into its equivalent Decimal number**.

The program should start reading the binary number from the user. Then the decimal equivalent number must be calculated. Finally, the program must display the equivalent decimal number on the screen.

Tips: solve input errors.

لتحويل من عشري الى ست عشري يتم البدء في السلسلة العشرية من اليمين بترقيم من بالترتيب من 0 , 1 , 2 , 3 , ... ويكون هو مجموع جداء كل جزء من العشري (1 او 0) ب 2 اس (الترقيم المناسب) ث مثلا:

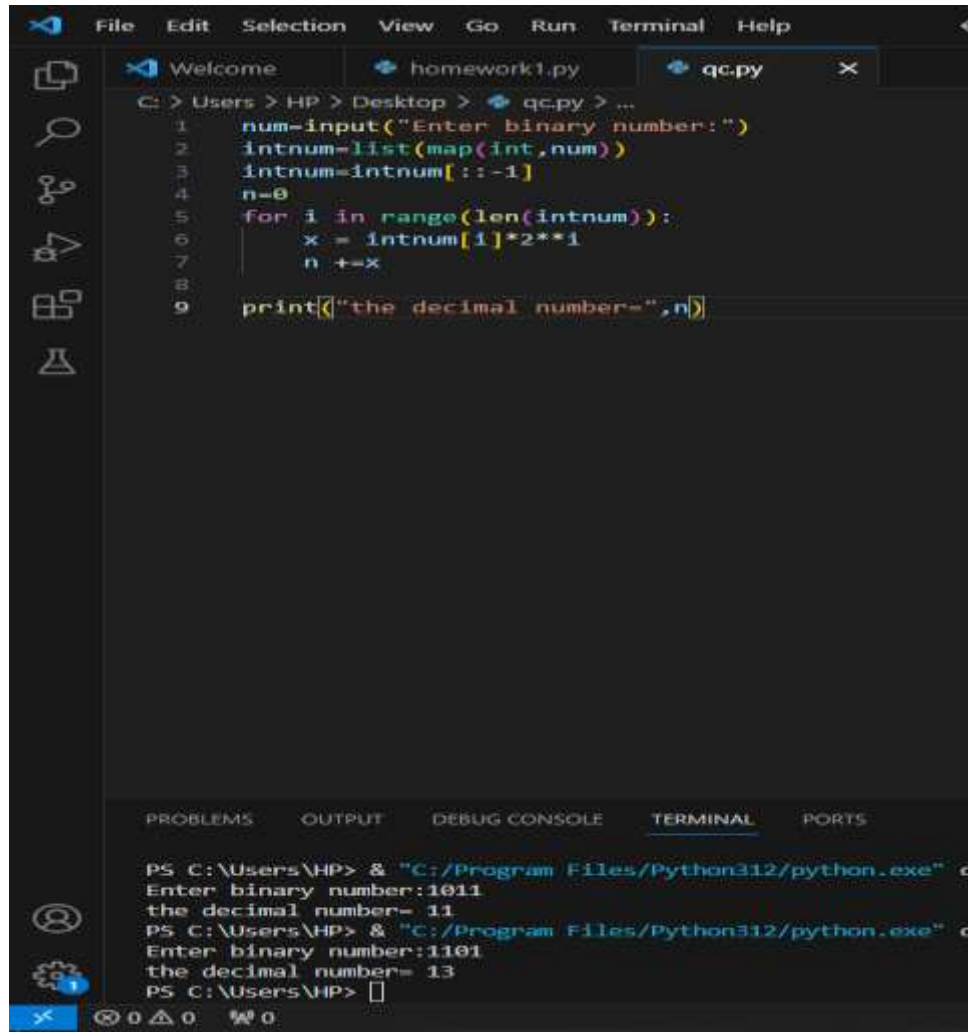
$$15 = 3^2 * 1 + 2^2 * 1 + 1^2 * 1 + 0^2 * 1 = 1111$$

و بالتالي بالكود تم أولا الطلب من المستخدم ادخال الرقم العشري ك string ثم نحوله الى ارقام بتطبيق int عليه باستخدام التابع map(int,num) و تحويله الى قائمة list لاجراء العمليات السابقة على عناصر القائمة و تم استخدام التعليمة intnum=intnum[::-1] من اجل البدء من اليمين وتم تعريف عداد n يساوي الصفر و استخدام حلقة for تمر على عناصر القائمة و يضرب العنصر ب 2 اس (index) العنصر حيث | تعبر عن index العناصر تبدا من 0 حتى نهاية طول القائمة بعد الضرب يضاف الناتج الى العداد للحصول على المجموع النهائي و طباعته

```
num=input("Enter binary number:")
intnum=list(map(int,num))
intnum=intnum[::-1]
n=0
for i in range(len(intnum)):
    x = intnum[i]*2**i
    n +=x

print("the decimal number=",n)
```

