



الجمهورية العربية السورية

وزارة التعليم العالي

جامعة تشرين

قسم هندسة الاتصالات والإلكترونيات

السنة الخامسة: وظيفة 1 برمجة شبكات

إعداد الطالب:

عدي احمد عفيف

الرقم الجامعي:

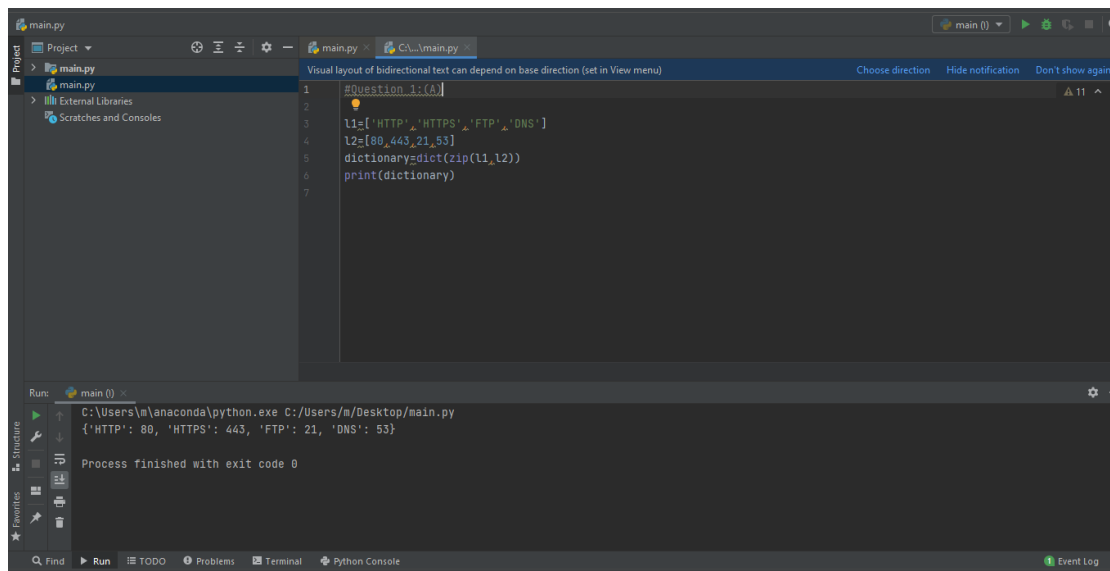
2997

السؤال الاول:A

*في هذا الكود قمنا بتعريف المتغير L1 يحوي قائمة تحتوي العناصر ['Http','Https','Ftp','DNS']

*ثم عرفنا المتغير L2 يحوي قائمة تحتوي العناصر [80,443,21,53]

*قمنا باستخدام التابع ZIP الذي يقوم بتجمع القائمتين L1 وL2 حيث يقوم بوضع كل عنصر من القائمة L1 والعنصر الموافق له من القائمة L2 ثم عرفنا دالة dict لتحويل الأزواج المجمعة الى قاموس ووضعناها في متغير dictionary ثم قمنا بطباعة هذا المتغير



```
1 #Question 1:(A)
2
3 l1=['HTTP','HTTPS','FTP','DNS']
4 l2=[80,443,21,53]
5 dictionary=dict(zip(l1,l2))
6 print(dictionary)
7
```

Run: main ()

C:\Users\m\anaconda\python.exe C:/Users/m/Desktop/main.py
{'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}

Process finished with exit code 0

السؤال الاول: B

*عرفنا المتحول X من النوع int يقوم فيه المستخدم بادخال العدد

*عرفنا الدالة def fact(x) والتي تاخذ معاملا واحدا هو X

*عرفنا المتغير f واعطيناه قيمة افتراضية 1

*وضعنا الشرط If الذي يتأكد فيما اذا كان العدد المدخل سالبا والذي يرجع رسالة خطأ "error number"

