Syrian Arab Republic

Lattakia - Tishreen University

Department of Communication and electrical engineering

5th , Network Programming : Homework No1

الجمهورية العربية السورية

اللاذقية -جامعــة تشريـــن

كلية الهندسة الكهربانية والميكانيكية

قسم هندسة الاتصالات والالكترونيات

السنة الخامسة: وظيفة1 برمجة شبكات

الوظيفة الأولى

محمد أحمد علي

2039

السؤال الأول

-A

```
1-A.py - C:\2\1-A.py (3.11.2)
File  Edit  Format  Run  Options  Window  Help
d={}
L1=['HTTP','HTTPS','FTP','DNS']
L2=[80,443,20,53]
for i in range(len(L1)):
    d[L1[i]]=L2[i]
print(d)
```

الخرج:

```
IDLE Shell 3.11.2
File  Edit  Shell  Debug  Options  Window  Help
Python 3.11.2 (tags/v3.11.2:878ead1, Feb  7 2023, 16:38:35) [MSC v.1934 64 bi
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\2\1-A.py ==========
{'HTTP': 80, 'HTTPS': 443, 'FTP': 20, 'DNS': 53}
>>>
```

-B

```
le  Edit  Format  Run  Options  Window  Help
def factorial(n):
    if n < 0:
        return None
    elif n == 0:
        return 1
    else:
        return n * factorial(n - 1)

# Get input from the user
number = int(input("Enter a number: "))

# Calculate the factorial
result = factorial(number)

# Print the result
if result is None:
    print("Error: Factorial is undefined for negative numbers.")
else:
    print(f"The factorial of {number} is {result}.")
```

الخرج:

```
IDLE Shell 3.11.2
ile  Edit  Shell  Debug  Options  Window  Help
Python 3.11.2 (tags/v3.11.2:878ead1, Feb  7 2023, 16:38:35) [MSC v.19
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more informat
>>
========== RESTART: C:\2\1-B.py ==========
Enter a number: 5
The factorial of 5 is 120.
>>
========== RESTART: C:\2\1-B.py ==========
Enter a number: -10
Error: Factorial is undefined for negative numbers.
>>
========== RESTART: C:\2\1-B.py ==========
Enter a number: 0
The factorial of 0 is 1.
>>
```

-C

File Edit Format Run Options Window Help

```
L=['Network','Bio','Programming','Physics','Music']
#for loop to take each object in list
for i in L:
    #startswith() method for test if the string object start with capital 'B'
    if i.startswith('B'):
        print(i)
```

الخرج:

```
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>
========== RESTART: C:\2\1-C.py ==========
Bio
>>
```

-D

1-D.py - C:\2\1-D.py (3.11.2)

File Edit Format Run Options Window Help

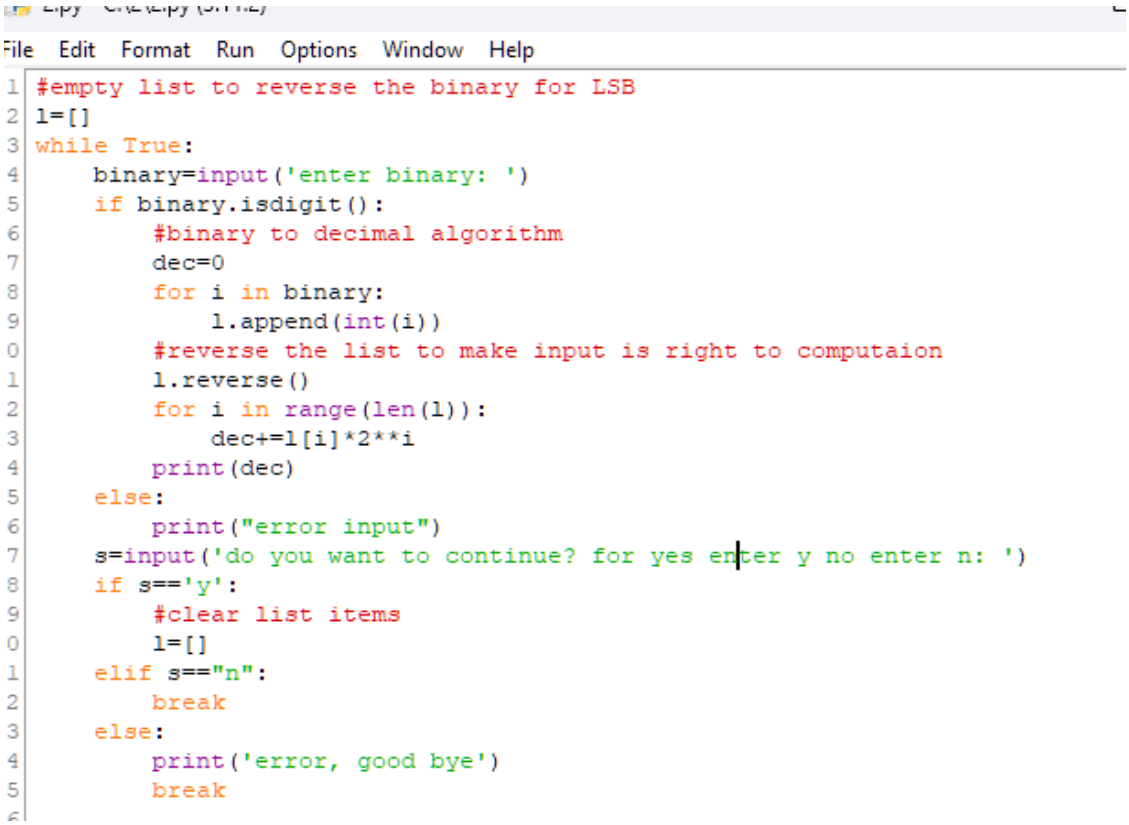
```
#dictionary comprehension
dict1={x:x+1 for x in range(11)}
print(dict1)
```