السكرين يوضح التنفيذ بعد اختيار العدد العشري 1101 و تحويله الى ست عشري:



```
File Edit Selection View Go Run Terminal Help
Welcome homework1.py qc.py
C:\Users\HP\Desktop> qc.py
1 num=input("Enter binary number:")
2 intnum=list(map(int,num))
3 intnum=intnum[::-1]
4 n=0
5 for i in range(len(intnum)):
6     x = intnum[i]*2**i
7     n +=x
8
9 print("the decimal number=",n)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c
Enter binary number:1011
the decimal number= 11
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c
Enter binary number:1101
the decimal number= 13
PS C:\Users\HP> 
```

Question3:

Type python quiz program that takes a text or json or csv file as input for (20 (Questions, Answers)). It asks the questions and finally computes and prints user results and store user name and result in separate file csv or json file.

- تم إنشاء ملف json يحوي قائمة عدد عناصرها 20 قاموس كل قاموس عبارة عن سؤال و جوابه بعد حفظ الملف quize.json تم الانتقال الى الكود
- أولا استدعاء مكتبة json
- ثانيا بناء تابع ببارامتر واحد و هو ملف الأسئلة الذي تم انشاءه و تعريف عداد s يساوي الصفر من اجل استخدامه لحساب عدد الإجابات الصحيحة لدى المستخدم

و يحوي التابع حلقة for للدخول الى عناصر قائمة quiz.json و طباعة السؤال من كل قاموس و طلب الإجابة من المستخدم و هنا يتم اختبار جواب المستخدم باستخدام الشرط if اذا كان نفسه جواب القاموس نفسه يزيد العداد واحد و هكذا.. في النهاية يعيد قيمة العداد النهائية التي تعبر عن عدد الإجابات الصحيحة للمستخدم

- تم بناء تابع اخر بثلاث بارامترات اسم المستخدم و العداد و ملف النتائج من اجل حساب و تخزين اسم المستخدم و نتيجته في ملف النتائج الذي تم بناءه باسم result.json لذلك تم وضع الاسم و النتيجة في قاموس (النتيجة هي قيمة العداد / عدد الأسئلة الكلي) و تخزين القاموس في متغير ثم فتح ملف النتائج بوضع الكتابة و تخزين المتغير السابق فيه باستخدام التعليمة

```
json.dump(result,f)
```

- بالنهاية تم انشاء تابع main ببارامترين هما ملف النتائج و ملف الأسئلة بدأ بفتح ملف json بوضع قراءة بتعليمة `with open(test_file,"r") as f:`

لذلك لا داعي لغلاق الملف عند الانتهاء و تم تحميل بيانات الملف الى متغير d

- تم البدء بعبارة الترحيب ثم طلب بإدخال الاسم ثم تم استدعاء التابع الأول و إعطاءه البارامتر d الذي خزن فيه بيانات ملف الأسئلة لاستجواب المستخدم و عد اجاباته الصحيحة و ثم استدعاء التابع الاخر و إعطاءه بارامترات الثلاث الاسم و العداد و ملف النتائج و ذلك لحساب نتيجة المستخدم و تخزينها في قاموس مع اسمه
- أخيرا تعريف مسارات الملفات التي انشئتها في كتغيرات و استدعاء التابع min و إعطائها له من اجل التنفيذ.

```
import json

def test(data):
    s=0
    for j in data:
        print(j["Question"])
        an=input("Enter your answer:")
        if an == j["answer"]:
```

