# W1

ID: 764

Câu 1: Một trong những lợi ích chính của phần mềm mã nguồn mở so với phần mềm mã nguồn đóng là gì?

Đáp án:

A. Mã nguồn mở thường có giao diện người dùng thân thiện hơn.

B. Mã nguồn mở luôn có hiệu suất cao hơn mã nguồn đóng.

C. Mã nguồn mở có thể được tùy chỉnh theo nhu cầu cụ thể của người dùng.

D. Mã nguồn mở không bao giờ gặp lỗi hoặc lỗ hổng bảo mật.

ID: 776

Câu 2: Ai là người phát triển ngôn ngữ lập trình Python?

Đáp án:

A. Dennis Ritchie

B. Guido van Rossum

C. Bjarne Stroustrup

D. James Gosling

ID: 835

Câu 3: Framework Anaconda hỗ trợ cài đặt trực tiếp môi trường nào để phát triển Python?

Đáp án:

A. Visual Studio Code

B. PyCharm

C. Sublime Text

D. Jupyter Notebook

ID: 857

Câu 4: Trong Python, lệnh nào dùng để hiển thị thông tin ra màn hình?

Đáp án:

A. printf()

B. print()

C. cout

D. display()

ID: 876

Câu 5: Để lấy được địa chỉ ô nhớ lưu trữ giá trị của một biến trong Python, chúng ta sử dụng hàm nào?

Đáp án:

A. address()

B. memory()

C. id()

D. ptr()

ID: 902

Câu 6: Kiểu dữ liệu nào trong Python được dùng để lưu trữ các số nguyên?

Đáp án:

A. float

B. int

C. str

D. complex

ID: 936

Câu 7: Đâu là phát biểu đúng về phần mềm mã nguồn mở?

Đáp án:

A. Chỉ được sử dụng bởi các tổ chức lớn

B. Có mã nguồn được công khai và mọi người đều có thể xem, sửa đổi và phân phối lại

C. Có mã nguồn được bảo vệ bản quyền

D. Chỉ được phát triển bởi các nhà phát triển chuyên nghiệp

ID: 994

Câu 8: So sánh nào sau đây chính xác nhất về đặc điểm của mã nguồn mở so với mã nguồn đóng?

Đáp án:

A. Mã nguồn mở cho phép người dùng sửa đổi và phân phối mã nguồn, còn mã nguồn đóng thì không.

B. Mã nguồn mở chỉ dành cho những phần mềm nhỏ, còn mã nguồn đóng dành cho phần mềm lớn.

C. Mã nguồn mở luôn miễn phí, còn mã nguồn đóng thì luôn phải trả phí.

D. Mã nguồn mở không được bảo mật tốt như mã nguồn đóng.

ID: 1020

Câu 9: Sự khác biệt chính giữa mô hình phát triển mã nguồn mở và mã nguồn đóng là gì?

Đáp án:

A. Mã nguồn mở được phát triển bởi một nhóm nhỏ các lập trình viên chuyên nghiệp, còn mã nguồn đóng được phát triển bởi cộng đồng lập trình viên toàn cầu.

B. Mã nguồn mở cho phép bất kỳ ai cũng có thể đóng góp vào quá trình phát triển, còn mã nguồn đóng thì không.

C. Mã nguồn mở luôn có chất lượng cao hơn mã nguồn đóng.

D. Mã nguồn mở không yêu cầu bất kỳ chi phí nào để sử dụng, còn mã nguồn đóng luôn yêu cầu chi phí sử dụng.

ID: 1034

Câu 10: Tại sao Python được coi là ngôn ngữ lập trình dễ đọc và dễ học?

Đáp án:

A. Vì Python có cú pháp phức tạp và nhiều quy tắc

B. Vì Python nhấn mạnh vào tính đơn giản và rõ ràng của mã lệnh

C. Vì Python chỉ có thể sử dụng để lập trình hướng đối tượng

D. Vì Python không có tài liệu hỗ trợ

ID: 1046

Câu 11: Một trong những ưu điểm của Python so với các ngôn ngữ lập trình khác là gì?

Đáp án:

A. Python luôn phải trả phí để sử dụng

B. Python có hệ sinh thái phong phú với nhiều công cụ hỗ trợ lập trình viên

C. Python không có các thư viện hỗ trợ tính toán và đồ họa

D. Python chỉ được sử dụng để phát triển trang web

ID: 1065

Câu 12: Google Colab có ưu điểm gì nổi bật so với các môi trường phát triển khác?

