

BIẾN, PHÉP GÁN VÀ BIỂU THỨC SỐ HỌC

Học xong bài này, em sẽ:

- ✓ Nhận được vai trò của biến và phép gán.
- ✓ Đặt được tên cho biến, sử dụng được phép gán và cách đưa ra giá trị của biến trong Python.
- ✓ Làm quen được với cửa sổ Code trong Python để soạn thảo, lưu và thực hiện chương trình.



Khi giao cho máy tính giải quyết một bài toán, máy tính sẽ cần lưu trữ dữ liệu phục vụ cho quá trình thực hiện thuật toán giải bài toán đó. Em hãy lấy ví dụ về một bài toán đơn giản và chỉ ra những dữ liệu nào cần được lưu trữ, những dữ liệu nào sẽ thay đổi qua các bước xử lí của máy tính.

1 Biến và phép gán

a) Biến trong chương trình

Dù lập trình bằng ngôn ngữ nào, ta cũng phải biết sử dụng biến để lưu trữ dữ liệu cần thiết cho chương trình, nhất là những chương trình được thực hiện nhiều lần. Biến là tên một vùng nhớ; trong quá trình thực hiện chương trình, giá trị của biến có thể thay đổi.

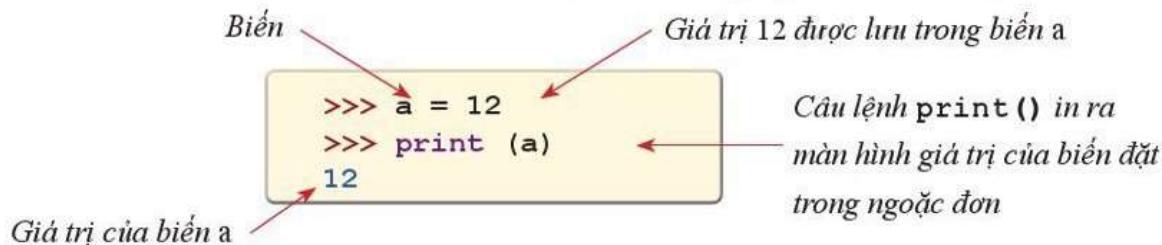


1

Em hãy chỉ ra các biến được sử dụng trong chương trình ở hình bên?



Hình 1 minh họa một ví dụ về chương trình trong Python sử dụng biến a.



Hình 1. Một chương trình Python

Lưu ý: Trong Python, các biến đều phải được đặt tên theo một số quy tắc.

- Không trùng với từ khoá (được sử dụng với ý nghĩa xác định không thay đổi) của Python (*Hình 2*).
- Bắt đầu bằng chữ cái hoặc dấu “_”.
- Chỉ chứa chữ cái, chữ số và dấu “_”.

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

Hình 2. Một số từ khoá thường dùng trong Python

Ví dụ 1.

n, delta, x1, Ab, _t12, Trường_sa là những tên biến đúng.

12t là tên biến sai (bắt đầu bằng chữ số).

A b là tên biến sai (chứa dấu cách).

Ab và **AB** là hai tên biến khác nhau.

b) Phép gán trong chương trình

Việc gán giá trị cho biến được thực hiện bằng phép gán (câu lệnh gán). Câu lệnh gán giá trị số học cho một biến là câu lệnh phổ biến nhất trong mọi chương trình ở mọi ngôn ngữ lập trình. Dạng đơn giản nhất của câu lệnh gán trong Python là:

Biến = <Biểu thức>

Phép gán được thực hiện như sau:

Bước 1. Tính giá trị biểu thức ở về phải.

Bước 2. Gán kết quả tính được cho biến ở về trái.

Ta thường gặp biểu thức số học ở về phải của một phép gán. Biểu thức số học có thể là một số, một tên biến hoặc các số và biến liên kết với nhau bởi các phép toán số học (xem *Bảng 1*). Trong biểu thức số học, có thể có các cặp ngoặc tròn xác định mức ưu tiên thực hiện phép tính tạo thành một biểu thức có dạng tương tự như cách viết trong toán học. Các phép toán được thực hiện theo thứ tự như trong toán học.

Bảng 1. Kí hiệu các phép toán số học trong Python

| Phép toán | Kí hiệu trong Python | Ví dụ |
|----------------------|----------------------|----------------|
| Cộng | + | $3 + 12 = 15$ |
| Trừ | - | $15 - 3 = 12$ |
| Nhân | * | $12 * 5 = 60$ |
| Chia | / | $16 / 5 = 3.2$ |
| Chia lấy phần nguyên | // | $16 // 5 = 3$ |
| Chia lấy phần dư | % | $16 \% 5 = 1$ |
| Luỹ thừa | ** | $2 ** 3 = 8$ |

Ví dụ 2. Thứ tự thực hiện phép tính trong biểu thức số học.