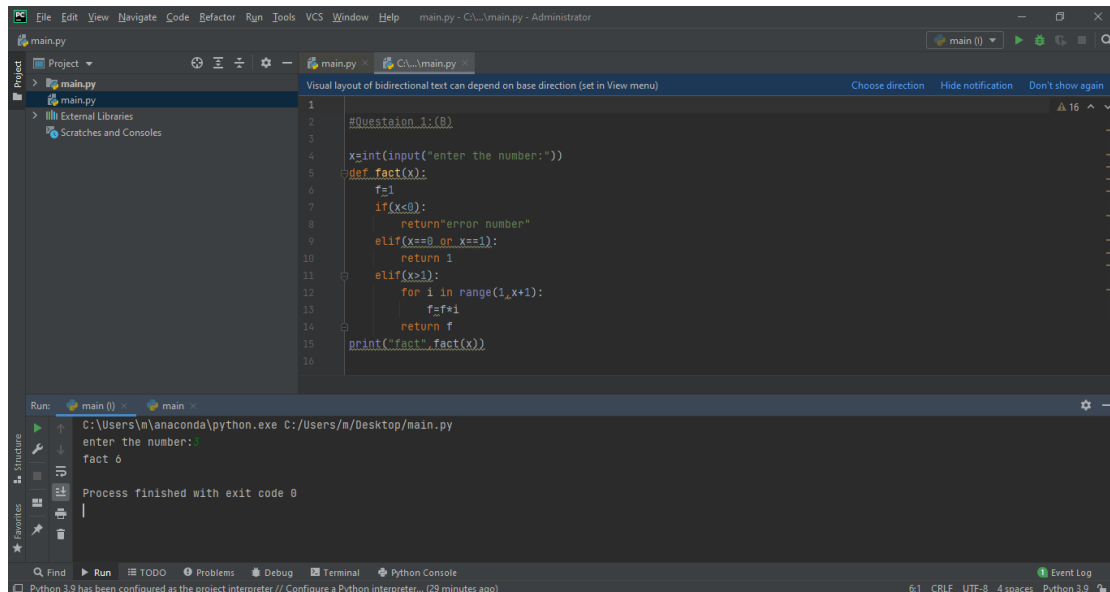
*سيتم التأكد اذا كان العدد المدخل 0 أو 1 سيتم ارجاع القيمة 1

*يتم التأكد ان العدد المدخل اكبر من 1 ثم تبدأ حلقة فور تاخذ قيم تكرارية من 1 حتى x الرقم المدخل

ثم في كل تكرار للحلقة for يتم ضرب قيمة المتغير f=i

*بعد انتهاء الحلقة التكرارية يتم ارجاع قيمة المتغير f

*ثم يقوم بطباعة نتيجة الدالة fact(x) باعطاء قيم للمتغير



```
1 #Question 1:(B)
2
3 x=int(input("enter the number:"))
4
5 def fact(x):
6     f=1
7     if(x<0):
8         return "error number"
9     elif(x==0 or x==1):
10        return 1
11    elif(x>1):
12        for i in range(1,x+1):
13            f=f*i
14        return f
15    print("fact",fact(x))
16
```

Run: main () × main ×

C:\Users\m\anaconda\python.exe C:/Users/m/Desktop/main.py

enter the number: 6

fact 6

Process finished with exit code 0

Python 3.9 has been configured as the project interpreter // Configure a Python interpreter... (28 minutes ago)