Đáp án:

A. Cho phép sử dụng GPU miễn phí cho các dự án AI

B. Chỉ chạy được trên hệ điều hành Windows

C. Không hỗ trợ thư viện học máy nào

D. Yêu cầu cài đặt phần mềm trên máy tính

ID: 1112

Câu 13: Lệnh sau đây trong Python sẽ in ra kết quả gì? print('Python', 'is', 'fun', sep='- -')

Đáp án:

A. Python is fun

B. Python is fun sep=--

C. Python- -is- -fun

D. . Python- -is fun

ID: 1184

Câu 14: Kết quả khi thực thi đoạn mã sau là gì?print('Hello', 'world', sep=' ')print('Python', 'programming', sep='-')

Đáp án:

A. In ra "Helloworld" và "Python-programming"

B. In ra "Hello world" và "Python-programming"

C. In ra "Hello world" và "Python programming"

D. In ra "Helloworld" và "Python programming"

ID: 1199

Câu 15: Đoạn mã sau đây có chức năng gì?

x = input("Enter a number: ")

y = int(x) + 10

print("Result:", y)

Đáp án:

A. Nhập một số, chuyển đổi thành số nguyên, cộng thêm 10 và in ra kết quả

B. Nhập một số, trừ đi 10 và in ra kết quả

C. Nhập một chuỗi kí tự, cộng thêm 10 và in ra kết quả

D. Nhập một chuỗi, trừ đi 10 và in ra kết quả

ID: 1212

Câu 16: Python có cần khai báo kiểu dữ liệu cho biến trước khi sử dụng không?

Đáp án:

A. Có, giống như trong ngôn ngữ C

B. Có, giống như trong ngôn ngữ Java

C. Không, kiểu dữ liệu sẽ được nhận tự động dựa vào giá trị gán cho biến

D. Không, nhưng cần phải chỉ định kiểu dữ liệu

ID: 1228

Câu 17: Khi nào một ô nhớ chứa giá trị sẽ được giải phóng trong Python?

Đáp án:

A. Khi biến chỉ tới ô nhớ đó bị xóa

B. Khi biến được gán một giá trị mới

C. Khi chương trình kết thúc

D. Khi ô nhớ không còn biến nào chỉ tới nó

ID: 1241

Câu 18: Trong Python, điều gì sẽ xảy ra nếu hai biến được gán cùng một giá trị?

Đáp án:

A. Hai biến sẽ trỏ đến cùng một ô nhớ chứa giá trị đó

B. Một trong hai biến sẽ bị ghi đè giá trị của biến kia

C. Hai biến sẽ có hai ô nhớ khác nhau nhưng cùng giá trị

D. Python sẽ báo lỗi vì trùng giá trị

ID: 1251

Câu 19: Trong Python, câu lệnh nào để kiểm tra kiểu dữ liệu của một biến?

Đáp án:

A. type()

B. typeof()

C. dtype()

D. var\_type()

ID: 1262

Câu 20: n trong câu lệnh n=’5’ là kiểu dữ liệu nào?

Đáp án:

A. integer

B. string

C. tuple

D. operator

ID: 1274

Câu 21: Kết quả khi chạy đoạn mã sau là gì?a = 5

b = "Hello"

c = 3.14

print(type(a))print(type(b))print(type(c))

Đáp án:

A. int, int, int

B. int, str, float

C. float, str, int

D. str, str, str

ID: 1278

Câu 22: Biến nào dưới đây không hợp lệ trong Python?

Đáp án:

A. my\_var

B. \_myvar

C. 2myvar

D. myVar2

ID: 1281

Câu 23: Kết quả của biểu thức type(3.14) là gì?

Đáp án:

A. float

B. int

C. str

D. complex

ID: 1287

Câu 24: Lệnh nào dưới đây được dùng để nhập dữ liệu từ người dùng trong Python?

Đáp án:

A. scan()

B. getinput()

C. readline()

D. input()

ID: 1291

Câu 25: Trong Python, để xuất dòng chữ "Xin chào thế giới!" ra màn hình, bạn sử dụng lệnh nào sau đây?

Đáp án:

A. print("Xin chào thế giới!")

B. console.log("Xin chào thế giới!")

C. echo "Xin chào thế giới!"

D. System.out.println("Xin chào thế giới!")

ID: 1300

Câu 26: Kiểu dữ liệu nào dưới đây là kiểu dữ liệu cơ bản trong Python?

Đáp án:

A. int

B. str

C. float

D. Tất cả các phương án A, B, C đều đúng

ID: 1343

Câu 27: Trong Python, cách nào sau đây để khai báo biến và gán giá trị đúng?

Đáp án:

A. var x = 10

B. x := 10

C. let x = 10

D. x = 10

ID: 1371

Câu 28: Trong lập trình Python, cho đoạn chương trình sau:a=b=1

c,d=1,2

print(a+b+c+d)

Kết quả trên màn hình là:

Đáp án:

A. 3

B. 5

C. 4

D. 6

ID: 1395

Câu 29: Kết quả của đoạn mã sau là gì?

days = ['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday']

print(days[2])

Đáp án:

A. Monday

B. Wednesday

C. Tuesday

D. Thursday

ID: 80651

Câu 30: Đâu là tên phần mềm mã nguồn mở phổ biến?

