

برمجة الحاسب الآلي

م. محمد أنس كواردة

الأكاديمية العربية الدولية – منصة أعد

مقدمة

برمجة الحاسب: هو عملية كتابة مجموعة من التعليمات تسمى code، مكتوبة بلغة برمجة محددة ليتم تنفيذها بواسطة الحاسب.

(اخبار الحاسب بما يجب عليه القيام به, بإعطائه سلسلة من التعليمات باستخدام لغة يفهمها الحاسب)

لغة البرمجة : مجموعة من القواعد و التعابير المحددة تستعمل لتوصيل التعليمات و الأوامر للحاسب بطريقة يفهمها و يقوم بتنفيذها.

مقدمة

الخوارزمية: مجموعة من محددة ومعرفة ومنتهاية من التعليمات لحل مشكلة محددة أو تنفيذ مهمة معينة، وهي عبارة عن مجموعة متسلسلة من التعليمات الواضحة لتحقيق هدف محدد.

خصائص الخوارزمية الجيدة

1. واضحة
2. محدودة الخطوات
3. حتمية
4. محددة الدخل و الخرج
5. عملية

لغات البرمجة

لغة البرمجة : مجموعة من القواعد و التعابير المحددة تستعمل لتوصيل التعليمات و الأوامر للحاسب بطريقة يفهمها و يقوم بتنفيذها.

يمكن تقسيم لغات البرمجة إلى نوعين رئيسيين

لغات البرمجة

لغات برمجة منخفضة المستوى: وهي تضمن فئتين رئيسيتين

- لغة الآلة: وهي اللغة الوحيدة التي يفهما الحاسب بشكل مباشر وتتألف من تسلسل من الأصفار و الواحدات، وتختلف من حاسب إلى آخر حسب بنيته.
- لغة التجميع **assembly**: تستخدم اختصارات معبرة من اللغة الإنكليزية لتعبر عن العمليات الرئيسية التي يقوم بها الحاسب و هي تحتاج محولا ليحولها إلى لغة الآلة يسمى **assembler** وتختلف تعليمات هذه اللغة بحسب بنية الحاسب المستخدم.

لغات البرمجة

لغات البرمجة عالية المستوى: تستخدم بعض الكلمات المفتاحية الشبيهة بالكلمات التي يستخدمها الإنسان في إعطاء الأوامر للحاسب، كما أنها مستقلة عن بنية الحاسب المستخدم من هذه اللغات (C، C++، JAVA، PYTHON، ...)، تتمثل فائدة استخدام لغة برمجة عالية المستوى في سهولة برمجة البرنامج وفهمه وصيانته.



الأكاديمية العربية الدولية
Arab International Academy

مقارنة بين اللغات منخفضة وعالية المستوى

لغات البرمجة المنخفضة المستوى Low Level Language	لغات البرمجة العالية المستوى High Level Language	
أكثر سهولة للحاسب لكي يفهمها	أكثر سهولة للمبرمجين	السهولة
أعلى كفاءة للذاكرة	أقل كفاءة من حيث الذاكرة	كفاءة التخزين في الذاكرة
يصعب على المبرمجين فهمها	سهولة الفهم للمبرمجين	سهولة الفهم للمبرمجين
معقدة في تحليل الأخطاء	أسهل في تحليل الأخطاء	سهولة التحليل للمبرمجين
أصعب نسبيًا في صيانتها	سهولة الصيانة	سهولة الصيانة
تعتمد على العتاد التي تستخدم عليه	يمكنها العمل على أي بيئة عمل	الاعتمادية على العتاد Hardware
تحتاج إلى مجمع Assembler لكي تعمل	تحتاج إلى مفسر أو مترجم لكي تعمل	المفسر / المجمع
أقل استخداما	تستخدم بشكل كبير في التطبيقات	الاستخدام

اللغات العالية المستوى

كما ذكرنا سابقاً فإن الحاسوب يفهم لغة واحدة فقط وهي لغة الآلة و بالتالي يجب تحويل الشيفرة المصدرية CODE المكتوب بأحد اللغات عالية المستوى إلى لغة الآلة و هذا يتم بطريقتين رئيسيتين

