



الجمهورية العربية السورية

جامعة تشرين

كلية الهندسة الميكانيكية والكهربائية

قسم هندسة الاتصالات والإلكترونيات

السنة الخامسة

وظيفة 1 برمجة الشبكات

First Network Programming Homework

إعداد الطالبة:

الرقم الجامعي: 2453

شازان جلال حسن

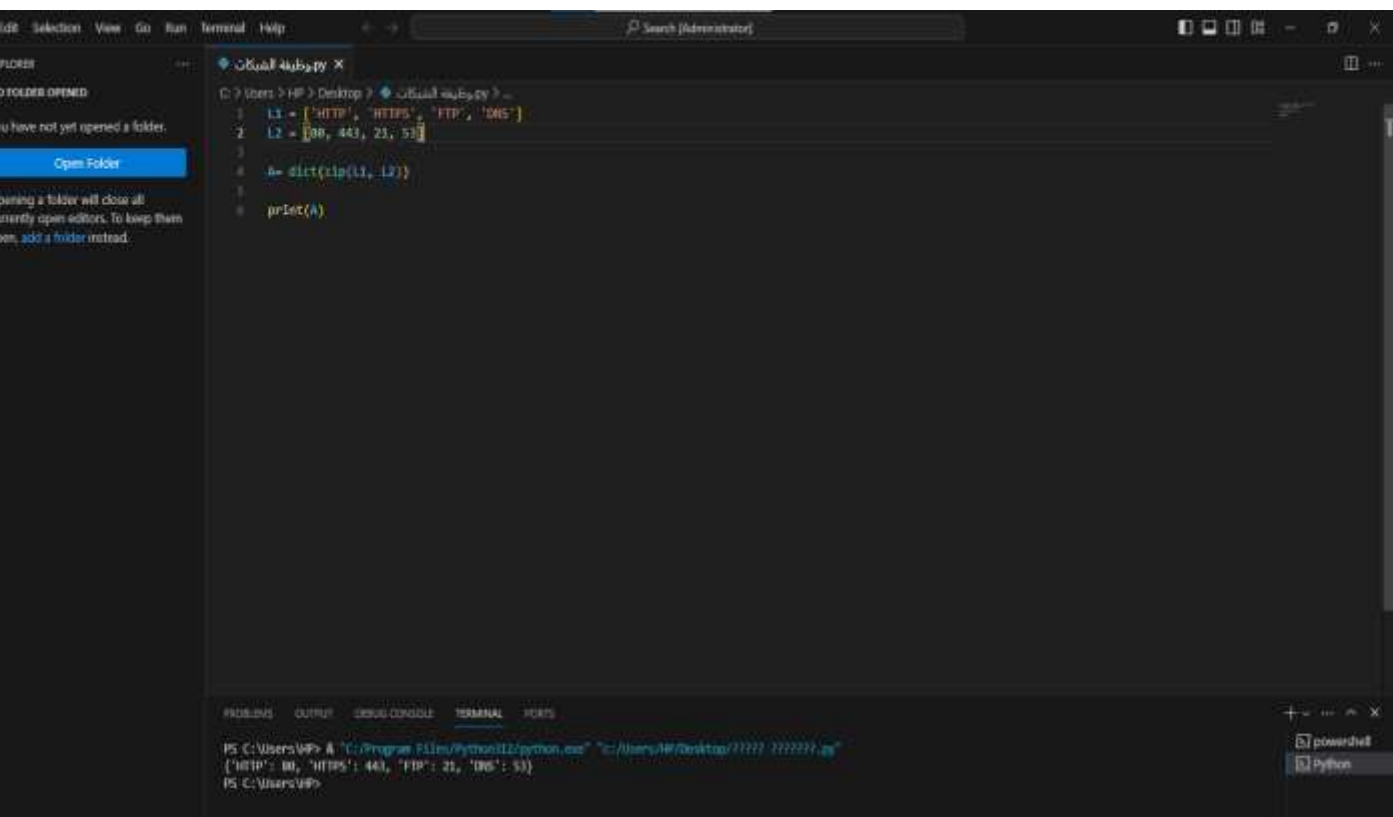
إشراف الدكتور:

مهند عيسى

Question 1:

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 }

```
L1 = ['HTTP', 'HTTPS', 'FTP', 'DNS']  
L2 = [80, 443, 21, 53]  
  
A= dict(zip(L1, L2))  
  
print(A)
```



The screenshot shows a Python IDE with a dark theme. The main editor window displays the following code:

```
1 L1 = ['HTTP', 'HTTPS', 'FTP', 'DNS']  
2 L2 = [80, 443, 21, 53]  
3  
4 A= dict(zip(L1, L2))  
5  
6 print(A)
```

The left sidebar shows a file explorer with a folder named 'C:\Users\VP\Desktop'. The bottom status bar shows the command prompt output:

```
PS C:\Users\VP> & "C:\Program Files\Python112\python.exe" "C:\Users\VP\Desktop\?????.py"  
{'HTTP': 80, 'HTTPS': 443, 'FTP': 21, 'DNS': 53}  
PS C:\Users\VP>
```

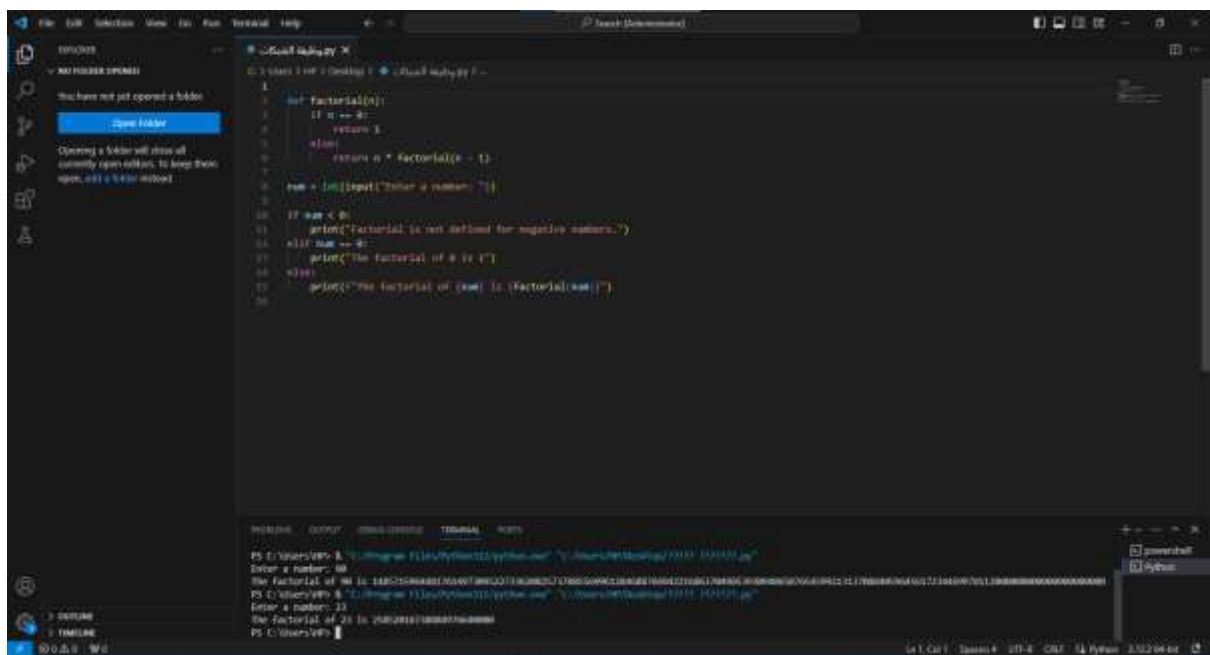
قمت باستخدام دالة zip() لربط العناصر من كلا القائمتين بحيث يتم إنشاء أزواج من العناصر، حيث يكون كل عنصر من القائمة الأولى مقرونًا بالعنصر المقابل من القائمة . تحويل الأزواج المرتبطة إلى قاموس باستخدام تعبير القاموس dict().