Đáp án:

A. Microsoft Office

B. Linux

C. Adobe Photoshop

D. Google Chrome

# W2

ID: 1426

Câu 1: Trong Python, cấu trúc rẽ nhánh dạng thiếu là:

Đáp án:

A. if <điều kiện> :

B. if <điều kiện> then ;

C. IF <điều kiện> :

D. IF <điều kiện> THEN ;

ID: 1444

Câu 2: Khi sử dụng câu lệnh if-elif-else, phần elif được thực thi khi:

Đáp án:

A. Điều kiện if đúng

B. Điều kiện if sai và điều kiện elif đúng

C. Điều kiện if và điều kiện elif đều sai

D. Điều kiện if và điều kiện elif đều đúng

ID: 1479

Câu 3: Cấu trúc lặp với số lần biết trước có dạng:

Đáp án:

A. for <biến đếm> in range([giá trị đầu] to <giá trị cuối>, [bước nhảy]):

<lệnh>

B. for <biến đếm> in range([giá trị đầu], <giá trị cuối>, [bước nhảy]):

<lệnh>

C. for <biến đếm> in range([giá trị đầu], <giá trị cuối>, [bước nhảy]):

D. for <biến đếm> in range([giá trị đầu], <giá trị cuối>, [bước nhảy])

<lệnh>

ID: 1489

Câu 4: Công dụng chính của vòng lặp for trong Python là gì?

Đáp án:

A. Để thực hiện một hành động một lần

B. Để tạo ra một số ngẫu nhiên

C. Để thực hiện một hành động nhiều lần với từng phần tử trong một chuỗi

D. Để khai báo biến

ID: 1519

Câu 5: Cấu trúc lặp với số lần chưa biết trước có dạng:

Đáp án:

1. while <điều kiện> do

<câu lệnh>

1. while <điều kiện>:
2. while <điều kiện>:

<câu lệnh>

1. while <điều kiện>

<câu lệnh>

ID: 1535

Câu 6: Vòng lặp While kết thúc khi nào?

Đáp án:

A. Khi đủ số vòng lặp

B. Khi điều kiện thực hiện vòng lặp while được thỏa mãn

C. Khi tìm được Output

D. Tất cả các phương án A, B, C

ID: 1581

Câu 7: Trong NNLT Python, câu lệnh sau cho kết quả trên màn hình là gì?

if 1<2 and 1>3:

print("false")

else:

print("true")

Đáp án:

A. TRUE

B. FALSE

C. true

D. false

ID: 1623

Câu 8: Trong lập trình Python để kiểm tra số tự nhiên n khác 0 là số chẵn hay lẻ, câu lệnh nào sau đây là đúng?

Đáp án:

A. if n//2==0: print('so chan')

else: print('so le')

B. if n//2==1: print('so chan')

else: print('so le')

C. if n%2==1: print('so chan')

else: print('so le')

D. if n%2==0: print('so chan')

else: print('so le')

ID: 1640

Câu 9: Kết quả của đoạn mã sau là gì nếu x = -10?

x = -10

if x>0:

print("x là số dương")

elif x<0:

print("x là số âm")

else:

print("x là số không")

Đáp án:

A. In ra "x là số dương"

B. In ra "x là số không"

C. In ra "x là số âm"

D. Chương trình báo lỗi

ID: 1647

Câu 10: Đoạn mã sau có mục đích gì?

a = 3

b = 7

if a == b:

print("a bằng b")

elif a>b:

print("a lớn hơn b")

else:

print("a nhỏ hơn b")

Đáp án:

A. So sánh giá trị của a và b và in ra kết quả phù hợp

B. Gán giá trị lớn hơn cho a

C. Gán giá trị của b cho a nếu a nhỏ hơn b

D. Kiểm tra xem a có bằng b hay không

ID: 1659

Câu 11: Kết quả của đoạn mã sau là gì nếu

x = 10 và y = 20?

if x>y:

print("x lớn hơn y")

elif x<y:

print("x nhỏ hơn y")

else:

print("x bằng y")

Đáp án:

A. In ra "x lớn hơn y"

B. In ra "x nhỏ hơn y"

C. In ra "x bằng y"

D. Chương trình báo lỗi

ID: 1673

Câu 12: Kết quả khi chạy đoạn mã sau?

num = 15

if num % 3 == 0 and num % 5 == 0:

print("num chia hết cho cả 3 và 5")

else:

print("num không chia hết cho cả 3 và 5")

Đáp án:

A. Luôn in ra "num chia hết cho cả 3 và 5"

B. In ra "num chia hết cho cả 3 và 5" nếu num chia hết cho cả 3 và 5, ngược lại in ra "num không chia hết cho cả 3 và 5"

C. Luôn in ra "num không chia hết cho cả 3 và 5"

D. Chương trình báo lỗi

ID: 1689

Câu 13: Điều gì xảy ra khi chạy đoạn mã sau?

a = 3

b = 4

if a == 3 and b == 4:

print("a bằng 3 và b bằng 4")

elif a == 3 and b != 4:

print("a bằng 3 và b không bằng 4")

else:

print("a không bằng 3")

Đáp án:

A. In ra "a bằng 3 và b bằng 4"

B. In ra "a không bằng 3"

C. In ra "a bằng 3 và b không bằng 4"

D. Chương trình báo lỗi

ID: 1712

Câu 14: Để đưa ra số lớn nhất trong 2 số a, b ta viết câu lệnh như thế nào?