Compiled Languages: وفيها يتم تحويل الشيفرة المصدرية كاملة إلى لغة الآلة مباشرة بواسطة برنامج يسمى **المترجم Compiler** مثل لغة C، C++

Interpreted Languages: ويتمم فيها تنفيذ البرنامج سطر بسطر و تنفيذ كل أمر لوحده و هنا يستعمل برنامج يسمى **المفسر Interpreter** مثل لغة Python، PHP، Javascript.

Python

Python هي لغة برمجة، عالية المستوى سهلة التعلم مفتوحة المصدر قابلة للتوسيع، تعتمد أسلوب البرمجة ال غرضية التوجه.

لغة بايثون هي لغة مُفسّرة، ومُتعددة الاستخدامات، وتستخدم بشكل واسع في العديد من المجالات، كبناء البرامج المستقلة باستخدام الواجهات الرسومية وفي تطبيقات الويب و تطبيقات الذكاء الاصطناعي.

Python

يمكن استخدام بايثون لعمل البرامج البسيطة للمبتدئين، ولإنجاز المشاريع الضخمة في الوقت نفسه. غالباً ما يُنصح المبتدؤون في ميدان البرمجة بتعلم هذه اللغة لأنها من بين أسرع اللغات البرمجية تعلماً.

تستخدم بايثون الإزاحة بالفراغات بدلاً من الأقواس المعقوفة { }، و بالتالي فإن البرنامج المكتوب بلغة Python يكون سهل القراءة و الفهم بشكل كبير.

لماذا بايثون؟

تعمل Python على أنظمة تشغيل مختلفة (Windows و Mac و Linux...).

تحتوي لغة بايثون على بناء جملة بسيط مشابه للغة الإنجليزية.

تحتوي Python على بناء جملة يسمح للمطورين بكتابة برامج ذات سطور أقل

من بعض لغات البرمجة الأخرى.

لماذا بايثون؟

- تعمل Python على نظام مترجم ، مما يعني أنه يمكن تنفيذ التعليمات البرمجية بمجرد كتابتها. هذا يعني أنه يمكن بناء نماذج أولية بسرعة كبيرة.
- يمكن التعامل مع بايثون بطريقة إجرائية أو طريقة موجهة للكائنات أو بطريقة وظيفية.

إعداد بيئة العمل

1- تنصيب مفسر لغة بايثون: قبل الشروع في كتابة برنامج بلغة البايثون يجب تنصيب مفسر للغة

البايثون على الجهاز الذي نعمل عليه، يمكن الحصول على مفسر لغة البايثون من <https://www.python.org>

[./python.org](https://www.python.org)

2- تنصيب الـIDE: يمكن كتابة برامج بلغة البايثون بأية برنامج محرر نصوص و لكن سنستعمل محرر

Visual Studio Code **vscode**.

3- كتابة أول برنامج بلغة البايثون.

Python - Basic Syntax

تعرف Python مجموعة من القواعد تستعمل لكتابة البرنامج، و على الرغم من أنها لديها بعض الشبه مع بعض اللغات الأخرى ك JAVA ، C ولكن هناك تختلف بشكل كبير عن هذه اللغات.

المعرفات Identifier

- Identifier (المعرف): هو اسم يستخدم لتعريف متحول أو تابع صنف ...، وال-Identifier يجب أن يبدأ بالأحرف من A-Z أو a-z أو _، متبوعاً بصفر أو أكثر من الحروف و الأرقام وال- (شرطة سفلية).
- لا تسمح Python باستخدام أحرف الترقيم مثل @ و \$ و % داخل المعرفات.
- لا يمكن استخدام الكلمات المفتاحية لبايثون كمعرفات.
- لغة بايثون حساسة لحالة الأحرف و بالتالي cat, Cat معرفان مختلفان.