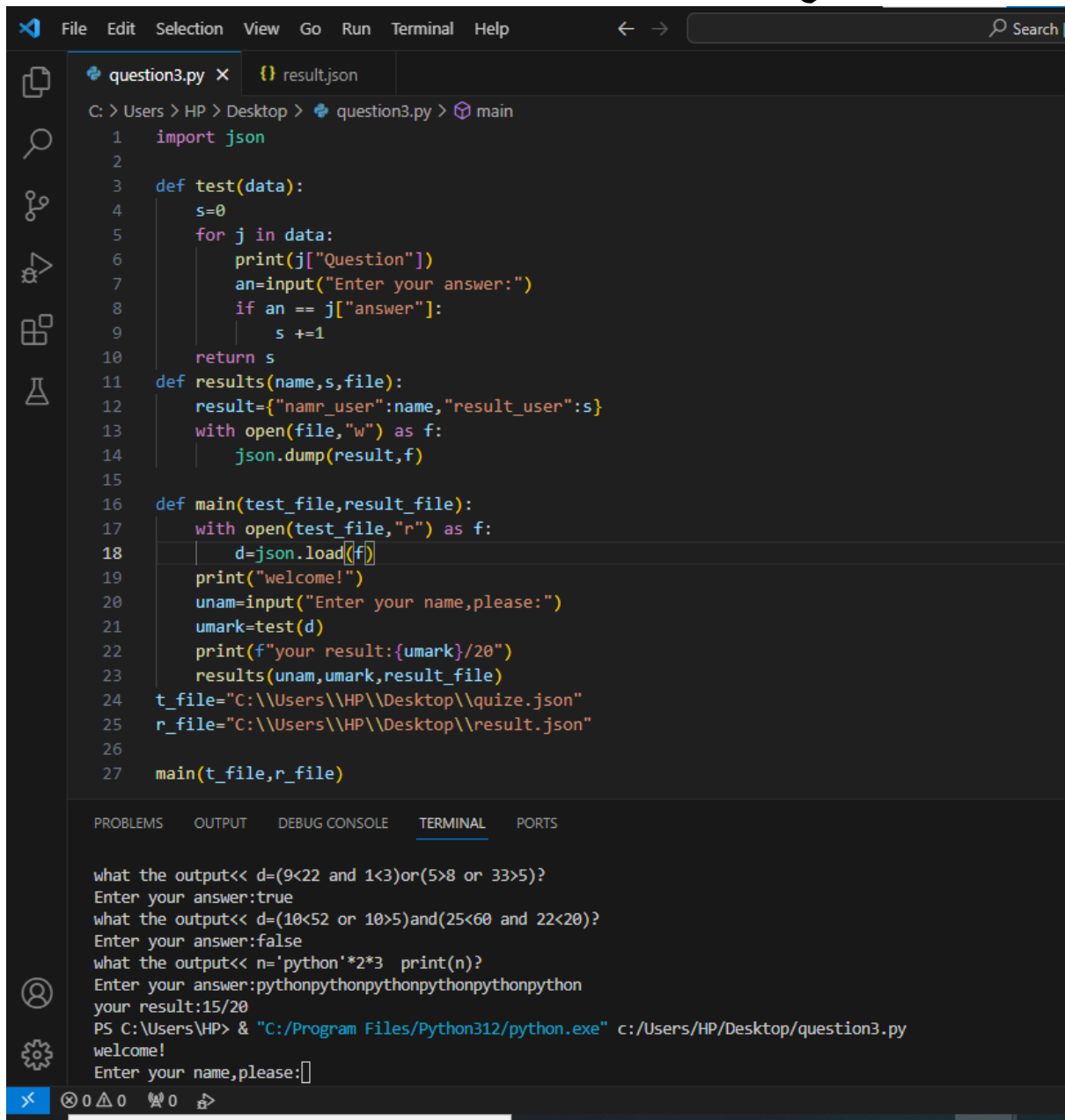
```
        s +=1
    return s
def results(name,s,file):
    result={"namr_user":name,"result_user":s}
    with open(file,"w") as f:
        json.dump(result,f)

def main(test_file,result_file):
    with open(test_file,"r") as f:
        d=json.load(f)
    print("welcome!")
    unam=input("Enter your name,please:")
    umark=test(d)
    print(f"your result:{umark}/20")
    results(unam,umark,result_file)
t_file="C:\\Users\\HP\\Desktop\\quize.json"
r_file="C:\\Users\\HP\\Desktop\\result.json"

main(t_file,r_file)
```

start run:

تم الترحيب و طلب الاسم بعد ادخال اسم zeinab تم عرض الأسئلة و تمت الإجابة عليها السكرين يوضح:



The screenshot shows a Python IDE with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, and Help. The left sidebar contains icons for Explorer, Search, Source Control, Run and Debug, Extensions, and Testing. The main editor area displays the code for 'question3.py' with line numbers 1 through 27. The code imports the 'json' module and defines three functions: 'test', 'results', and 'main'. The 'test' function iterates over a list of questions, prints each question, and checks if the user's answer matches the correct answer, incrementing a score 's' if correct. The 'results' function creates a dictionary with the user's name and score, and writes it to a JSON file. The 'main' function loads a JSON file, prints a welcome message, prompts for the user's name, calls the 'test' function, prints the result, and calls the 'results' function. The file paths for the JSON files are specified as 'C:\\Users\\HP\\Desktop\\quize.json' and 'C:\\Users\\HP\\Desktop\\result.json'. The bottom panel shows the 'TERMINAL' tab with the execution output, which includes the program's output and user input.