Đáp án:

A. if a>b:

Print (a)

B. if a=b:

print(a)

else:

print(b)

C. if a<b:

print(a)

else:

print(b)

D. if a>b:

print(a)

else:

print(b)

ID: 4015

Câu 15: Cho đoạn lệnh sau:

for i in range(5):

print(i)

Các giá trị của i trên màn hình là:

Đáp án:

A. 0 1 2 3 4 5

B. 1 2 3 4 5

C. 0 1 2 3 4

D. 1 2 3 4

ID: 4029

Câu 16: Cho đoạn lệnh sau:

for i in range(1,5):

print(i)

Trên màn hình i có các giá trị là:

Đáp án:

A. 0 1 2 3 4 5

B. 1 2 3 4 5

C. 0 1 2 3 4

D. 1 2 3 4

ID: 4038

Câu 17: Cho đoạn lệnh sau:

for i in range(1,10,2):

print(i)

Trên màn hình i có các giá trị là:

Đáp án:

A. 1,3,5,7,9

B. 1,2,3,4,5,6,7,8,9,10

C. 1,3,5,7,9,10

D. 1,3,5,7,10

ID: 4046

Câu 18: cho đoạn chương trình sau:

s=0

for i in range(6):

s = s+i

Sau khi thực hiện đoạn chương trình trên giá trị của s là:

Đáp án:

A. 1

B. 15

C. 6

D. 21

ID: 4049

Câu 19: Cho đoạn chương trình sau:

S = 0

for i in range(1,7,2):

s = s+i

Sau khi thực hiện đoạn chương trình trên giá trị của s là:

Đáp án:

A. 9

B. 15

C. 6

D. 21

ID: 4063

Câu 20: Cho đoạn code sau:

chuoi = ['ICTU', 'vi', 'yeu', 'ma', 'den']

tong\_chieu\_dai = 0

for tu in chuoi:

tong\_chieu\_dai += len(tu)

print("Tổng chiều dài các từ trong chuỗi là:", tong\_chieu\_dai)

Kết quả in ra sẽ là gì?

Đáp án:

A. Tổng chiều dài các từ trong chuỗi là: 5

B. Tổng chiều dài các từ trong chuỗi là: 15

C. Tổng chiều dài các từ trong chuỗi là: 14

D. Tổng chiều dài các từ trong chuỗi là: 20

ID: 4085

Câu 21: Sử dụng vòng lặp for, hãy viết đoạn code in ra các số từ 5 đến 15, chọn đáp án đúng.

Đáp án:

A. for i in range(5,15):

print(i)

B. for i in range(4,16):

print(i+1)

C. for i in range(4,16):

print(i)

D. for i in range(5,16,1):

print(i)

ID: 4103

Câu 22: Hãy viết đoạn code sử dụng vòng lặp for để in ra các số chẵn từ 2 đến 20, chọn đáp án đúng:

Đáp án:

A. for i in range(1, 20, 2):

print(i+1)

B. for i in range(2, 21, 2):

print(i)

C. for i in range(2, 21):

if i % 2 == 0:

print(i)

D. Tất cả các phương án trên đều đúng

ID: 4114

Câu 23: Tính tống S = 1 + 2 + 3 + … + n + … cho đến khi S > 10^9. Điều kiện nào sau đây cho vòng lặp while là đúng?

Đáp án:

A. While S <=10^9:

B. While S>=10^9:

C. While S=10^9:

D. While S !=10^9:

ID: 4184

Câu 24: Cho đoạn chương trình sau:

s=0

i=1

while i <=5:

s=s+1

i=i+1

print(s)

Sau khi thực hiện đoạn chương trình trên giá trị của s là:

Đáp án:

A. 9

B. 15

C. 5

D. 10

ID: 4197

Câu 25: Cho đoạn mã sau:

i = 1

while i <= 5:

if i % 2 == 0:

print(i, "is even")

i += 1

Kết quả của đoạn mã trên là gì?

Đáp án:

A. 1 is even, 3 is even, 5 is even

B. 2 is even, 4 is even

C. 1 is even, 2 is even, 3 is even, 4 is even, 5 is even

D. Không có kết quả gì được in ra

ID: 4209

Câu 26: Cho đoạn mã như sau:

n = 5

while n>0:

print(n)

n -= 2

Kết quả của đoạn mã trên là gì?

Đáp án:

A. 5, 4, 3, 2, 1

B. 5, 3, 1

C. 5, 3

D. 5, 3, 1, -1

ID: 4217

Câu 27: Cho đoạn mã:

x = 0

while x<4:

x += 1

if x == 3:

continue

print(x)

Kết quả của đoạn mã trên là gì?

