

الاسم: إيلين عبدالله غصة

الرقم الجامعي: 2258

p1 وظيفة برمجة

Question 1:

A-If you have two lists, L1=['HTTP','HTTPS','FTP','DNS'] L2=[80,443,20,53], convert it to generate this dictionary d={'HTTP':80,'HTTPS':443,'FTP':20,'DNS':53 }

```
d= {}  
L1 = ['HTTP', 'HTTPS', 'FTP', 'DNS']  
L2 = [80, 443, 20, 53]  
for i,j in zip(L1,L2):  
    d[i]=j  
print(d)
```

```
C:\python312\python.exe "C:\Users\lenovo\Desktop\lec6 codes\1.py"  
{'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}
```

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

```
1.py 2.py ×  
1 x=int(input())  
2 f=1  
3 for i in range(1,x+1):  
4     f=f*i  
5 print(f)  
6
```

```
C:\python312\python.exe "C:\Users\lenovo\Desktop\lec6 codes\2.py"  
5  
120
```

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 ()', starts with() method

```
1.py 2.py 3.py x
1 L= ['Network', 'Bio', 'Programming', 'Physics', 'Music']
2 i = 0
3 for i in range(len(L)):
4     if L[i].startswith("B"):
5         print(L[i])
6
7
8
```

```
C:\python312\python.exe "C:\Users\lenovo\Desktop\lec6 codes\3.py"
Bio
```

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}

```
1.py 4.py x 2.py 3.py
1 d= {a:a+1 for a in range(0,11)}
2 print(d)
3
4
5
```

```
C:\python312\python.exe "C:\Users\lenovo\Desktop\lec6 codes\4.py"
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}
```

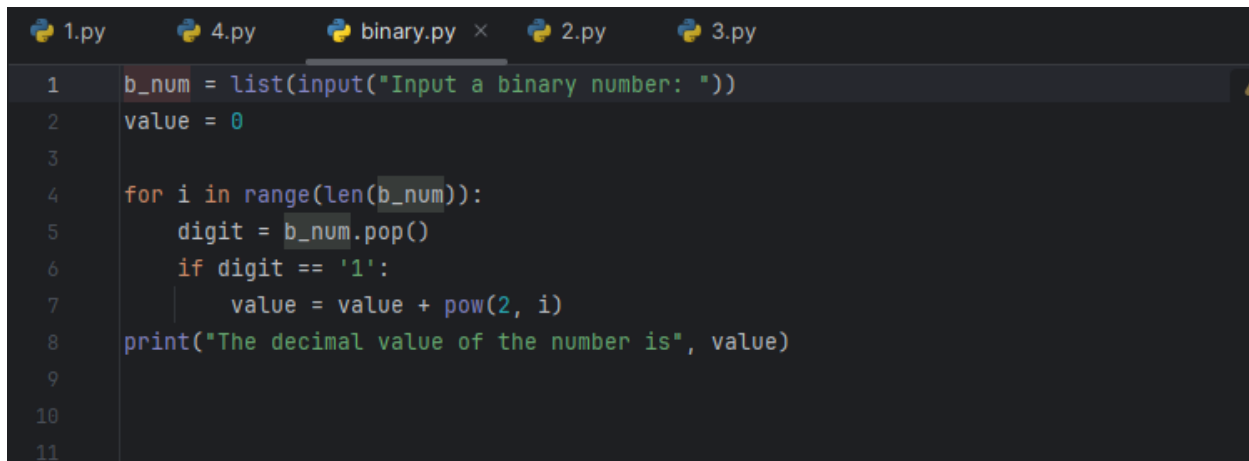
Question 2:

Convert from Binary to Decimal.

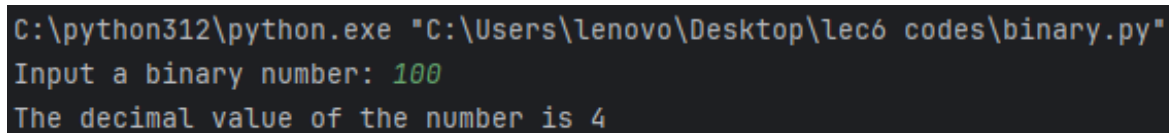
Write a Python program that converts a Binary number into its equivalent Decimal number.

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.

A screenshot of a Python IDE with a dark theme. The top bar shows several open files: 1.py, 4.py, binary.py (selected), 2.py, and 3.py. The main editor area displays the following Python code:

```
1 b_num = list(input("Input a binary number: "))
2 value = 0
3
4 for i in range(len(b_num)):
5     digit = b_num.pop()
6     if digit == '1':
7         value = value + pow(2, i)
8 print("The decimal value of the number is", value)
9
10
11
```

A screenshot of a Windows command prompt window. The first line shows the command to run the program: `C:\python312\python.exe "C:\Users\lenovo\Desktop\lec6 codes\binary.py"`. The second line shows the user input: `Input a binary number: 100`. The third line shows the program output: `The decimal value of the number is 4`.

Question 3:

Working with Files” Quiz Program”.

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.