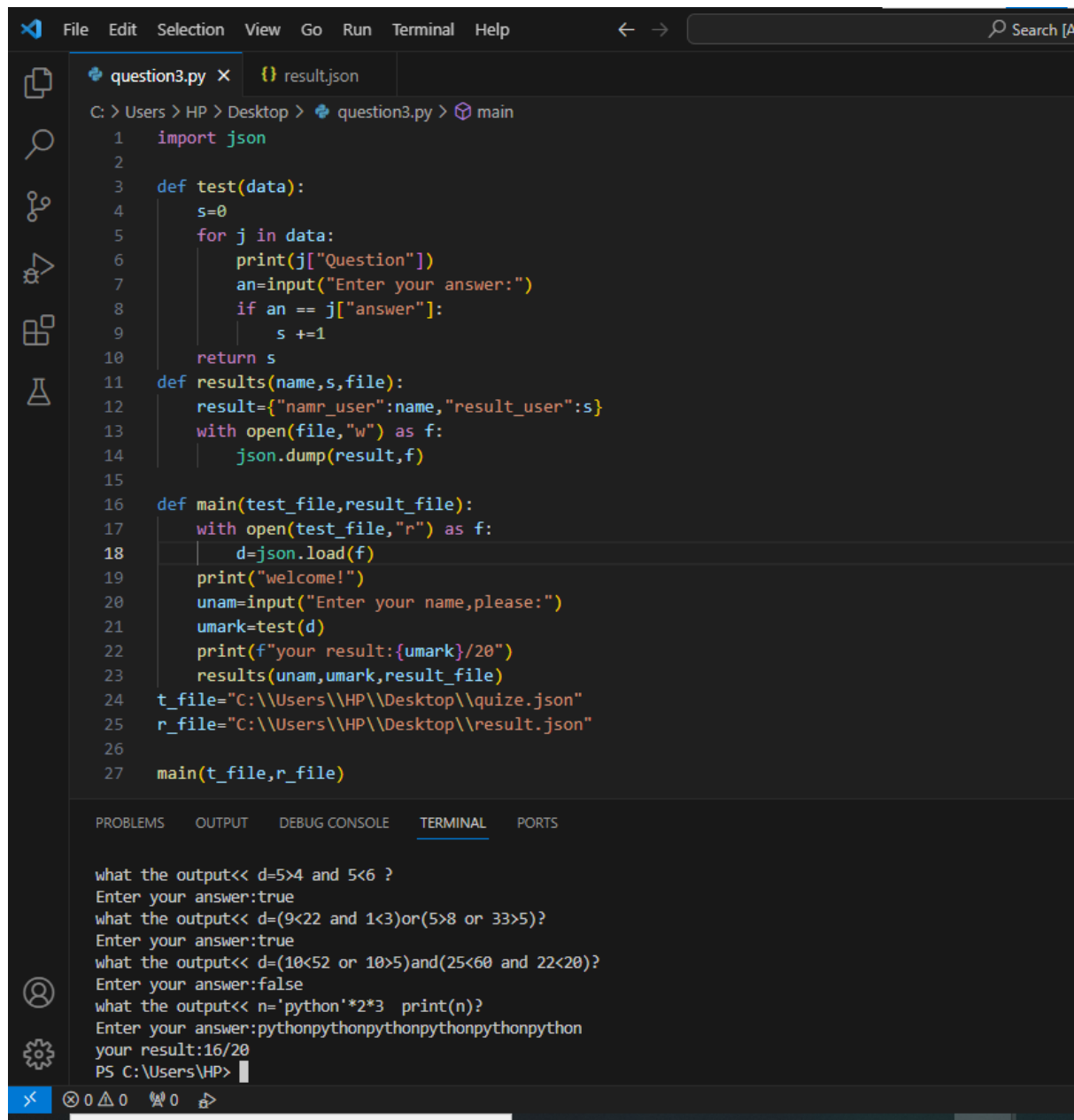
```
1 import json
2
3 def test(data):
4     s=0
5     for j in data:
6         print(j["Question"])
7         an=input("Enter your answer:")
8         if an == j["answer"]:
9             s +=1
10    return s
11
12 def results(name,s,file):
13     result={"namr_user":name,"result_user":s}
14     with open(file,"w") as f:
15         json.dump(result,f)
16
17 def main(test_file,result_file):
18     with open(test_file,"r") as f:
19         d=json.load(f)
20     print("welcome!")
21     unam=input("Enter your name,please:")
22     umark=test(d)
23     print(f"your result:{umark}/20")
24     results(unam,umark,result_file)
25
26 t_file="C:\\Users\\HP\\Desktop\\quize.json"
27 r_file="C:\\Users\\HP\\Desktop\\result.json"
28
29 main(t_file,r_file)
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
what the output<< d=(9<22 and 1<3)or(5>8 or 33>5)?
Enter your answer:true
what the output<< d=(10<52 or 10>5)and(25<60 and 22<20)?
Enter your answer:false
what the output<< n='python'*2*3 print(n)?
Enter your answer:pythonpythonpythonpythonpythonpythonpython
your result:15/20
PS C:\Users\HP> & "C:/Program Files/Python312/python.exe" c:/Users/HP/Desktop/question3.py
welcome!
Enter your name,please:
```