Đáp án:

A. 1, 2, 4

B. 1, 2, 3, 4

C. 0, 1, 2, 4

D. 1, 2, 3

ID: 4226

Câu 28: Kết quả đoạn mã sau là gì?

i = 0

while i<3:

print(i)

i += 1

else:

print("Done")

Đáp án:

A. 0, 1, 2, Done

B. 0, 1, Done

C. 1, 2, Done

D. Done, 0, 1, 2

ID: 4236

Câu 29: Cho đoạn mã sau:

n = 0

while n<6:

if n == 4:

break

print(n)

n += 1

Kết quả của đoạn mã trên là gì?

Đáp án:

A. 0, 1, 2, 3

B. 0, 1, 2, 3, 4, 5

C. 0, 1, 2

D. Không có gì được in ra

ID: 4252

Câu 30: Cho đoạn mã:

x = 10

while x>0:

x -= 3

print(x)

Kết quả của đoạn mã trên là gì?

Đáp án:

A. 7, 4, 1, -2

B. 10, 7, 4, 1, -2

C. 7, 4, 1

D. 10, 7, 4, 1

# W3

ID: 1802

Câu 1: Hàm trong Python là gì?

Đáp án:

A. Một khối lệnh thực hiện một công việc hoàn chỉnh và có thể được gọi lại nhiều lần

B. Một biến được sử dụng trong chương trình

C. Một loại dữ liệu trong Python

D. Một thư viện của Python

ID: 1810

Câu 2: Biến global trong Python là gì?

Đáp án:

A. Biến được khai báo trong hàm

B. Biến có phạm vi ảnh hưởng toàn cục trong chương trình

C. Biến chỉ có thể được sử dụng trong hàm

D. Biến chỉ được sử dụng một lần

ID: 1816

Câu 3: Hàm thư viện là gì?

Đáp án:

A. Hàm do người dùng định nghĩa

B. Hàm không có kết quả trả về

C. Hàm được xây dựng sẵn trong Python và cần khai báo thư viện chứa nó

D. Hàm chỉ có một tham số

ID: 1826

Câu 4: Tham số mặc định là gì?

Đáp án:

A. Tham số bắt buộc phải có giá trị khi gọi hàm

B. Tham số có giá trị được cung cấp sẵn và không cần truyền giá trị khi gọi hàm

C. Tham số chỉ xuất hiện trong hàm đệ quy

D. Tham số không có giá trị mặc định

ID: 1836

Câu 5: Làm thế nào để gọi một hàm có đối số và kết quả trả về?

Đáp án:

A. FunctionName([parameter])

B. def FunctionName([parameter])

C. result = FunctionName([parameter])

D. return FunctionName([parameter])

ID: 1852

Câu 6: Hàm nào sau đây được sử dụng để làm tròn số trong Python?

Đáp án:

A. round

B. exit

C. random

D. eval

ID: 1864

Câu 7: Hàm eval trong Python dùng để làm gì?

Đáp án:

A. Để lấy số ngẫu nhiên

B. Để thoát chương trình

C. Để làm tròn số

D. Để tính giá trị của một biểu thức chuỗi

ID: 4290

Câu 8: Viết hàm để tính giai thừa của một số n sử dụng đệ quy. Điều kiện dừng của hàm là gì?

Đáp án:

A. Khi n = 0

B. Khi n = 1

C. Khi n = 2

D. Khi n = 10

ID: 4295

Câu 9: Khi nào cần sử dụng từ khóa global trong hàm?

Đáp án:

A. Khi muốn in giá trị của biến toàn cục

B. Khi muốn thay đổi giá trị của biến toàn cục trong hàm

C. Khi muốn khai báo biến cục bộ

D. Khi không có biến toàn cục nào

ID: 4301

Câu 10: Viết một hàm tính tổng của các chữ số của một số nguyên n. Kết quả của hàm là gì nếu n = 1234?

Đáp án:

A. 10

B. 1234

C. 6

D. 24

ID: 4308

Câu 11: Viết hàm để kiểm tra xem một số có phải là số nguyên tố hay không. Điều kiện nào sau đây đúng để một số n là số nguyên tố?

Đáp án:

A. n > 1 và không chia hết cho bất kỳ số nào ngoài 1 và chính nó

B. n > 0

C. n chia hết cho 2

D. n không chia hết cho bất kỳ số nào ngoài 1

ID: 4317

Câu 12: Khi nào nên sử dụng hàm đệ quy thay vì vòng lặp?

Đáp án:

A. Khi bài toán có thể giải quyết một cách tự nhiên bằng cách chia nhỏ vấn đề thành các bài toán con

B. Khi bài toán đơn giản

C. Khi không thể sử dụng vòng lặp

D. Khi muốn giảm thiểu bộ nhớ

ID: 4323

Câu 13: Kết quả của chương trình sau là:

def PhepNhan(Number):

return Number \* 10;

print(PhepNhan(5))

Đáp án:

A. 50

B. 10

C. Chương trình lỗi

D. 5

ID: 4336

Câu 14: Xét đoạn chương trình sau:

def meeting():

ten=input("Nhập họ tên của em: ")

print("Xin chào",ten)

meeting()

Khi chạy chương trình, nếu nhập Lê Hiếu Học thì kết quả là:

Đáp án:

A. Xin chào Lê Hiếu Học

B. Lê Hiếu Học

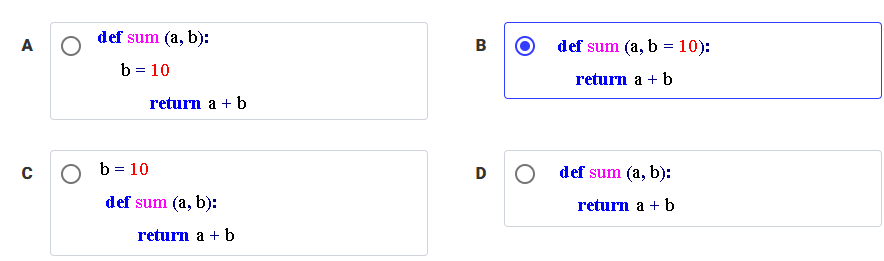
C. Xin chào ten

D. Xin chào em Lê Hiếu Học

ID: 4370

Câu 15: Để viết một hàm tính tổng hai số nguyên a và b, với giá trị mặc định b=10, đoạn code nào dưới đây là đúng?

Đáp án:

ID: 4376

Câu 16: Kết quả của đoạn lệnh sau là gì?

def sum (a, b = 10):

return a + b

result1 = sum(5)

print("Result 1:", result1)

Đáp án:

A. Result 1: 5

B. Result 1: 15

C. Result 1: 10

D. lỗi không trả về kết quả

ID: 4385

Câu 17: Kết quả của đoạn lệnh sau là gì?

def sum (a, b = 10):

return a + b

result2 = sum(3, 7)

print("Result 2:", result2)

Đáp án:

A. Result 2: 3

B. Result 2: 7

C. Result 2: 10

D. Result 2: 13

ID: 4395

Câu 18: Cho đoạn chương trình sau:

def sum(a , b):

return a + b

a = 5

b = 10

s = sum(a, b)

print("{0}+{1}={2}".format(a,b,s))

Kết quả in ra màn hình của đoạn chương trình trên là gì?

Đáp án:

A. a + b = s

B. a + b = 15

C. 5 + 10 = 15

D. s = 15

ID: 4399

Câu 19: Cho đoạn chương trình sau:

def sum (a = 5, b = 10):

return a + b

result2 = sum()

print("Result 2:", result2)

Kết quả in ra màn hình là gì?

Đáp án:

A. Result 2: 15

B. Result 2: 0

C. Result 2: 10

D. Result 2: 5

ID: 4414

Câu 20: Cho đoạn chương trình sau:

def sum (a=10, b):

return a + b

result2 = sum(7)

print("Result 2:", result2)

Kết quả in ra màn hình là:

Đáp án:

A. Result 2: 10

B. Result 2: 17

C. Result 2: 2

D. SyntaxError

ID: 4420

Câu 21: Giả sử ta có a=3, b=11, c=b/a. Để in kết quả của c ra màn hình là một số được làm tròn đến 2 chữ số thập phân, ta sử dụng câu lệnh:

Đáp án:

A. print(round(c,2))

B. print(round(2))

C. print(round(c;2))

D. round(c,2)

ID: 4425

Câu 22: Để lấy được một số ngẫu nhiên nằm trong khoảng (a,b) với a=10, b=100 ta dùng lệnh:

Đáp án:

A. random.randrange(10, 100[)(sai nhưng đúng với đề lms hiện tại)

B. random.randrange(a,b)

C. random(a, b)

D. random(10, 100)

ID: 4460

Câu 23: Để tạo và in ra một số nguyên ngẫu nhiên nằm trong khoảng từ 1 đến 10, ta dùng các lệnh nào?

Đáp án:

A. import random

val = random.randint(1,10)

print(val)

B. val = random.randint(1,10)

print(val)

C. import random

val = random.randint(1,10)

D. import random

val = random.randint(10)

print(val)

ID: 4473

Câu 24: Cho câu lệnh Python như sau:

print ("min('python', 'java', 'php') : ", min('python', 'java', 'php'))

Kết quả câu lệnh trên là gì?

Đáp án:

A. min('python', 'java', 'php') : php

B. min('python', 'java', 'php') : java

C. php

D. java

ID: 4480

Câu 25: Cho câu lệnh Python như sau:

print ("min('helloworld') : ", min("helloworld"))

Kết quả câu lệnh trên là:

Đáp án:

A. min('helloworld') : d

B. min('helloworld') : h

C. min('helloworld') : l

D. min('helloworld') : o

ID: 4487

Câu 26: Cho câu lệnh Python như sau: print ("max('python') : ", max("python"))Kết quả câu lệnh trên là:

Đáp án:

A. max('python') : p

B. max('python') : n

C. max('python') : t

D. max('python') : y

ID: 4496

Câu 27: Cho câu lệnh Python như sau: print ("max('python', 'java', 'php') : ", max('python', 'java', 'php')) Kết quả câu lệnh trên là:

Đáp án:

A. max('python', 'java', 'php') : python

B. max('python', 'java', 'php') : java

C. max('python', 'java', 'php') : 'php'

D. max('python', 'java', 'php') : ‘python’

ID: 4507

Câu 28: Cho đoạn chương trình như sau:

g=5

def increment():

g=g+1

increment()

print(g)

Kết quả khi chạy đoạn chương trình trên là gì?