Identifier

أسماء معرفات صحيحة	أسماء معرفات غير صحيحة
score	@core
return_value	Return
highest_score	highest score
name1	1name
convert_to_string	convert to_string

Identifier

يجب دائما اعطاء المعارفات أسماء معبرة، مما يساعد في فهم البرنامج وسهولة صيانتة.

صحيح $c=10$

أكثر منطقية، وأكثر للفهم ,اسهل لمعرفة ماذا يمثل $count=10$

Python Reserved Words

- ☐ القائمة التالية تتضمن كلمات بايثون الأساسية، هذه كلمات محجوزة ولا يمكنك استخدامها كثابت أو متغير أو أي أسماء معرفات أخرى.
- ☐ جميع الكلمات المفتاحية لـ Python تحتوي على أحرف صغيرة فقط.

Python Reserved Words

and	as	assert	break	class	continue
def	del	elif	else	except	False
finally	for	from	global	if	import
in	is	lambda	None	nonlocal	not
or	pass	raise	return	True	try
while	with	yield			

Indentation in Python

```
j = 1  
while(j <= 5):  
    print(j)  
    j = j + 1
```

لا تستعمل بايثون الاقواس لتحديد كتل التعليمات عند تعريف التوابع أو عند التحكم في مسار البرنامج .

تستعمل بايثون المسافات البادئة للإشارة إلى كتل التعليمات, حيث أن جميع التعليمات التي تنتمي إلى نفس الكتلة البرمجية يجب أن يكون لها نفس المسافات البادئة.

Comments

التعليقات: هي شرح أو تعليقات توضيحية يمكن إضافتها إلى البرنامج.

يتم إضافتها بغرض تسهيل فهم الكود المصدري على البشر، ويتم تجاهلها بواسطة مترجم بايثون. تعزز التعليقات قابلية قراءة الكود وتساعد المبرمجين على فهم الكود بسهولة فائقة، إضافة شرح عن البرنامج.

Comments

تعليقات السطر واحد :حيث
يستعمل الرمز# للإشارة على بدء
التعليق و اعتبار كل النص الذي
يأت بعده هو تعليق يتم تجاهله
من المترجم حتى نهاية السطر.

```
7 # This is one line comment, and will be ignored by interpreter.  
8  
9 name='jone' # all text fter hash sign will be ignored
```

Comments

```
12      #this is comment line 1.  
13      #this is comment line 2.  
14      #this is comment line 3.  
15      #this is comment line 4.  
16  
17  
18      '''thi is multi line comment  
19      using triple-qouted string  
20      python intepreter will ignore it'''
```

تعليقات عدة أسطر: يمكن استعمال الهاش # عند بداية كل سطر من التعليق.

أو استعمال نص مع ثلاث علامات تنصيص قبله وبعده حيث سيتم تجاهله من قبل المفسر أيضاً.

المتحولات Variables

```
24 counter=100 # define integer variable
25 miles=10.0 #define floating variable
26 greeting ='Hellow World' #define string variable.
```

- المتغيرات هي أماكن في الذاكرة يتم تخزين البيانات فيها أثناء تشغيل البرنامج.
- لكل متغير اسم خاص يتم تحديده باستخدام قواعد المعارف المذكور سابقا.
- يتم اسناد قيم على المتغيرات باستعمال علامة المساواة = .
- يتم تحديد نوع بيانات المتغير تبعاً لنوع البيانات التي تم تخزينها فيه.

المتحولات Variables

تسمح بايثون بإسناد قيم واحدة إلى عدة متحولات بنفس الوقت

```
28 a = b = c = 100
```

كما تسمح بايثون بإسناد عدة قيم لعدة متحولات باستعمال تعليمة واحدة

```
name ,age ,weight = 'jone', 10 ,70.5
```

المتحولات Variables

لمعرفة نوع المتحول يمكن استخدام الدالة type()

```
39 name='my name'  
40 type(name)
```

يمكن حذف متحول باستخدام التعليم del

```
42 del(name)
```

تمت المحاضرة وبحمد الله