End run:

بعد الانتهاء من الإجابة طبع لي النتيجة و خزنها مع الاسم في ملف النتائج
result.json



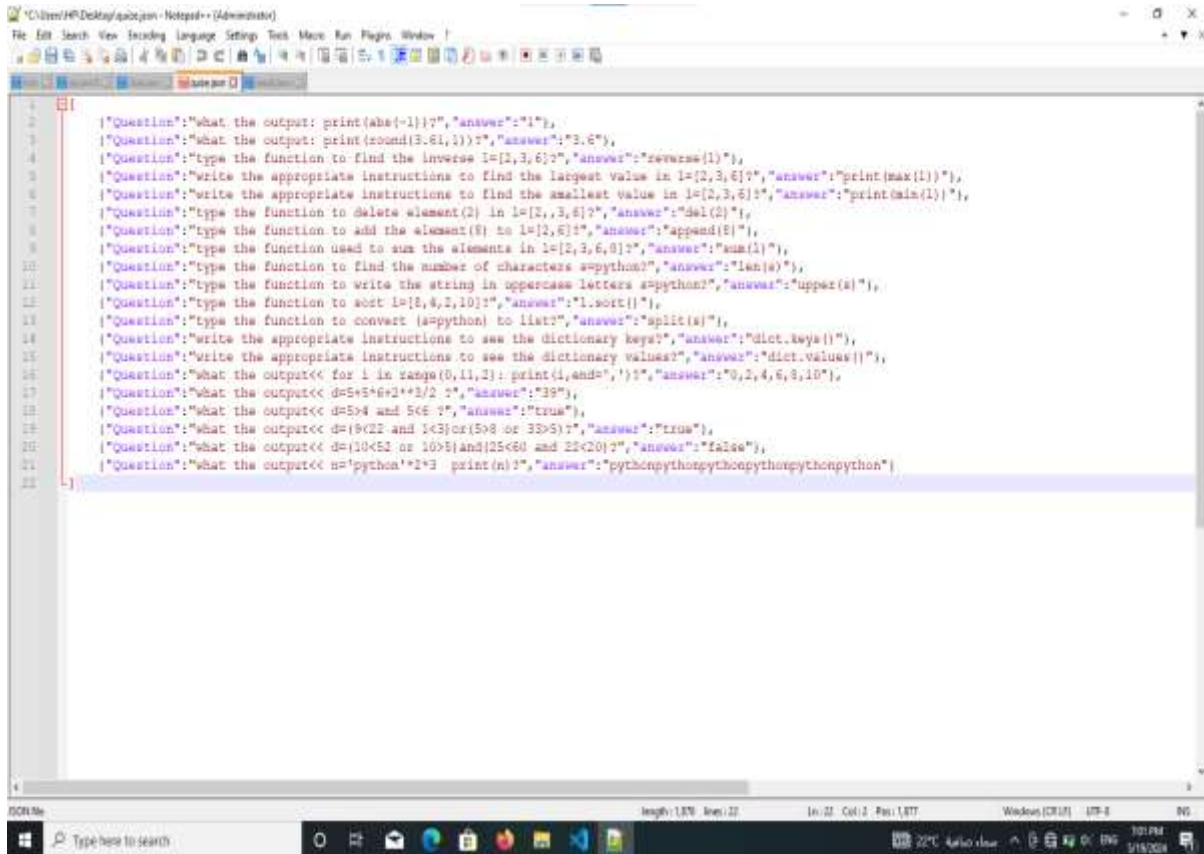
The screenshot shows a Python IDE with two tabs: `question3.py` and `result.json`. The `question3.py` tab is active, displaying the following code:

```
1 import json
2
3 def test(data):
4     s=0
5     for j in data:
6         print(j["Question"])
7         an=input("Enter your answer:")
8         if an == j["answer"]:
9             s +=1
10    return s
11
12 def results(name,s,file):
13     result={"namr_user":name,"result_user":s}
14     with open(file,"w") as f:
15         json.dump(result,f)
16
17 def main(test_file,result_file):
18     with open(test_file,"r") as f:
19         d=json.load(f)
20     print("welcome!")
21     unam=input("Enter your name,please:")
22     umark=test(d)
23     print(f"your result:{umark}/20")
24     results(unam,umark,result_file)
25
26 t_file="C:\\Users\\HP\\Desktop\\\\quize.json"
27 r_file="C:\\Users\\HP\\Desktop\\\\result.json"
28
29 main(t_file,r_file)
```