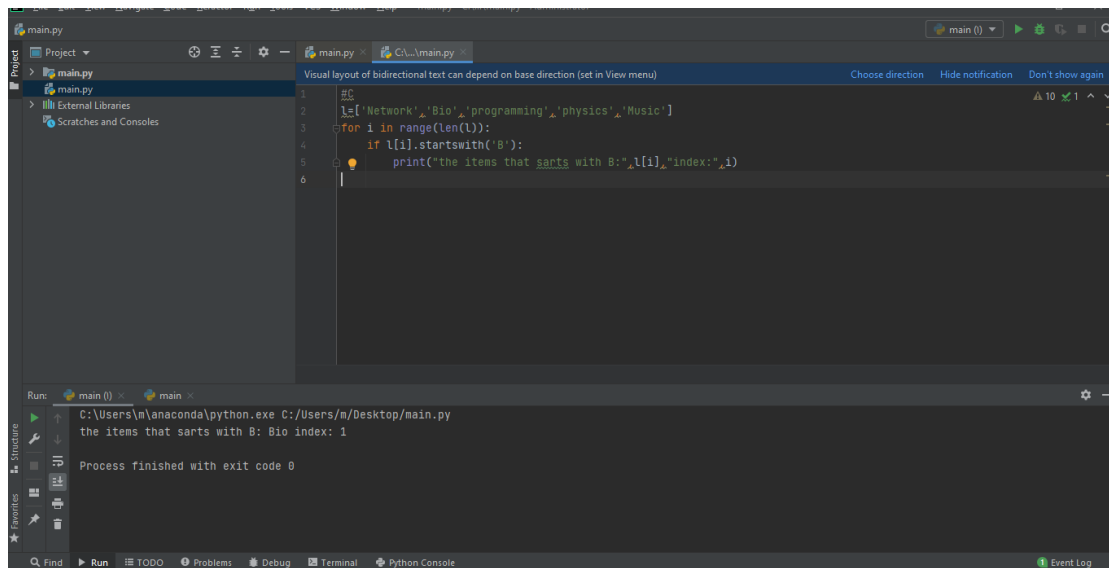
السؤال الاول:C

نقوم اولاً بتعريف المتغير **L** يحوي قائمة تحتوي العناصر التالية
['Network','Bio','programming','physics','Music']

*نعرف حلقة **for** يقوم فيها المتغير **i** بتكرار العملية لكل عنصر في القائمة **len(l)-1** الى **0**

*يتم التحقق فيما اذا كان العنصر الحالي يبدأ بالحرف **"B"**

اذا كان الشرط صحيحاً سيتم طباعة العنصر وموقعه في القائمة



```
1 #C
2 l=['Network','Bio','programming','physics','Music']
3 for i in range(len(l)):
4     if l[i].startswith('B'):
5         print("the items that sarts with B:"l[i],"index:",i)
6
```

Run: main (l) × main ×

C:\Users\m\anaconda\python.exe C:\Users\m\Desktop/main.py
the items that sarts with B: Bio index: 1

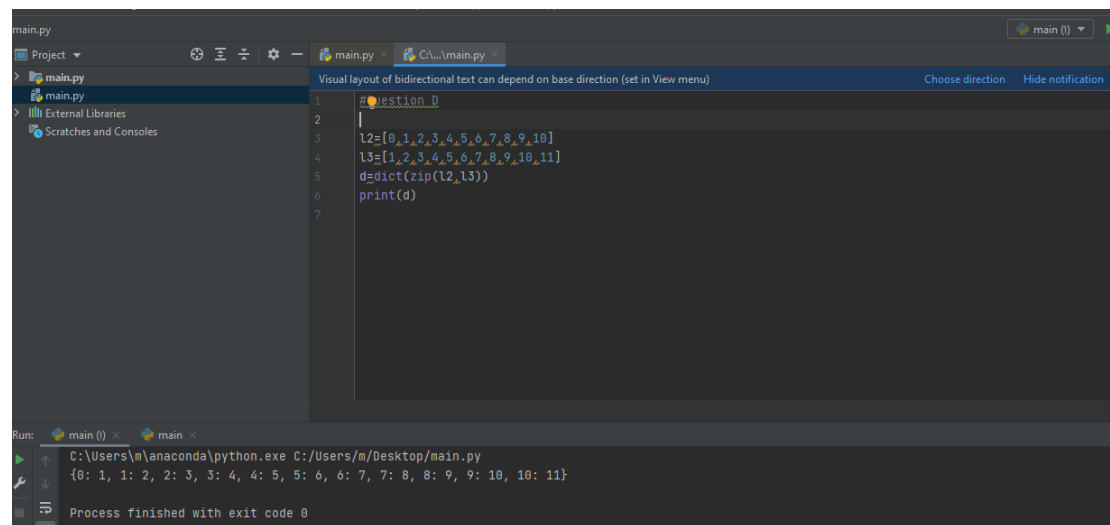
Process finished with exit code 0

السؤال الاول:D

*في هذا الكود قمنا بتعريف المتغير L1 يحوي قائمة تحتوي العناصر
[0,1,2,3,4,5,6,7,8,9,10]

*ثم عرفنا المتغير L2 يحوي قائمة تحتوي العناصر [1,2,3,4,5,6,7,8,9,10,11]

*قمنا باستخدام التابع ZIP الذي يقوم بتجمع القائمتين L1 وL2 حيث يقوم بوضع كل
عنصر من القائمة L1 والعنصر الموافق له من القائمة L2 ثم عرفنا دالة dict لتحويل
الازواج المجمع الى قاموس ووضعناها في متغير d ثم قمنا بطباعة هذا المتغير



```
main.py
Project
main.py
main.py
External Libraries
Scratches and Consoles

Visual layout of bidirectional text can depend on base direction (set in View menu)
Choose direction Hide notification

1 #Question_D
2 |
3 l2=[0,1,2,3,4,5,6,7,8,9,10]
4 l3=[1,2,3,4,5,6,7,8,9,10,11]
5 d=dict(zip(l2,l3))
6 print(d)
7

Run:
main () x main x
C:\Users\m\anaconda\python.exe C:/Users/m/Desktop/main.py
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}
Process finished with exit code 0
```