>>> (3 + 5) * 2
16

Phép cộng trong ngoặc thực hiện trước

>>> 3 + 5 * 2
13

Phép nhân thực hiện trước

Lưu ý:

- Trước và sau mỗi tên biến, mỗi số hoặc dấu phép tính có thể có số lượng tùy ý các dấu cách (dấu trắng).
- Trong biểu thức chỉ sử dụng các cặp ngoặc tròn để xác định thứ tự thực hiện các phép tính.

Ví dụ 3. Hai câu lệnh gán sau là tương đương:

v = (a * x + b) * x + c
v = (a*x+b)*x+c



2

Em hãy viết mỗi biểu thức toán học ở bảng bên thành biểu thức tương ứng trong Python.

$2a + 3b$

$xy : z$

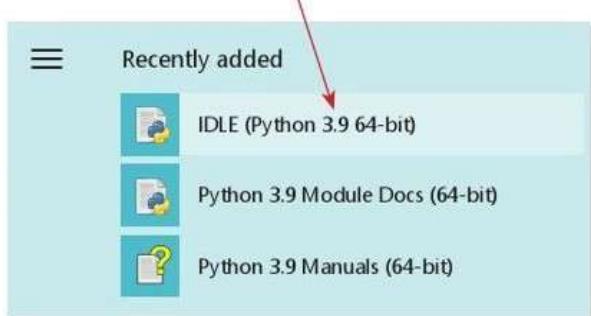
$b^2 - 4ac$

$(a:b)c$

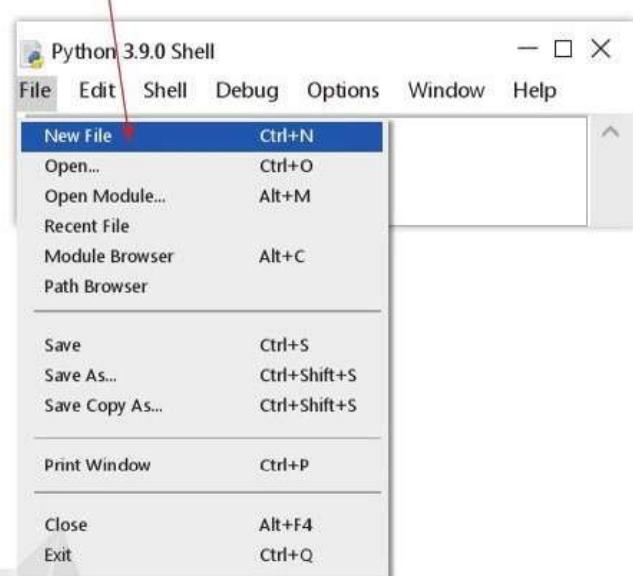
② Soạn thảo chương trình

Các môi trường ngôn ngữ lập trình bậc cao đều cho phép soạn thảo và lưu chương trình ở dạng tệp. Cửa sổ Shell của Python cho ta gõ và thực hiện ngay từng câu lệnh vừa đưa vào, nhưng không cho ta lưu lại những câu lệnh đã soạn thảo để thực hiện lại. Theo những bước được chỉ dẫn trong Hình 3. Cửa sổ để soạn thảo chương trình (còn gọi là cửa sổ Code) cho ta soạn thảo và lưu được tệp chương trình Python, chạy (thực hiện) chương trình này để thấy kết quả và có thể chỉnh sửa chương trình.

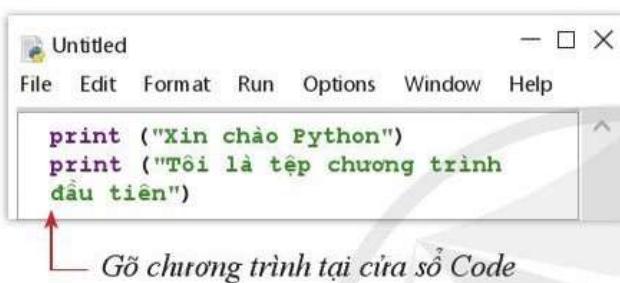
Bước 1. Khởi động IDLE



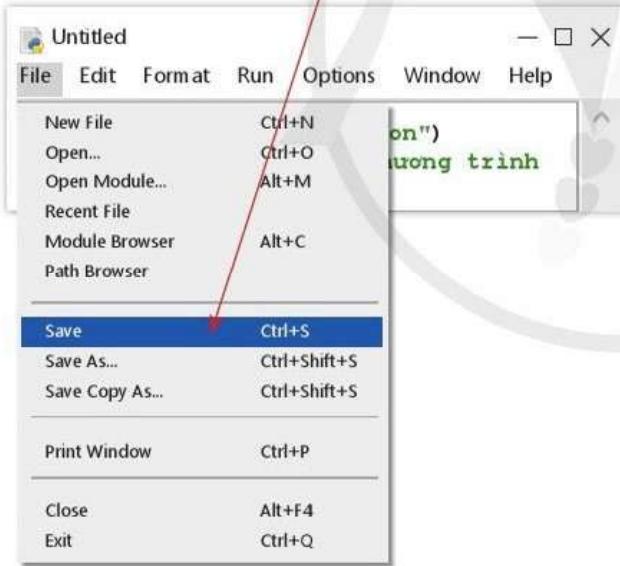
Bước 2. Mở tệp mới để soạn thảo chương trình



