question1 :A)

```
L1 = ['HTTP', 'HTTPS', 'FTP', 'DNS']      #creat first list
L2 = [80, 443, 21, 53]                    #creat second list
M = {key: value for key, value in zip(L1, L2)}            # creat dictionary
print(M)
```

```
# Define the list.py'
{'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
PS C:\Users\Hp\Desktop>
```

B)

```
def factorial(n):
  if n < 0:
    return "Factorial is not defined for negative numbers."
  elif n == 0 or n == 1:
    return 1
  else:
    return n * factorial(n - 1)
number = int(input("Enter a number: "))
result = factorial(number)
print(f"The factorial of {number} is {result}")
```

#define a function with one parameter n this function returns the factorial for un inserted number n

```
PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS                    Python Debug Console

PS C:\Users\Hp\Desktop>  & 'c:\Users\Hp\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\Hp\.vscode\extensions\ms-pytho
n.debugpy-2024.4.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '50570' '--' 'c:\Users\Hp\Desktop\Untitled-1.py'
Enter a number: 7
The factorial of 7 is 5040
PS C:\Users\Hp\Desktop>
```

C)

```
L = ['Network', 'Bio', 'Programming', 'Physics', 'Music']
# Use list comprehension to find items starting with 'B'
items_B = [item for item in L if item.startswith('B')]
# Print the matching items
for item in items_B:
  print('the item is',item)
```

```
4.4.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '50893' '--' 'C:\Users\Hp\Desktop\# Define the list.py'
the item is Bio
PS C:\Users\Hp\Desktop>
```

```
D) #creat a dictionary using for loop
h = {i: i+1 for i in range(11)}
print('the dictionary : ',h)
```

```
PS C:\Users\Hp\Desktop>  & 'c:\Users\Hp\AppData\Local\Programs\Python\Python312\python.exe' 'c:\Users\Hp\.vscode\extensions\ms-python.debugpy-202
4.4.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '51342' '--' 'C:\Users\Hp\Desktop\# Use dictionary comprehension to genera.p
y'
the dictionary :  {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\Desktop>
```

QUESTION2:

```
def binary_to_decimal(b):       # defin the function
  decimal_number = 0
  l = len(b)
  #creat a loop and calculate the decimal number
  for i in range(l):
    decimal_number += int(b[l - i - 1]) * (2 ** i)
  return decimal_number
def is_binary_number(s):
  return all(c in '01' for c in s)
def main():
#make sure the numbers intered are 0 or 1
  while True:
    b = input("Enter a binary number: ")
    if is_binary_number(b):
      decimal_number = binary_to_decimal(b)
      print(f"The decimal equivalent of binary {b} is {decimal_number}")
```

```
      break
    else:
      print("Invalid input. Please enter a valid binary number(only0sand1s).")

if __name__ == "__main__":
```

```
4.4.0-win32-x64\bundled\libs\debugpy\adapter/../..\debugpy\launcher' '56696' '--' 'C:\Users\Hp\Des
Enter a binary number: 7
Invalid input. Please enter a valid binary number (only 0s and 1s).
Enter a binary number: 01
The decimal equivalent of binary 01 is 1
PS C:\Users\Hp\Desktop>
```

```
  main()
```

QUESTION 3:

```
import json
# Initialize variables
questions = []
scores = 0
number = 1
# Load questions from file
with open("questions.txt", 'r') as f:
  questions = json.load(f)

print("Python Quiz Program")
print("Enter 't' for True or 'f' for False (if applicable)")
name = input("Enter your full name: ")
for item in questions:
  ques = item["question"]
  correct_answer = item["answer"].strip().lower()
  print("Question", number, ":", ques)
  ans = input("The answer is: ").strip().lower()
  if ans == correct_answer:
    scores += 1
    print("Correct")
  else:
```

```
    print("Wrong")
  number += 1
# Write result to file
result = {name: scores}
with open("results.txt", 'a') as m:  # Use 'a' to append to the file
  json.dump(result, m)
  m.write('\n')  # Add a new line for each result
```

```
Python Quiz Program
Enter 't' for True or 'f' for False (if applicable)
Enter your full name: hala hawla
Question 1 : What is 3+3?
The answer is: 6
Correct
Question 2 : What is 4*4?
The answer is: 16
Correct
Question 3 : Who wrote 'Romeo and Juliet'?
The answer is: me
Wrong
Question 4 : What is the chemical symbol for water?
The answer is:
```

QUESTION4:

```
#creat a class
class bankaccount:
# set attributs
  def __init__ (self,account_number,account_holder,balance=0.0):
    self.account_number=account_number
    self.account_holder=account_holder
    self.balance=balance
  def deposit(self,amount):      # creat a method to calculate the added amount
    self.balance +=amount
    print(amount)
  def withdraw(self,amount):    # creat a method to calculate the drawn amount
    if self.balance>=amount:
      self.balance -=amount
```

```
    print(amount)
# a method to print the balance after adding or drawing
  def get_balance(self):
    print(self.balance)
#creat an instance from the class bankaccount
p = bankaccount("123456","hala")
print("account number:" , p.account_number , "user name:" , p.account_holder)
print("deposit:")
p.deposit(1000)
print("draw:")
p.withdraw(500)
print("balance:")
p.get_balance()
#a subclass from the class bankaccount with one extra paramere
class savingaccount(bankaccount):
  def __init__(self, account_number, account_holder, balance=0,interset_rate=0):
    super().__init__(account_number, account_holder, balance)
    self.interest_rate=interset_rate
  def apply_interest(self):
    self.balance +=self.balance * (self.interest_rate / 100)
  def overrideprint(self):
    print("new balance:" ,self.balance ,"new interest rate:" ,
self.interest_rate)
h=savingaccount("4455","hala",5,6)
h.apply_interest()
h.overrideprint()
```

```
account number: 123456 user name: hala
deposit:
1000
draw:
500
balance:
500.0
new balance: 5.3 new interest rate: 6
PS C:\Users\Hp>
```