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

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)

num = int(input("Enter a number: "))

if num < 0:
    print("Factorial is not defined for negative numbers.")
elif num == 0:
    print("The factorial of 0 is 1")
else:
    print(f"The factorial of {num} is {factorial(num)}")
```



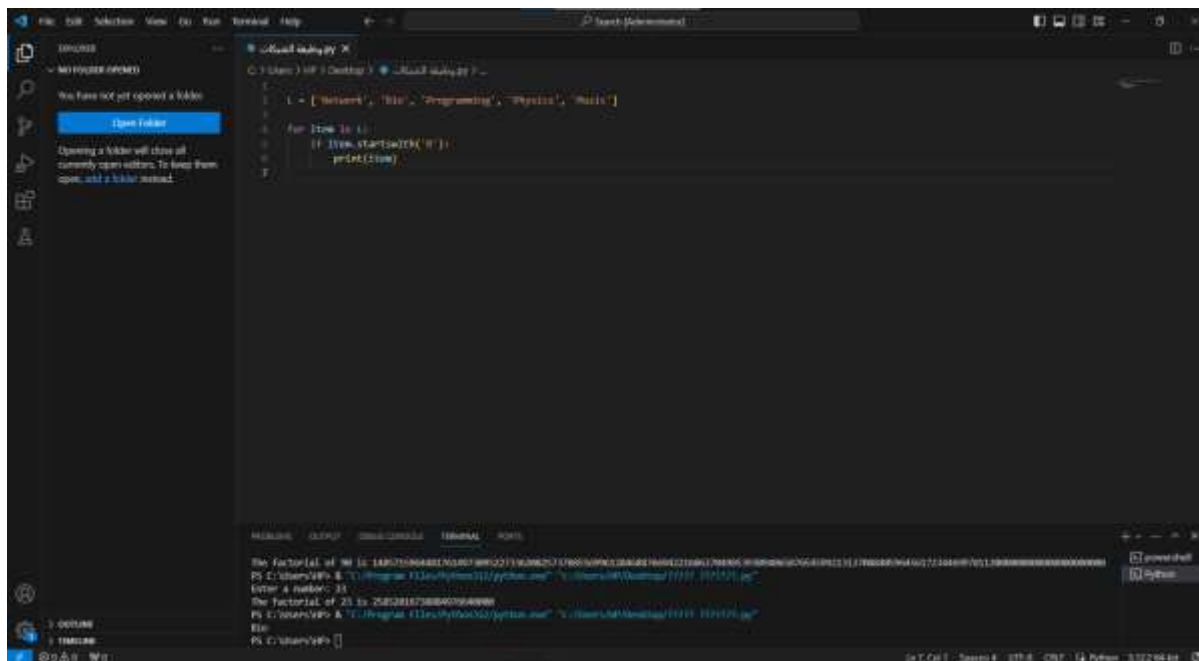
لحساب العامل لعدد مُدخل من المستخدم، 1: اطلب من المستخدم إدخال العدد ثم يتحقق من أن العدد المُدخل موجبًا لأن العامل يُحسب للأعداد الموجبة فقط قم بحساب العامل باستخدام حلقة ثم اطبع نتيجة العامل. عند التشغيل سيُطلب من المستخدم إدخال رقم عبر الخط الأمر لاحتساب العامل. إذا كان العدد موجبًا، سيقوم البرنامج بطباعة العامل للعدد.

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

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

for item in L:
    if item.startswith('B'):
        print(item)
```