Bước 3. Soạn thảo chương trình

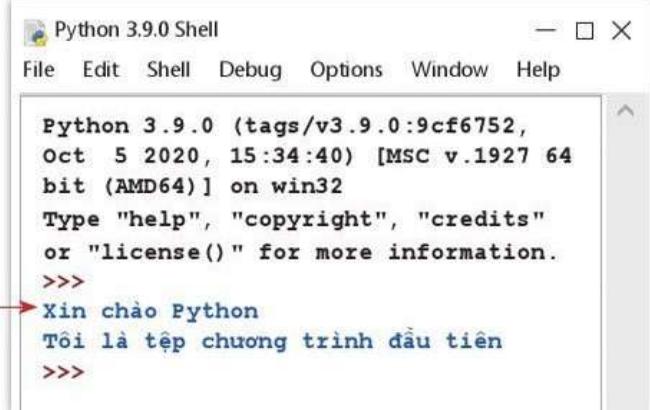
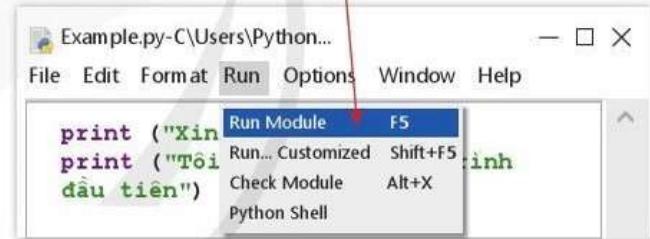


Bước 4. Lưu tệp chương trình



Cửa sổ Shell hiển thị kết quả chạy chương trình

Bước 5. Chạy chương trình



Hình 3. Các bước soạn thảo và thực hiện chương trình Python



Bài 1. Em hãy nêu ba tên biến đúng, ba tên biến sai. Với tên biến sai, em hãy giải thích tại sao đó không phải là tên biến.

Bài 2.

- 1) Ở cửa sổ Code, em hãy soạn thảo chương trình như trong *Hình 4*, chạy và cho biết kết quả hiển thị trên màn hình.
- 2) Thực hiện từng câu lệnh trong *Hình 4* ở cửa sổ Shell. Sau đó hãy thay phép nhân bằng một phép toán khác và xem kết quả.

```
File Edit Format Run Options Window Help
A = 123
B = 5
C = A * B
print(C)
```

Hình 4. Chương trình tính toán và đưa ra kết quả

Bài 3. Em hãy hoàn thiện chương trình ở *Hình 5* bằng cách viết biểu thức gán cho biến pound để nhận được chương trình chuyển đổi đơn vị đo khối lượng từ đơn vị ki-lô-gam sang pound, biết rằng 1 kg bằng 2,205 pound. Em hãy thay đổi giá trị gán cho biến kilo để chạy thử nghiệm chương trình.



Mảnh vườn trồng cúc đại đóa có chiều rộng m mét, chiều dài n mét. Mỗi mét vuông trồng được một khóm hoa. Mỗi khóm hoa bán được a nghìn đồng. Em hãy viết chương trình để đưa ra màn hình tổng số tiền thu được khi bán hết hoa trong vườn, với bộ dữ liệu đầu vào là m = 5, n = 18, a = 30.



Câu 1. Xét đoạn chương trình ở hình bên.

Em hãy cho biết c hay d nhận giá trị lớn hơn.

```
File Edit Format Run Options Window Help
kilo = 4.5
pound =
print(pound)
```

Hình 5. Chương trình chuyển đổi đơn vị đo khối lượng

```
a = 15.8
b = 6.2
c = a // b
d = a % b
```

Tóm tắt bài học

- ✓ Giá trị lưu trữ trong biến có thể thay đổi. Cần đặt tên biến theo các quy tắc của ngôn ngữ lập trình.
- ✓ Trong Python:
 - Câu lệnh gán có dạng: *Biến* = <Biểu thức>
 - Ở cửa sổ Shell máy tính thực hiện ngay từng câu lệnh.
 - Ở cửa sổ Code, ta có thể soạn thảo và lưu một tệp chương trình, chạy và chỉnh sửa chương trình.