السؤال الثاني:2

*نقوم بتحويل العدد الثنائي الذي يدخله المستخدم الى قائمة

*نعرف متغير sum يعبر عن المجموع للقيام بعملية التحويل من ثنائي الى عشري

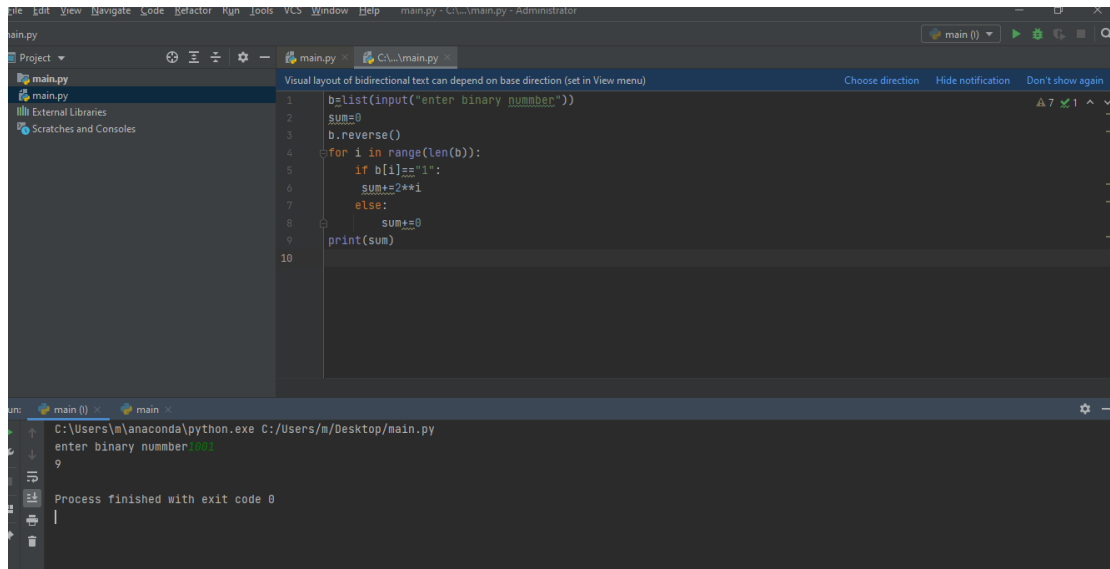
*نقوم بعكس القائمة المعبرة عن الرقم العشري المدخل

*ثم حلقة for تتكرر بحسب طول القائمة

*نختبر اذا كانت قيمة العنصر ضمن القائمة تساوي 1 نضيف على sum حسب قانون التحويل

2 للقوة i والتي توافق قيمة فهرس القائمة والا يضيف 0

*بالنهاية نطبع قيمة sum وهي المعبرة عن قيمة تحويل الثنائي الى عشري



```
1 b=list(input("enter binary number"))
2 sum=0
3 b.reverse()
4 for i in range(len(b)):
5     if b[i]=="1":
6         sum+=2**i
7     else:
8         sum+=0
9 print(sum)
```

main ()

C:\Users\m\anaconda\python.exe C:/Users/m/Desktop/main.py

enter binary number 101

5

Process finished with exit code 0

السؤال الثالث:

*نستدعي json

*نعرف مجموعة من المتغيرات

متغير الاسئلة هو من النوع قاموس ليحتوي على الاسئلة والاجوبة

يحتوي متغير الدرجات على الدرجة الاجمالية للاسئلة التي تمت لاجابة عليها

يشير المتغير num الى رقم السؤال

ثم نفتح الملف لقراءته وتنزيله واغلاقه

*ادخل الاسم وهو اسم الطالب الذي سيجيب على الاسئلة

*ادخل حلقة for التي تتكرر وفقا لمفاتيح القاموس

*داخل الحلقة يتم طباعة السؤال مع رقمه ويتم عرض السؤال ومن ثم تخزين اجابة الطالب في المتغير ans