Đáp án:

A. 6

B. Lỗi biến g ở dòng thứ 3

C. 5

D. Lỗi biến g ở dòng thứ 5

ID: 4534

Câu 29: Kết quả của đoạn mã sau là gì?

def multiply(a, b):

return a \* b

result = multiply(3, 4)

print(result)

Đáp án:

A. 12

B. 7

C. 34

D. 0

ID: 4593

Câu 30: Cho câu lệnh Python như sau:

print ("min('hello', 'hi', 'nice') : ", min('hello', 'hi', 'nice'))

Kết quả câu lệnh trên là:

Đáp án:

A. min('hello', 'hi', 'nice') : hello

B. min('hello', 'hi', 'nice') : hi

C. min('hello', 'hi', 'nice') : nice

D. min('hello', 'hi', 'nice') : ‘hello’

# W4

ID: 9994

Câu 1: Câu lệnh để khởi tạo một danh sách rỗng trong Python?

Đáp án:

A. list={ }

B. list = [ ]

C. list = ( )

D. list = set ( )

ID: 9995

Câu 2: Phương thức nào dưới đây được sử dụng để thêm một phần tử vào cuối danh sách?

Đáp án:

A. add()

B. append()

C. insert ()

D. extend()

ID: 9996

Câu 3: Các phần tử trong List được đặt trong cặp dấu ngoặc nào?

Đáp án:

A. Ngoặc đơn ( )

B. Ngoặc vuông [ ]

C. Ngoặc nhọn { }

D. Ngoặc kép “ ”

ID: 9997

Câu 4: Kết quả của đoạn mã sau là gì?fruits = ["apple", "banana", "cherry"]print(fruits[1])

Đáp án:

A. apple

B. banana

C. cherry

D. None

ID: 10006

Câu 5: Cho đoạn mã sau:

numbers = [1, 2, 3, 4, 5]

numbers.pop(2)

print(numbers)

Kết quả sau khi chạy đoạn mã là gì?

Đáp án:

A. [1, 2, 4, 5]

B. [1, 2, 3, 4]

C. [2, 3, 4, 5]

D. [1, 3, 4, 5]

ID: 10009

Câu 6: Kết quả của đoạn mã sau là gì?

numbers = [10, 20, 30, 40, 50]

print(numbers[1:4])

Đáp án:

A. [10, 20, 30]

B. [20, 30, 40]

C. [30, 40, 50]

D. [20, 30, 40, 50]

ID: 10013

Câu 7: Các phần tử trong danh sách được ngăn cách nhau bằng ký tự nào?

Đáp án:

A. Dấu chấm phẩy ;

B. Dấu hai chấm :

C. Dấu phẩy ,

D. Dấu gạch ngang -

ID: 10015

Câu 8: Phương thức nào trả về số phần tử có trong danh sách?

Đáp án:

A. count()

B. len()

C. size()

D. sum()

ID: 10016

Câu 9: Để truy cập đến phần tử thứ 3 trong danh sách, cú pháp nào là đúng?

Đáp án:

A. list[1]

B. list[2]

C. list[3]

D. list[-1]

ID: 10018

Câu 10: Phương thức nào sẽ xóa phần tử đầu tiên có giá trị bằng x trong danh sách?

Đáp án:

A. del()

B. remove()

C. pop()

D. clear()

ID: 10019

Câu 11: Cú pháp nào dưới đây là đúng để tạo một danh sách mới bằng cách sử dụng hàm list()?

Đáp án:

A. list([1, 2, 3])

B. list(1, 2, 3)

C. list(1, 2, 3])

D. list([1, 2, 3)

ID: 10021

Câu 12: Phương thức nào trả về danh sách các phần tử đã được sắp xếp theo thứ tự tăng dần?

Đáp án:

A. order()

B. sorted()

C. arrange()

D. align()

ID: 10025

Câu 13: Trong Python, khi truy cập đến phần tử cuối cùng trong danh sách, chỉ số nào thường được sử dụng?

Đáp án:

A. -1

B. 0

C. 1

D. len(list)

ID: 10026

Câu 14: Để truy cập đến phần tử đầu tiên trong danh sách, cú pháp nào là đúng?

Đáp án:

A. list[1]

B. list[0]

C. list[]

D. list[-1]

ID: 10027

Câu 15: Cho danh sách my\_list = [1, 2, 3, 4, 5]. Câu lệnh nào xóa phần tử có giá trị 3 khỏi danh sách?

Đáp án:

A. my\_list.pop(3)

B. my\_list.remove(3)

C. del my\_list[2]

D. my\_list.delete(3)

ID: 10031

Câu 16: Lệnh nào sau đây sẽ trả về danh sách chứa các phần tử từ chỉ số 2 đến chỉ số 4 trong danh sách my\_list = [10, 20, 30, 40, 50, 60]?