الخرج:

```
Type "help", "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\2\1-D.py ==========
{0: 1, 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 6: 7, 7: 8, 8: 9, 9: 10, 10: 11}
>>>
```

السؤال الثاني

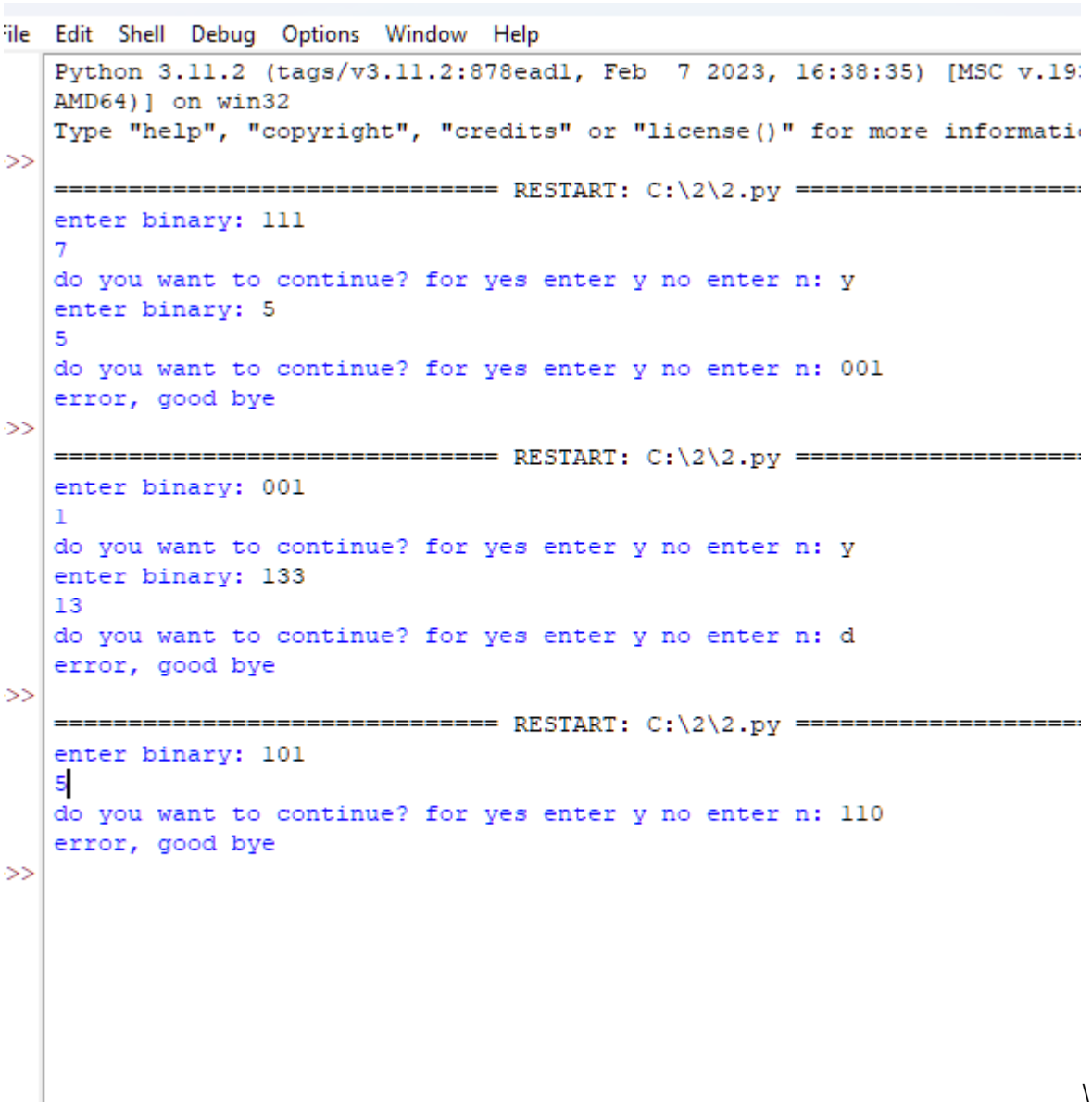
1. يتم إنشاء قائمة فارغة l لتخزين الأرقام الثنائية المعكوسة.
2. يتم دخول حلقة while True التي تعمل بشكل مستمر حتى يتم إيقافها.
3. يتم طلب إدخال رقم ثنائي من المستخدم باستخدام input() وتخزينه في المتغير binary.
4. يتم التحقق مما إذا كانت السلسلة المدخلة هي رقم ثنائي صالح باستخدام isdigit().
5. إذا كان الإدخال صالحاً، يتم تنفيذ خوارزمية تحويل الأعداد الثنائية إلى العشرية.
6. يتم إنشاء متغير dec وتعيينه بقيمة صفر.
7. يتم تحويل الأرقام الثنائية إلى أرقام صحيحة وإضافتها إلى القائمة l باستخدام حلقة for.
8. يتم عكس القائمة l حتى يكون الإدخال معكوساً وجاهزاً للحساب.
9. يتم استخدام حلقة for لحساب القيمة العشرية من الأعداد الثنائية باستخدام خوارزمية