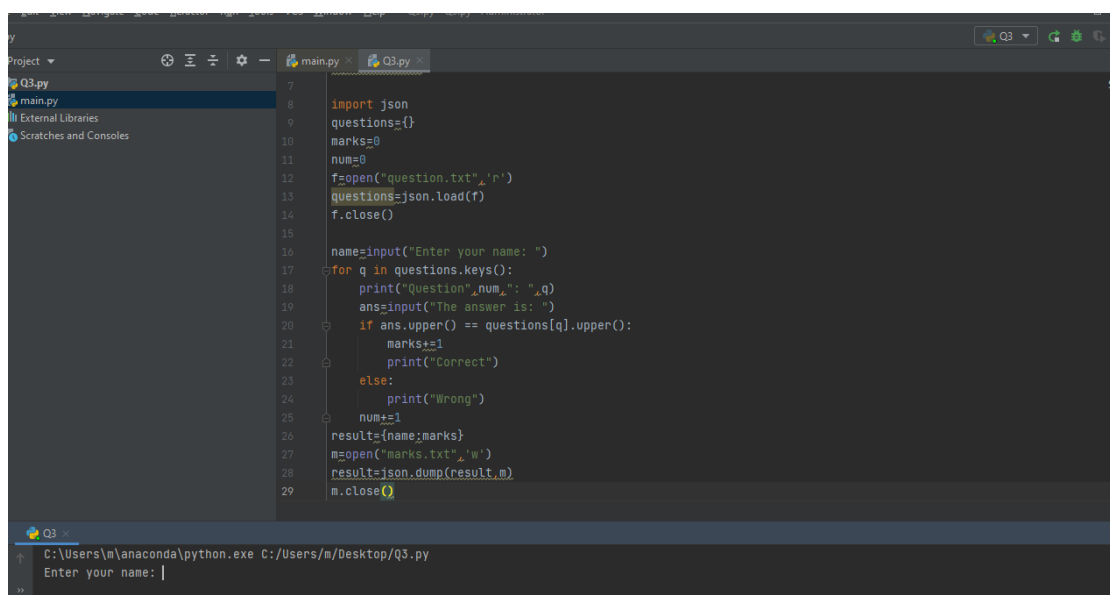
ثم مقارنة قيمة اجابة الطالب مع قيمة مفتاح القاموس الذي يمثل الاجابة الصحيحة على السؤال بعد تحويل حالة كل منهما الى حروف كبيرة لمراعاة حالة الاحرف وتوحيدها

وفي حالة التساوي تكون اجابة الطالب صحيحة فيتم اضافة العلامة لمجموع العلامات وتطبع رسالة ان الاجابة صحيحة والا تكون الاجابة خاطئة وتظهر رسالة بذلك

*بعد الانتهاء من حلقة for يتم تخزين ضمن متغير result النتيجة من النوع dictionary

المفتاح هو اسم الطالب وقيمه مجموع علامات الطالب النهائية بعد الاجابة على جميع الاسئلة

*ثم نفتح ملف ويكون للكتابة ونخزن ضمنه النتيجة ثم نقوم باغلاقه



```
7
8 import json
9 questions={}
10 marks=0
11 num=0
12 f=open("question.txt","r")
13 questions=json.load(f)
14 f.close()
15
16 name=input("Enter your name: ")
17 for q in questions.keys():
18     print("Question",num," : ",q)
19     ans=input("The answer is: ")
20     if ans.upper() == questions[q].upper():
21         marks+=1
22         print("Correct")
23     else:
24         print("Wrong")
25     num+=1
26 result={name:marks}
27 m=open("marks.txt","w")
28 result=json.dump(result,m)
29 m.close()
```

Q3 x
C:\Users\m\anaconda\python.exe C:\Users\m\Desktop\Q3.py
Enter your name: |

```
}
,"The capital of Syria is Damascus": "t"
,"The capital of Lebanon is Beirut": "t"
,"The capital of Egypt is Cairo": "t"
,"The capital of Yemen is Ankara": "f"
,"The capital of Kuwait is Kuwait": "t"
,"The capital of Qatar is Doha": "t"
,"The capital of Algeria is Algeria": "t"
,"The capital of Iraq is Tunisia": "f"
,"The capital of Russia is Moscow": "t"
,"The capital of Turkey is Istanbul": "f"
,"The capital of Germany is Berlin": "t"
,"The capital of Japan is Beijing": "f"
,"The capital of China is Tokyo": "f"
,"The capital of France is Paris": "t"
,"The capital of United States of America is Washington": "t"
,"The capital of Bahrain is Manama": "t"
,"The capital of Iran is Roma": "f"
,"The capital of Tunisia is Tripoli": "f"
,"The capital of Sudan is Cairo": "f"
,"The capital of Morocco is Riyadh": "f"
{
```


