التكليف الثاني

1-What is EOP (Event-driven programming)?

تستخدم البرمجة القائمة على الأحداث بشكل شائع في تطبيقات واجهة المستخدم الرسومية (GUI)

حيث تؤدي إجراءات المستخدم مثل نقرات الأزرار أو تحديدات القائمة إلى إجراءات أو تحديثات محددة على الشاشة.

تتضمن بعض المكتبات الشائعة التي تعتمد على الأحداث ما يلي:

1. JavaScript (DOM)

)نموذج كائن المستند) لتطوير الويب.

2)Java Swing and JavaFX

لتطوير تطبيقات سطح المكتب.

3) C # with Windows Forms or WPF

. Windows لتطوير تطبيقات

4) Python

.Tkinter أو PyQt مع مكتبات مثل لتطوير واجهة المستخدم الرسومية

بشكل عام ، تسمح البرمجة القائمة على الأحداث للمطورين بإنشاء تطبيقات يمكنها الاستجابة ديناميكيًا لتفاعلات المستخدم والأحداث الخارجية ، مما يوفر تجربة مستخدم تفاعلية وجذابة

2)code calculates the circumference of circle?

لحساب محيط الدائرة:

محيط الدائرة = 2 × ط × نصف قطر الدائرة

import math

def calculate\_circle\_circumference(radius)

circumference = 2 \* math.pi \* radius

return circumference

radius = float(input("أدخل نصف قطر الدائرة: "))

circumference = calculate\_circle\_circumference(radius)

print("محيط الدائرة هو:", circumference)

2)code calculates the space of circle?

لحساب مساحة الدائرة

مساحة الدائرة = نصف قطر الدائرة \* نصف قطر الدائرة \* ط

import math

def calculate\_area(radius):

area = math.pi \* radius\*\*2

return area

radius = float(input("أدخل قيمة نصف قطر الدائرة: "))

circle\_area = calculate\_area(radius)

print("مساحة الدائرة هي:", circle\_area)

2)code calculates the space of triangle?

لحساب مساحة المثلث:

مساحة المثلث = (قاعدة المثلث \* الارتفاع/2)

base = float(input("أدخل قاعدة المثلث: "))

height = float(input("أدخل الارتفاع: "))

area = (base \* height) / 2

print("مساحة المثلث هي:", area)

2)code calculates the perimeter of triangle?

كود يحسب محيط المثلث

محيط المثلث=مجموع أطوال أضلاعه

a = float(input("أدخل طول الضلع الأول: "))

b = float(input("أدخل طول الضلع الثاني: "))

c = float(input("أدخل طول الضلع الثالث: "))

perimeter = a + b + c

print("محيط المثلث هو:", perimeter)

2)code calculates the perimeter of rectangle?

محيط المستطيل= 2\*(الطول+العرض)

def calculate\_perimeter(length, width):

perimeter = 2 \* (length + width)

return perimeter

length = float(input("أدخل طول المستطيل: "))

width = float(input("أدخل عرض المستطيل: "))

perimeter = calculate\_perimeter(length, width)

print("محيط المستطيل هو:", perimeter)

2)code calculates the space of rectangle?

الطول \*العرض= مساحة المستطيل

length = float(input("أدخل طول المستطيل: "))

width = float(input("أدخل عرض المستطيل: "))

area = length \* width

print("مساحة المستطيل هي:", area)

2)code calculates the perimeter of square?

محيط المربع= 4 \* طول ضلعة

def calculate\_perimeter(side\_length):

perimeter = 4 \* side\_length

return perimeter

side\_length = float(input("أدخل طول ضلع المربع: "))

perimeter = calculate\_perimeter(side\_length)

print("محيط المربع هو:", perimeter)

2)code calculates the space of rectangle?

مساحة المربع =طول الضلع \* نفسة

side = float(input("أدخل طول ضلع المربع: "))

area = side \*\* 2

print("مساحة المربع هي:", area)

```

3)the different between Static Method and Class Method?

Class Method والـ Static Method

هما نوعان من الوظائف في البرمجة الشيئية، وهما يُستخدمان في (Classes) في اللغات التي تدعم هذا المفهوم مثلPython و Java وغيرها.

إليك الفرق بينهم:

1)Class Method (الوظيفة التابعة للصف):

تُعرف باستخدام مُشتقة (decorator) خاصة تسمى "@classmethod".

تتعامل مع نفسه وتمتلك إمكانية الوصول إلى الخصائص الساكنة (static properties) والوظائف الأخرى في class

تأخذ دائمًا المعامل الأول كمرجع إلى class نفسه، غالبًا يُسمى "cls".

يمكن استخدامها لإنشاء وتعديل خصائص تخص class بشكل عام.

class MyClass:

class\_variable = 0

@classmethod

def increment\_variable(cls):

cls.class\_variable += 1

2)Static Method (الوظيفة الثابتة):

- تُعرف باستخدام مُشتقة (decorator) خاصة تسمى "@staticmethod".

ليست مُرتبطة بأي معاملات خاصة بالصف أو بالمثيلات (instances) وتتعامل فقط مع المدخلات التي تُمر إليها.

لا تتيح لها الوصول إلى الخصائص الساكنة أو الوظائف الأخرى في الصف.

class MathOperations:

@staticmethod

def add(x, y):

return x + y

MathOperations.add(8,4)