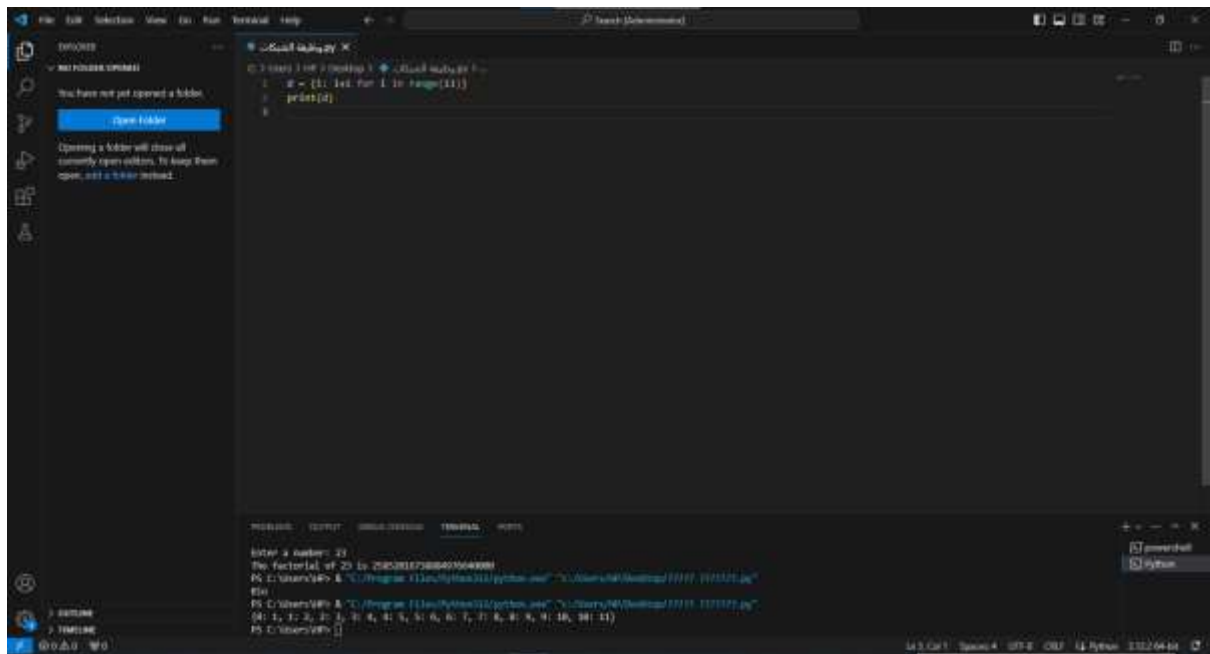


-نقوم بالتكرار على كل عنصر في القائمة L باستخدام حلقة - for لكل عنصر ، نتحقق إذا كان يبدأ بحرف 'B' باستخدام التابع startswith() إذا بدأ العنصر بحرف 'B' ، نقوم بطباعته على الشاشة . عند التشغيل سيقوم بإخراج كل العناصر من القائمة L التي تبدأ بحرف 'B'

D: Using Dictionary comprehension, Generate this dictionary

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

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



نستخدم مفهوم القاموس لإنشاء قاموس d , $\{i: i+1 \text{ for } i \text{ in range}(11)\}$

تتشكل أزواج مفتاح-قيمة حيث المفتاح هو الرقم i والقيمة هي $i+1$. دالة `range(11)` تنتج أرقاماً من 0 إلى 10 (شاملة) التي تُستخدم كمفاتيح في القاموس. عند التشغيل ، سيقوم بتوليد القاموس $d=\{0:1, 1:2, 2:3, 3:4, 4:5, 5:6, 6:7, 7:8, 8:9, 9:10, 10:11\}$ وطباعته على الشاشة

Question 2:

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

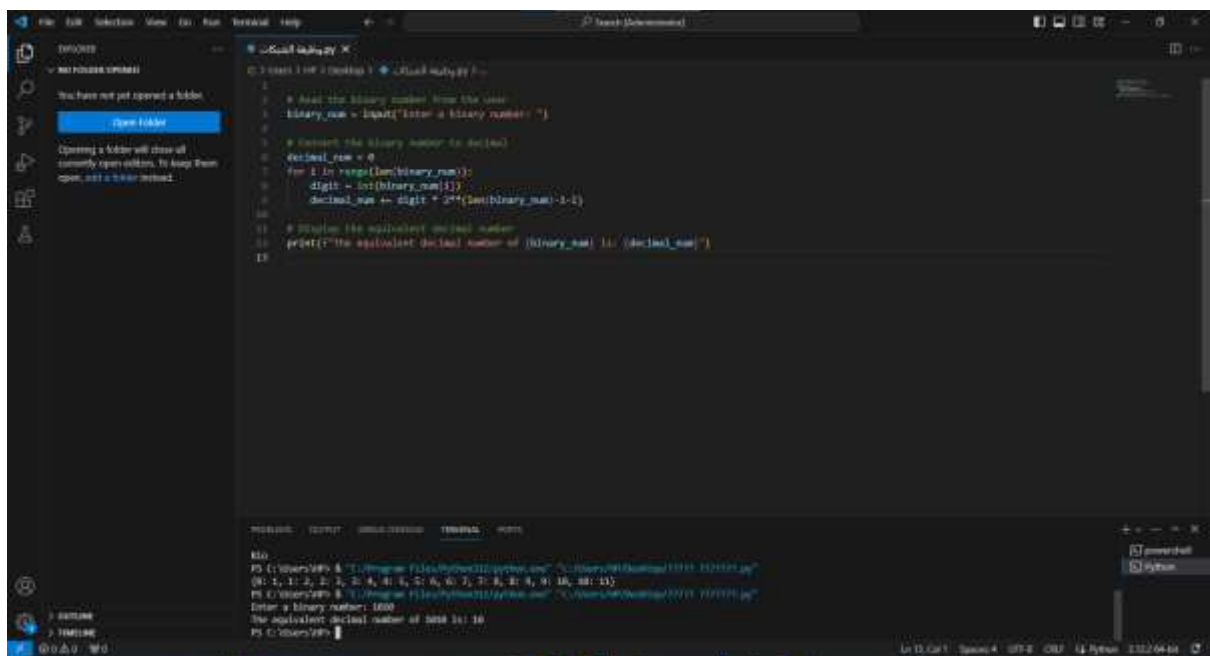
The program should start reading the binary number from the user. Then the decimal equivalent number must be calculated. Finally, the program must display the equivalent decimal number on the screen.

Tips: solve input errors.

```
# Read the binary number from the user
binary_num = input("Enter a binary number: ")

# Convert the binary number to decimal
decimal_num = 0
for i in range(len(binary_num)):
    digit = int(binary_num[i])
    decimal_num += digit * 2**(len(binary_num)-1-i)

# Display the equivalent decimal number
print(f"The equivalent decimal number of {binary_num} is: {decimal_num}")
```



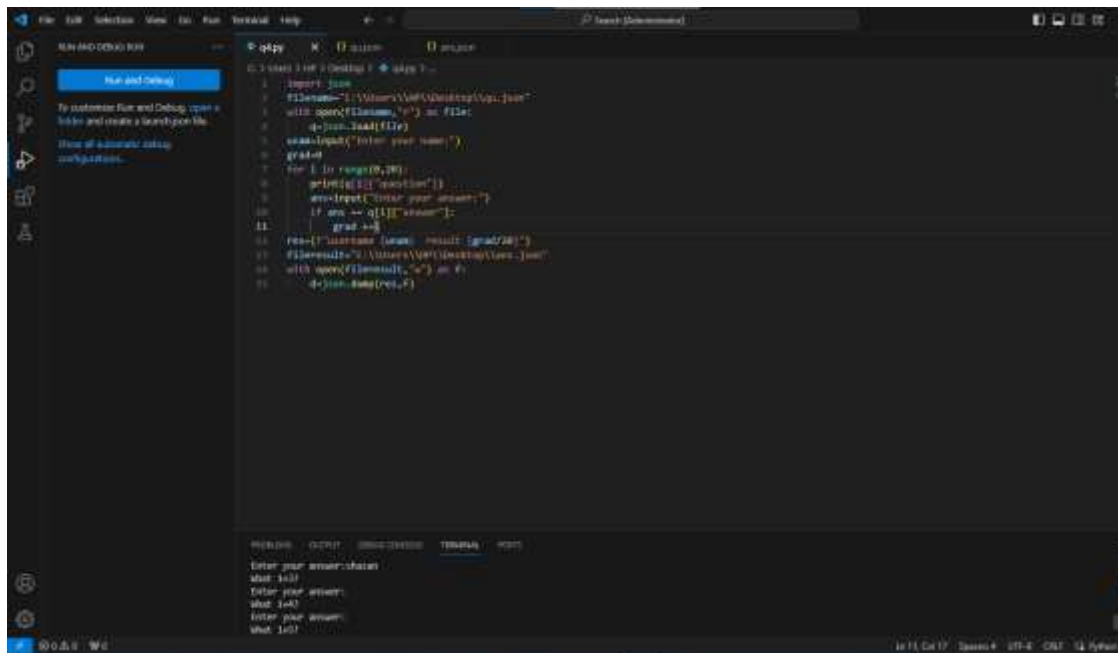
نقرأ الرقم الثنائي المدخل من المستخدم - ثم نقوم بتحويل الرقم الثنائي إلى ما يعادله العشري عن طريق تكرار كل رقم، ضربه في القوة المناسبة للعدد 2، وجمعها - ثم نعرض الرقم العشري المكافئ عند التشغيل يتم إدخال رقم ثنائي عندما يُطلب من المستخدم وسوف يخرج الرقم العشري المكافئ