File Edit Format Run Options Window Help

```
#empty list to reverse the binary for LSB
l=[]
while True:
    binary=input('enter binary: ')
    if binary.isdigit():
        #binary to decimal algorithm
        dec=0
        for i in binary:
            l.append(int(i))
        #reverse the list to make input is right to computaion
        l.reverse()
        for i in range(len(l)):
            dec+=l[i]*2**i
        print(dec)
    else:
        print("error input")
    s=input('do you want to continue? for yes enter y no enter n: ')
    if s=='y':
        #clear list items
        l=[]
    elif s=="n":
        break
    else:
        print('error, good bye')
        break
```

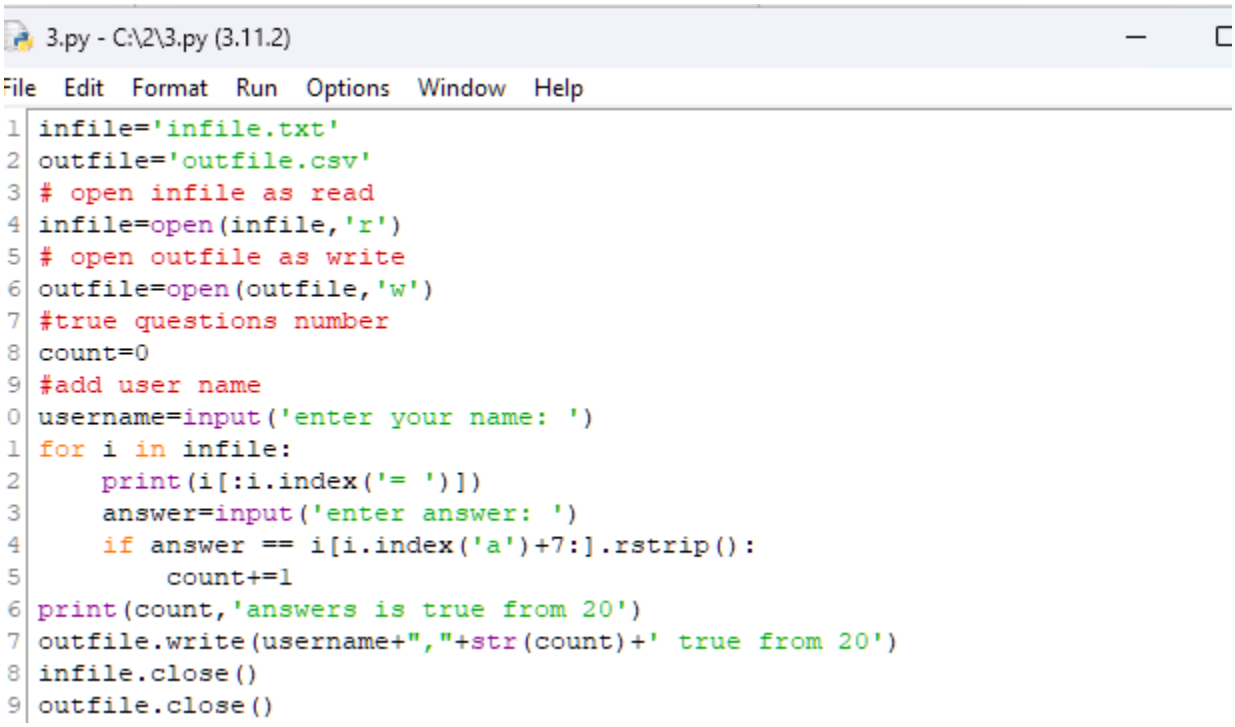
الخرج:


File Edit Shell Debug Options Window Help
Python 3.11.2 (tags/v3.11.2:878ead1, Feb 7 2023, 16:38:35) [MSC v.19
AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more informati
>>
========== RESTART: C:\2\2.py ==========
enter binary: 111
7
do you want to continue? for yes enter y no enter n: y
enter binary: 5
5
do you want to continue? for yes enter y no enter n: 001
error, good bye
>>
========== RESTART: C:\2\2.py ==========
enter binary: 001
1
do you want to continue? for yes enter y no enter n: y
enter binary: 133
13
do you want to continue? for yes enter y no enter n: d
error, good bye
>>
========== RESTART: C:\2\2.py ==========
enter binary: 101
5
do you want to continue? for yes enter y no enter n: 110
error, good bye
>>

\

السؤال الثالث

1. يتم تعريف اسم الملف الذي يحتوي على الأسئلة في المتغير infileباسم الملف infile.txt.
2. يتم تعريف اسم الملف الذي سيتم كتابة النتائج فيه في المتغير outfileباسم الملف outfile.csv.
3. يتم فتح الملف infileللقراءة باستخدام open(infile, 'r')ويتم تخزينه في المتغير infile.
4. يتم فتح الملف outfileللكتابة باستخدام open(outfile, 'w')ويتم تخزينه في المتغير outfile.
5. يتم تعريف متغير countبقيمة صفر لتعداد عدد الإجابات الصحيحة.
6. يُطلب من المستخدم إدخال اسمه باستخدام input('enter your name: ')ويتم تخزين الاسم في المتغير username.
7. يتم استخدام حلقة forلقراءة كل سطر في الملف infile.
8. يتم طباعة الجزء الأول من السؤال من السطر الحالي باستخدام print(i[:i.index('= ')]).
9. يُطلب من المستخدم إدخال إجابته باستخدام input('enter answer: ')ويتم تخزين الإجابة في المتغير answer.
10. يتم مقارنة الإجابة المدخلة مع الإجابة الصحيحة الموجودة في السطر



الملف الخاص بالاسئلة والاجوبة:

1x 1= answer 1

1x 2= answer 2

1x 3= answer 3

1x 4= answer 4

1x 5= answer 5

1x 6= answer 6

1x 7= answer 7

1x 8= answer 8

1x 9= answer 9

1x 10 = answer 10

2x 1= answer 2

2x 2= answer 4

2x 3= answer 6

2x 4= answer 8

2x 5= answer 10

2x 6= answer 12

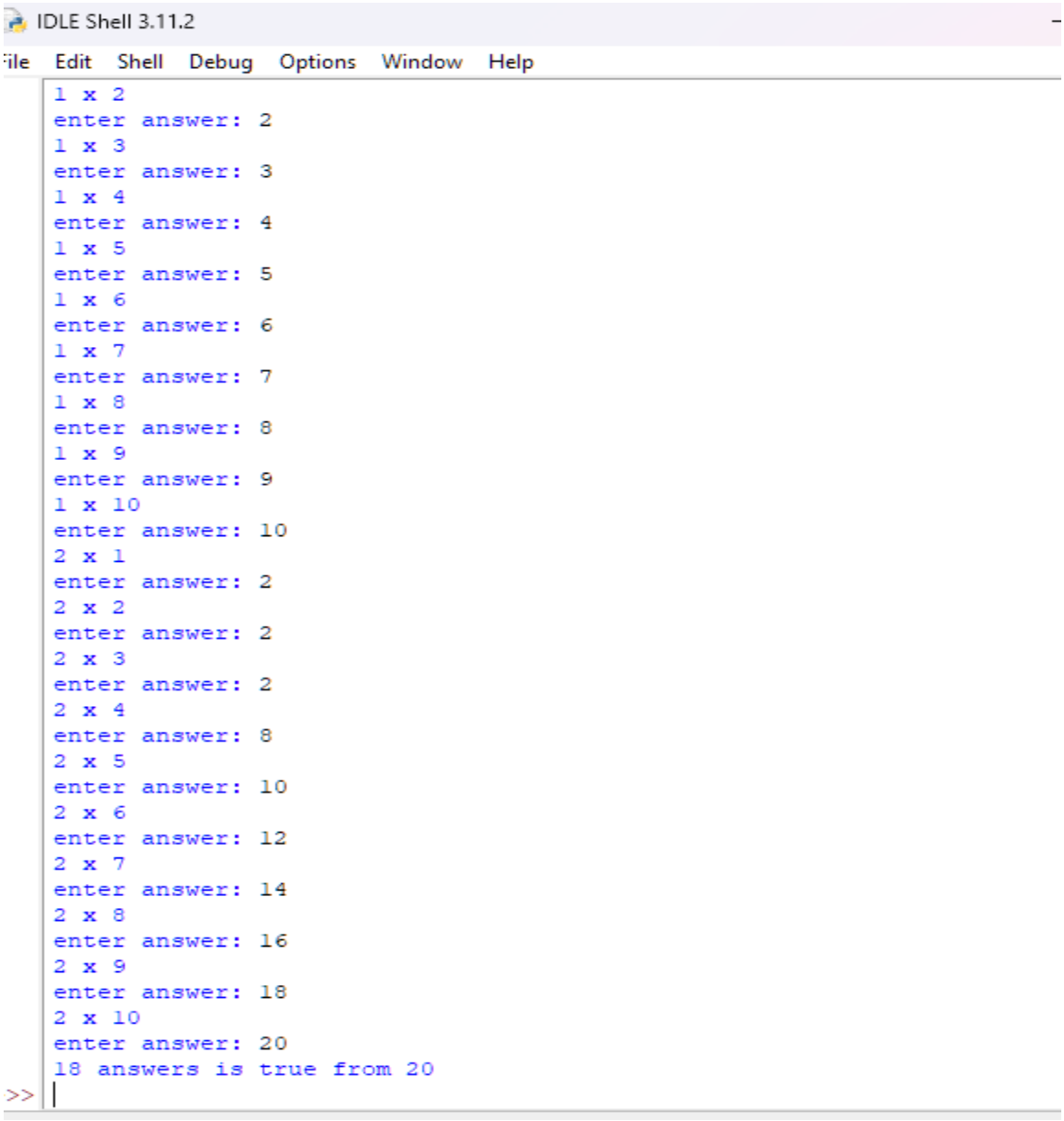
2x 7= answer 14

2x 8= answer 16

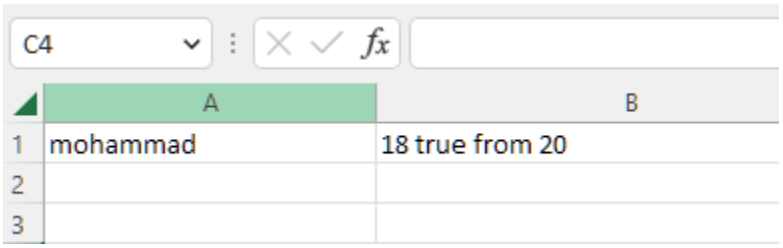
2x 9= answer 18

2x 10= answer 20

الخرج:



الملف الناتج



السؤال الرابع

File Edit Format Run Options Window Help

```python
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 amount <= self.balance:
            self.balance -= amount
        else:
            print("Insufficient funds.")

    def get_balance(self):
        return self.balance

    def __str__(self):
        return f"Account Holder: {self.account_holder}\nAccount Number: {self.account_number}\nBalance: {sel

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):
        self.balance += self.balance * (self.interest_rate / 100)

    def __str__(self):
        return f"Account Holder: {self.account_holder}\nAccount Number: {self.account_number}\nBalance: {sel


# Create an instance of BankAccount
bank_account = BankAccount("5555", "Mohammad Ali")

# Perform a deposit of $1000
bank_account.deposit(1000)
print(bank_account)
print()

# Perform a withdrawal of $500
bank_account.withdraw(500)
print(bank_account)
print()

# Create an instance of SavingsAccount
savings_account = SavingsAccount("6666", "Mohammad Mohammad", 5)
savings_account.deposit(200)
# Call apply_interest() and print() functions
savings_account.apply_interest()
print(savings_account)

```

يتم تعريف كلاس يسمى "BankAccount" ويحتوي على عدة أعضاء ووظائف. يستخدم هذا الكلاس لتمثيل حساب مصرفي.

يحتوي الكلاس "BankAccount" على المتغيرات التالية:

- "account_number": يستخدم لتخزين رقم الحساب المصرفي.

- "account_holder": يستخدم لتخزين اسم صاحب الحساب.
- "balance": يستخدم لتخزين الرصيد الحالي للحساب.

ويحتوي الكلاس "BankAccount" على الوظائف التالية:

- "**init**()": وظيفة البناء (Constructor) ، تستخدم لتهيئة الكائن وتعيين القيم الأولية للمميزات. تستقبل معاملين وهما "account_number" و "account_holder" وتعينهما للمتغيرات المناسبة.
- "deposit()": تستخدم لإيداع مبلغ محدد في الحساب. تستقبل معامل واحد وهو "amount" وتزيد القيمة المحددة من الميزة."balance"
- "withdraw()": تستخدم لسحب مبلغ محدد من الحساب. تستقبل معامل واحد وهو ."amount" إذا كان المبلغ المحدد أقل من أو يساوي الرصيد المتاح في الحساب، يتم خصم المبلغ من الميزة ."balance" إلا أنه إذا كان المبلغ أكبر من الرصيد المتاح، يتم طباعة رسالة."Insufficient funds."
- "get_balance()": تستخدم لاسترداد الرصيد الحالي للحساب.
- "**str**()": وظيفة تحويل الكائن إلى سلسلة نصية أي القيام ب override للتابع print. تُرجع سلسلة نصية تحتوي على معلومات الحساب مثل اسم صاحب الحساب، رقم الحساب والرصيد الحالي.

ثم يتم تعريف كلاس يسمى "SavingsAccount" والذي يرث من ."BankAccount" يضيف هذا الكلاس ميزة إضافية وهي "interest_rate" معدل الفائدة

ويحتوي الكلاس "SavingsAccount" على الوظائف التالية:

- "**init**()": وظيفة البناء (Constructor) ، تقوم بتهيئة الكائن وتعيين القيم الأولية للمميزات. تستقبل معاملين وهما "account_number" و "account_holder" و "interest_rate" وتستدعي وظيفة البناء في الكلاس الأساسي باستخدام."super().**init**(account_number, account_holder)"
- "apply_interest()": تستخدم لحساب الفائدة وإضافتها إلى الرصيد الحالي للحساب. تقوم بحساب المبلغ المستحق من الفائدة باستخدام الصيغة "self.balance * (self.interest_rate / 100)" وتضيفها إلى الميزة."balance"

في الجزء الأخير من الكود، يتم إنشاء مثال لكائن من الكلاس "BankAccount" باستخدام البيانات المحددة. يتم إيداع مبلغ قدره 1000 دولار في الحساب ويتم طباعة معلومات الحساب. ثم يتم سحب مبلغ قدره 500 دولار من الحساب ومرة أخرى يتم طباعة معلومات الحساب.

ثم يتم إنشاء مثال آخر لكائن من الكلاس "SavingsAccount" باستخدام البيانات المحددة. يتم إيداع مبلغ قدره 200 دولار في الحساب ويتم استدعاء وظيفة "apply_interest()" لحساب الفائدة وإضافتها إلى الرصيد. ثم يتم طباعة معلومات الحساب مرة أخرى.

الخرج:

```
============================== RES:
Account Holder: Mohammad Ali
Account Number: 5555
Balance: 1000.0

Account Holder: Mohammad Ali
Account Number: 5555
Balance: 500.0

Account Holder: Mohammad Mohammad
Account Number: 6666
Balance: 210.0
Interest Rate: 5
>>
```