The `result.json` tab is empty. The terminal window at the bottom shows the execution of the program:

```
what the output<< d=5>4 and 5<6 ?
Enter your answer:true
what the output<< d=(9<22 and 1<3)or(5>8 or 33>5)?
Enter your answer:true
what the output<< d=(10<52 or 10>5)and(25<60 and 22<20)?
Enter your answer:false
what the output<< n='python'*2*3 print(n)?
Enter your answer:pythonpythonpythonpythonpythonpythonpython
your result:16/20
PS C:\Users\HP>
```

json file as input for (20 (Questions, Answers))



The screenshot shows a Notepad++ window with a JSON file. The file contains 20 entries, each with a "Question" and an "answer". The questions cover various Python topics including absolute values, rounding, list operations (reverse, max, min, del, append, sum), string operations (len, upper, split), dictionary operations (keys, values), and conditional expressions. The answers are provided in a concise format, often using code snippets.

```
1 [{"Question": "what the output: print(abs(-1))?", "answer": "1"},  
2 [{"Question": "what the output: print(round(3.61,1))?", "answer": "3.6"},  
3 [{"Question": "type the function to find the inverse l=[2,3,6]?", "answer": "reverse(l)"},  
4 [{"Question": "write the appropriate instructions to find the largest value in l=[2,3,6]?", "answer": "print(max(l))"},  
5 [{"Question": "write the appropriate instructions to find the smallest value in l=[2,3,6]?", "answer": "print(min(l))"},  
6 [{"Question": "type the function to delete element(2) in l=[2,,3,6]?", "answer": "del(l[2])"},  
7 [{"Question": "type the function to add the element(8) to l=[2,6]?", "answer": "append(8)"},  
8 [{"Question": "type the function used to sum the elements in l=[2,3,6,8]?", "answer": "sum(l)"},  
9 [{"Question": "type the function to find the number of characters s='python'", "answer": "len(s)"},  
10 [{"Question": "type the function to write the string in uppercase letters s='python'", "answer": "upper(s)"},  
11 [{"Question": "type the function to sort l=[5,4,2,10]?", "answer": "l.sort()"},  
12 [{"Question": "type the function to convert s='python' to list?", "answer": "split(s)"},  
13 [{"Question": "write the appropriate instructions to see the dictionary keys", "answer": "dict.keys()"},  
14 [{"Question": "write the appropriate instructions to see the dictionary values", "answer": "dict.values()"},  
15 [{"Question": "what the output<< for i in range(0,11,3): print(i,end=',')?", "answer": "0,2,4,6,8,10"},  
16 [{"Question": "what the output<< d=5*5*6+2**3/2 ?", "answer": "38"},  
17 [{"Question": "what the output<< d=5<4 and 5<6 ?", "answer": "true"},  
18 [{"Question": "what the output<< d=(9<22 and 1<3) or (5>8 or 33>5) ?", "answer": "true"},  
19 [{"Question": "what the output<< d=(10<51 or 10>5) and (25<60 and 25<20) ?", "answer": "false"},  
20 [{"Question": "what the output<< n='python'*2*3 print(n) ?", "answer": "pythonpythonpythonpythonpythonpython"}]
```

```
[  
    {"Question": "what the output: print(abs(-1))?", "answer": "1"},  
    {"Question": "what the output: print(round(3.61,1))?", "answer": "3.6"},  
    {"Question": "type the function to find the inverse  
l=[2,3,6]?", "answer": "reverse(l)"},  
    {"Question": "write the appropriate instructions to find the largest value  
in l=[2,3,6]?", "answer": "print(max(l))"},  
    {"Question": "write the appropriate instructions to find the smallest value  
in l=[2,3,6]?", "answer": "print(min(l))"},  
    {"Question": "type the function to delete element(2) in  
l=[2,,3,6]?", "answer": "del(2)"},  
    {"Question": "type the function to add the element(8) to  
l=[2,6]?", "answer": "append(8)"},  
    {"Question": "type the function used to sum the elements in  
l=[2,3,6,8]?", "answer": "sum(1)"},
```

```

    {"Question": "type the function to find the number of characters  
s=python?", "answer": "len(s)"},
    {"Question": "type the function to write the string in uppercase letters  
s=python?", "answer": "upper(s)"},
    {"Question": "type the function to l=[8,4,2,10]?", "answer": "l.sort()"},
    {"Question": "type the function to convert (s=python) to  
list?", "answer": "split(s)"},
    {"Question": "write the appropriate instructions to see the dictionary  
keys?", "answer": "dict.keys()"},
    {"Question": "write the appropriate instructions to see the dictionary  
values?", "answer": "dict.values()"},
    {"Question": "what the output<< for i in range(0,11,2):  
print(i,end=',')?", "answer": "0,2,4,6,8,10"},
    {"Question": "what the output<< d=5+5*6+2**3/2 ?", "answer": "39"},
    {"Question": "what the output<< d=5>4 and 5<6 ?", "answer": "true"},
    {"Question": "what the output<< d=(9<22 and 1<3)or(5>8 or  
33>5)?", "answer": "true"},
    {"Question": "what the output<< d=(10<52 or 10>5)and(25<60 and  
22<20)?", "answer": "false"},
    {"Question": "what the output<<  
n='python'*2*3  print(n)?", "answer": "pythonpythonpythonpythonpythonpython"}
]