السؤال الرابع:

*لدينا class Bank يعبر عن عمل عمليات الایداع والسحب والتحويل(النقل) في البنك
*بالتالي يكون لدينا في التابع الباني المتغيرين account يدل على الحساب وbalance يدل على الرصيد

*التابع deposit يدل على ايداع مبلغ في الحساب

*التابع withdraw يدل على سحب مبلغ من الحساب بحيث اذا كانت قيمة المبلغ المراد سحبه اكبر من الرصيد لا يتم السحب وتظهر رسالة بذلك والا يتم السحب

*التابع transfer يدل على تحويل مبلغ مالي معين من حساب لحساب اخر ويناقش حالة مقارنة قيمة المبلغ المراد تحويله مع الرصيد المالي في الحساب

*account 1

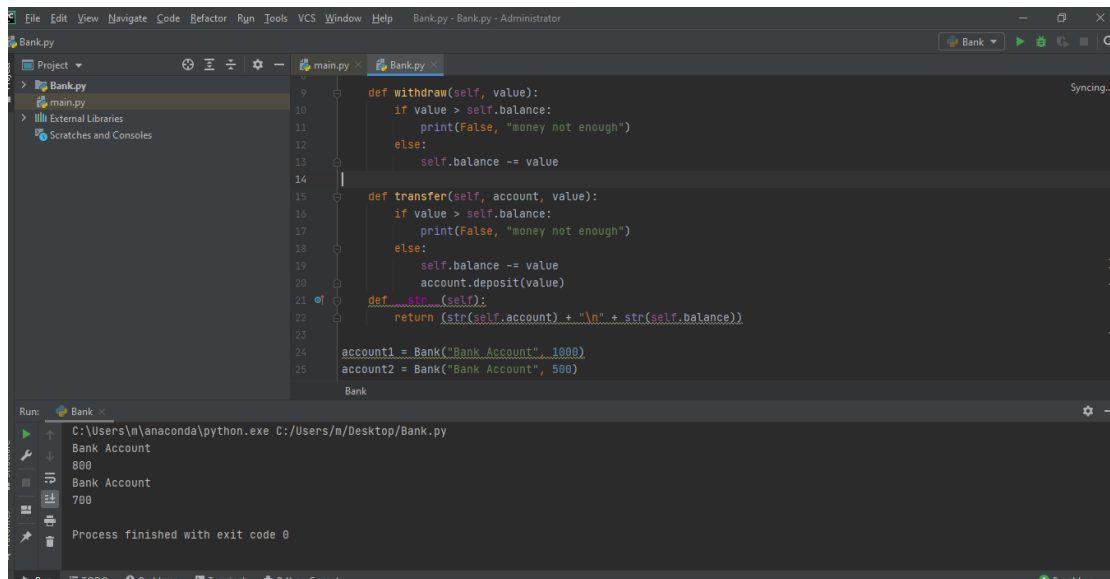
*يدل على الحساب الاول ويحوي رصيد 10000

*account 2

*يدل على الحساب الثاني ويحوي رصيد 500

*بعد ذلك قمنا بعملية تحويل مبلغ 200 من الحساب الاول للحساب الثاني

*في النهاية عرضنا رصيد كل من الحسابين فيكون أصبح الرصيد في الحساب الاول 800 وفي الحساب الثاني 700



```
File Edit View Navigate Code Refactor Run Tools VCS Window Help Bank.py - Bank.py - Administrator
Bank.py
Project
  Bank.py
  main.py
  External Libraries
  Scratches and Consoles
main.py
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
def withdraw(self, value):
    if value > self.balance:
        print(False, "money not enough")
    else:
        self.balance -= value

def transfer(self, account, value):
    if value > self.balance:
        print(False, "money not enough")
    else:
        self.balance -= value
        account.deposit(value)

def __str__(self):
    return (str(self.account) + "\n" + str(self.balance))

account1 = Bank("Bank Account", 1000)
account2 = Bank("Bank Account", 500)

Bank

Run: Bank
C:\Users\n\anaconda\python.exe C:/Users/n/Desktop/Bank.py
Bank Account
800
Bank Account
700
Process finished with exit code 0
```