Question 3:

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.

```
import json
filename="C:\\Users\\HP\\Desktop\\qu.json"
with open(filename,"r") as file:
    q=json.load(file)
unam=input("Enter your name:")
grad=0
for i in range(0,20):
    print(q[i]["question"])
    ans=input("Enter your answer:")
    if ans == q[i]["answer"]:
        grad +=1
res={f"username {unam} result {grad/20}"}
fileresult="C:\\Users\\HP\\Desktop\\ans.json"
with open(fileresult,"w") as f:
    d=json.dump(res,f)
```

يتم فتح ملف JSON الذي يحتوي على الأسئلة والإجابات باستخدام الدالة open مع وضع الوضع "قراءة" يتم استخدام json.load(file) لتحميل المحتوى من الملف إلى متغير q يُطلب من المستخدم إدخال اسمه يُبدأ في طباعة الأسئلة وطلب إدخال إجابة من المستخدم لكل سؤال .. إذا كانت الإجابة صحيحة، يتم زيادة المتغير grad بواحد يتم إنشاء قاموس res يحتوي على اسم المستخدم ونتيجته. يتم فتح ملف JSON جديد لكتابة النتيجة فيه .. يُقوم بكتابة النتيجة في الملف الجديد باستخدام json.dump(res, f).



Question 4:

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

```

class BankAccount:
    def __init__(self, account_number, account_holder):

```



```

        self.account_number = account_number
        self.account_holder = account_holder
        self.balance = 0.0

    def deposit(self, amount):
        self.balance += amount

    def withdraw(self, amount):
        if self.balance >= amount:
            self.balance -= amount
        else:
            print("Insufficient balance.")

    def get_balance(self):
        return self.balance

class SavingsAccount(BankAccount):
    def __init__(self, account_number, account_holder, interest_rate):
        super().__init__(account_number, account_holder)
        self.interest_rate = interest_rate

    def apply_interest(self):
        interest = self.balance * self.interest_rate
        self.balance += interest

    def print(self):
        print(f"Current balance: {self.balance}")
        print(f"Interest rate: {self.interest_rate}")

# Create an instance of BankAccount
bank_account = BankAccount("123456789", "John Doe")
bank_account.deposit(1000)
print(f"Current balance: {bank_account.get_balance()}")
bank_account.withdraw(500)
print(f"Current balance: {bank_account.get_balance()}")

# Create an instance of SavingsAccount
savings_account = SavingsAccount("987654321", "Jane Smith", 0.05)
savings_account.deposit(1000)
savings_account.apply_interest()
savings_account.print()

```


الحالي ومعدل الفائدة - .يتم إنشاء مثال من BankAccount برقم حساب واسم حامل الحساب. يتم إجراء إيداع بقيمة 1000 دولار، يليه سحب بقيمة 500 دولار. يتم طباعة الرصيد الحالي بعد كل عملية - .يتم إنشاء مثال من SavingsAccount برقم حساب، اسم حامل الحساب، ومعدل فائدة بنسبة 0.05. يتم إجراء إيداع بقيمة 1000 دولار، يطبق الفائدة على الرصيد، ويتم طباعة الرصيد الحالي ومعدل الفائدة