```

Json file after run for sort user name and result

السكرين بعد التنفيذ:

```
File Edit Selection View Go Run Terminal Help
question3.py result.json x
C:\Users\HP\Desktop> result.json
1 {"namr_user": "zeinab", "result_user": 16}

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
what the output<< d=5>4 and 5<6 ?
Enter your answer:true
what the output<< d=(9<22 and 1<3)or(5>8 or 33>5)?
Enter your answer:true
what the output<< d=(10<52 or 10>5)and(25<60 and 22<20)?
Enter your answer:false
what the output<< n='python'*2*3 print(n)?
Enter your answer:pythonpythonpythonpythonpythonpythonpython
your result:16/20
P5 C:\Users\HP>
```

Question4:

Define a class **BankAccount** with the following attributes and methods:

Attributes: `account_number` (string), `account_holder` (string), `balance` (float, initialized to 0.0)

Methods: `deposit(amount)`, `withdraw(amount)` , `get_balance()`

- Create an instance of **BankAccount**, - Perform a deposit of \$1000, - Perform a withdrawal of \$500.

- Print the current balance after each operation.

- Define a subclass **SavingsAccount** that inherits from

BankAccount and adds `interest_rate` Attribute and

`apply_interest()` method that Applies interest to the balance

based on the interest rate.

And **Override print()** method to print the current balance and rate.

- Create an instance of SavingsAccount , and call apply_interest() and print() functions.