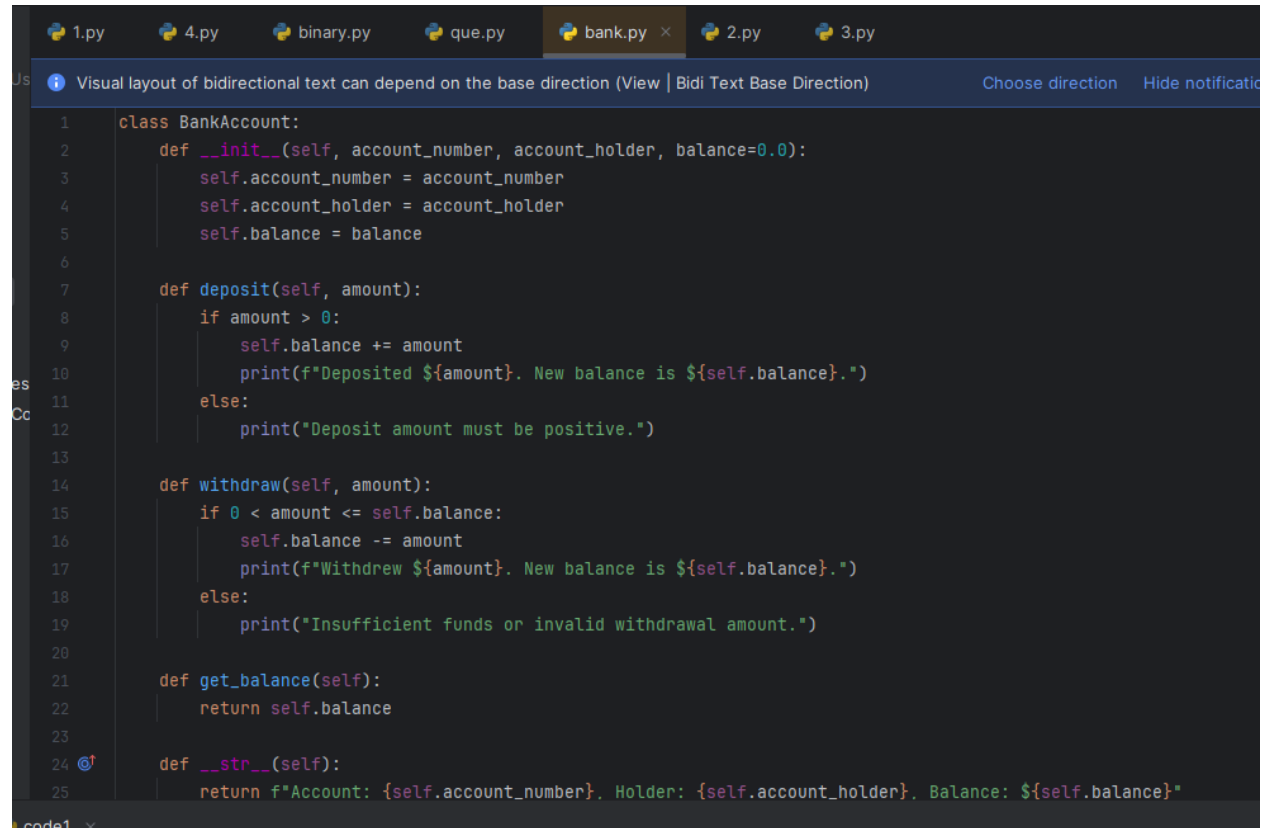
```
1.py 4.py binary.py que.py x 2.py 3.py
1 import json
2 que = {}
3 res = 0
4 num = 1
5 e= open("que.txt",'r')
6 que = json.load(e)
7 e.close()
8 print("Enter t or f ")
9 name = input("Enter your name: ")
10 for i in que.keys():
11     print("Question",num," : ", i)
12     answer = input("The answer ")
13     if answer.lower() == que[i].lower():
14         res = res + 1
15
16     num = num + 1
17
18 result={name:res}
19 m = open("result.txt",'w')
20 final = json.dump(result,m)
21 m.close()
```

result - المفكرة

ملف تحرير تنسيق عرض تعليمات
{elen ": 11"}
t

Question 4:

Object-Oriented Programming - Bank Class 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



```
1 class BankAccount:
2     def __init__(self, account_number, account_holder, balance=0.0):
3         self.account_number = account_number
4         self.account_holder = account_holder
5         self.balance = balance
6
7     def deposit(self, amount):
8         if amount > 0:
9             self.balance += amount
10            print(f"Deposited ${amount}. New balance is ${self.balance}.")
11        else:
12            print("Deposit amount must be positive.")
13
14    def withdraw(self, amount):
15        if 0 < amount <= self.balance:
16            self.balance -= amount
17            print(f"Withdrew ${amount}. New balance is ${self.balance}.")
18        else:
19            print("Insufficient funds or invalid withdrawal amount.")
20
21    def get_balance(self):
22        return self.balance
23
24    def __str__(self):
25        return f"Account: {self.account_number}, Holder: {self.account_holder}, Balance: ${self.balance}"
```

```
1.py 4.py binary.py que.py bank.py x 2.py 3.py
Visual layout of bidirectional text can depend on the base direction (View | Bidi Text Base Direction) Choose direction Hide notification Don't sh
28 class SavingsAccount(BankAccount):
32
33     def apply_interest(self):
34         interest = self.balance * self.interest_rate
35         self.balance += interest
36         print(f"Interest applied at rate {self.interest_rate}. New balance is ${self.balance}.")
37
38     def __str__(self):
39         return f"{super().__str__()}, Interest Rate: {self.interest_rate}"
40
41
42 account = BankAccount( account_number="123456789", account_holder="elen ghasa")
43 account.deposit(1000)
44 account.withdraw(500)
45 print(account)
46
47
48 savings_account = SavingsAccount( account_number="456789321", account_holder="jamil", interest_rate=0.02)
49 savings_account.deposit(1000)
50 savings_account.apply_interest()
51 print(savings_account)
```