- تم انشاء كلاس الاب BankAccount اولا تم اختيار التابع الافتراضي `__init__` بارامتره الأول يخزن معرف ال `self` و البارامترات المطلوبة رقم الحساب و صاحبه و المبلغ يتم تخزين البارامترات ضمن متغيرات باستخدام `self`. واستخدام () قبل كل بارامتر لضمان سرية و عدم الوصول اليه
- تعريف تابع للإيداع بارامتره الأول `self` أيضا و كمية الإيداع يخزن أيضا في متغير و التابع وظيفته يعيد قيمة المبلغ الذي تم ايداعه
- تابع للسحب بارامتره الأول `self` أيضا و كمية السحب يخزن في متغير و وظيفة التابع يعيد قيمة المبلغ الذي تم سحبه
- تابع `get_balance()` لطباعة قيمة المبلغ في الحساب بعد كل عملية سحب او إيداع
- بعد الانتهاء من الكلاس الاب يتم انشاء حساب ابن `Account1` أي يرث منه جميع خصائصه فيتم وضع البارامترات ك `string` رقم الحساب و صاحبه و ك `float` المبلغ `0.0` بعده تم استدعاء تابع الإيداع و اعطاه قيمة `$ 1000` و بعد عملية الإيداع يتم استدعاء التابع `get_balance()` لمعرفة قيمة المبلغ الحالي في الحساب ثم استدعاء تابع السحب و اعطاه `$500` يتم سحبه و استدعاء التابع `get_balance()` لطباعة المبلغ الحالي في الحساب
- يتم انشاء كلاس `Saving_Account` يرث الاب أيضا تم استخدام التابع `__ini__` من الاب مع إضافة بارامتر اخر `interest_rate` معدل الفائدة تم استخدام التابع `super()` للدلالة على انه يرث هذه البارامترات من الاب و يخزن البارامتر الإضافي في متغير باستخدام `self`.
- تم بناء تابع `apply_interest()` أيضا من اجل حساب الفائدة = المبلغ * معدل الفائدة و تضاف الفائدة الى المبلغ الحالي
- و التابع `__str__` يعيد قيمة المبلغ في الحساب و معدل الفائدة و ينفذ عند استخدام امر الطباعة للحساب
- في النهاية تم انشاء حساب `s_account` يرث الكلاس أي يرث الاب أيضا و اعطاه رقم الحساب و صاحبه و المبلغ و معدل الفائدة و تم استدعاء التابع `apply_interest()` من اجل حساب الفائدة و اضافتها الى المبلغ و بتنفيذ امر طباعة الحساب يتم طباعة المبلغ الحالي مع الفائدة من التابع `__str__`


```

class BankAccount:
    def __init__(self,account_number,account_holder,balance):
        #instance variables
        self._account_number=account_number
        self._account_holder=account_holder
        self.balance=balance
    def deposit(self,amount1):
        self.balance +=amount1
        return amount1
    def withdraw(self,amount2):
        self.balance -= amount2
        return amount2
    def get_balance(self):
        print(f"You have just used your bank account and the current balance
after last operation is:",self.balance,"$")
Account1=BankAccount("125964","zeinabsa",0.0)
Account1.deposit(1000)
Account1.get_balance()
Account1.withdraw(500)
Account1.get_balance()
class Saving_Account(BankAccount):
    def __init__(self,account_number,account_holder,balance,interest_rate):
        super().__init__(account_number,account_holder,balance)
        self.interest_rate=interest_rate
    def apply_interest(self):
        interest=self.balance*self.interest_rate
        self.balance+= interest
    def __str__(self):
        return (f"The current your balance is: {self.balance} $ and your
interest rate= {self.interest_rate} %")
S_account=Saving_Account("756492","ahmad",650.0,15)
S_account.apply_interest()
print(S_account)

```

السكرين يوضح التنفيذ:

```
File Edit Selection View Go Run Terminal Help Search [Administrator]
qcpy x
C:\Users\HP\Desktop> qcpy > Saving Account > __str__
1 class BankAccount:
2     def __init__(self,account_number,account_holder,balance):
3         #instance variables
4         self.account_number=account_number
5         self.account_holder=account_holder
6         self.balance=balance
7     def deposit(self,amount1):
8         self.balance +=amount1
9         return amount1
10    def withdraw(self,amount2):
11        self.balance -= amount2
12        return amount2
13    def get_balance(self):
14        print(f"You have just used your bank account and the current balance after last operation is:",self.balance,"$")
15    Account1=BankAccount("125864","zeinabsa",0.0)
16    Account1.deposit(1000)
17    Account1.get_balance()
18    Account1.withdraw(500)
19    Account1.get_balance()
20    class Saving_Account(BankAccount):
21        def __init__(self,account_number,account_holder,balance,interest_rate):
22            super().__init__(account_number,account_holder,balance)
23            self.interest_rate=interest_rate
24        def apply_interest(self):
25            Interest=self.balance*self.interest_rate
26            self.balance+= interest
27        def __str__(self):
28            return f"The current your balance is: {self.balance} $ and your interest rate= {self.interest_rate} %"
29    $_account=Saving_Account("756492","ahmad",0.0,15)
30    $_account.apply_interest()
31    print($_account)
32
33
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\HP> & "C:/Program Files/Python311/python.exe" c:/Users/HP/Desktop/qc.py
You have just used your bank account and the current balance after last operation is: 1000.0 $
You have just used your bank account and the current balance after last operation is: 500.0 $
The current your balance is: 9750.0 $ and your interest rate= 15 %
PS C:\Users\HP>
```