Đáp án:

A. my\_list[1:4]

B. my\_list[2:5]

C. my\_list[2:4]

D. my\_list[3:5]

ID: 10033

Câu 17: Nếu danh sách x = [1, 2, 3], lệnh nào sau đây sẽ thêm số 4 vào đầu danh sách?

Đáp án:

A. x.append(4)

B. x.insert(0, 4)

C. x.add(4)

D. x.extend(4)

ID: 10034

Câu 18: Câu lệnh nào sau đây trả về giá trị lớn nhất trong danh sách a = [7, 14, 3, 8]?

Đáp án:

A. max(a)

B. a.max()

C. largest(a)

D. max\_value(a)

ID: 10048

Câu 19: Cho danh sách a = [3, 1, 2]. Lệnh nào trả về danh sách mới với các phần tử đã sắp xếp theo thứ tự tăng dần?

Đáp án:

A. sorted(a)

B. sort(a)

C. a.sort(a)

D. a.reverse()

ID: 10053

Câu 20: Cho danh sách my\_list = [10, 20, 30, 40], lệnh nào sẽ trả về danh sách my\_list đã đảo ngược?

Đáp án:

A. my\_list.reverse(::-1)

B. my\_list[::-1]

C. reversed(my\_list)

D. my\_list[::-2]

ID: 10056

Câu 21: Cho danh sách colors = ['red', 'green', 'blue']. Lệnh nào dưới đây thay đổi giá trị của phần tử thứ hai thành 'yellow'?

Đáp án:

A. colors[2] = 'yellow'

B. colors[1] = 'yellow'

C. colors[0] = 'yellow'

D. colors.replace('green', 'yellow')

ID: 10058

Câu 22: Cho danh sách data = [7, 14, 21], câu lệnh nào sau đây trả về tổng các giá trị trong danh sách?

Đáp án:

A. sum(‘data’)

B. total(data)

C. sum(data)

D. sum([data])

ID: 10060

Câu 23: Nếu danh sách x = [1, 2, 3], lệnh nào sau đây sẽ thêm số 4 vào vị trí 1 trong danh sách?

Đáp án:

A. x.append(1, 4)

B. x.insert(1, 4)

C. x.add(4)

D. x.extend(4)

ID: 10061

Câu 24: Cho danh sách a=[1, 2, 3, 4, 5], lệnh nào sau đây sẽ thực hiện hiển thị phần tử thứ 2 tính từ cuối danh sách?

Đáp án:

A. a[0]

B. a[2]

C. a[-1]

D. a[-2]

ID: 10062

Câu 25: Cho danh sách a=[1, 2, 3, 4, 5], lệnh nào sau đây sẽ thực hiện hiển thị 3 phần tử đầu tiên trong danh sách?

Đáp án:

A. a[0:2]

B. a[3]

C. a[1:3]

D. a[0:3]

ID: 10063

Câu 26: Cho danh sách a=[1, 2, 3, 4, 5], câu lệnh: print(“List[1:0]=”,a[1:0]) cho kết quả là gì?

Đáp án:

A. Hiển thị danh sách rỗng

B. Hiển thị phần tử đầu tiên

C. Hiển thị phần tử thứ 2

D. Hiển thị phần tử thứ 1

ID: 10064

Câu 27: Để khai báo list có 10 phần tử với giá trị mặc định là 0 ta dùng lệnh nào?

Đáp án:

A. lst=[0]\*10

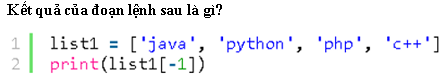
B. lst=[0]

C. lst=[0\*10]

D. lst=(0)\*10

ID: 10066

Câu 28: [hình ảnh]



Đáp án:

A. php

B. c++

C. python

D. java

ID: 10068

Câu 29: Cho đoạn chương trình sau

list=[‘123’, ‘abc’, ‘xyz’, ‘def’]

print(“phan tu co gia tri lon nhat la: ”, max(list))

Kết quả sau khi chạy đoạn chương trình là:

Đáp án:

A. phan tu co gia tri lon nhat la: 123

B. phan tu co gia tri lon nhat la: abc

C. phan tu co gia tri lon nhat la: xyz

D. phan tu co gia tri lon nhat la: def

ID: 10069

Câu 30: list=[‘123’, ‘abc’, ‘xyz’, ‘def’]

print(“phan tu co gia tri nho nhat la: ”,min(list))

Kết quả sau khi chạy đoạn chương trình là:

Đáp án:

A. phan tu co gia tri nho nhat la: 123

B. phan tu co gia tri nho nhat la: abc

C. phan tu co gia tri nho nhat la: xyz

D. phan tu co gia tri nho nhat la: def

# W5

ID: 10430

Câu 1: Dictionary trong Python là gì?

Đáp án:

A. Một cấu trúc dữ liệu lưu trữ các giá trị theo thứ tự.

B. Một cấu trúc dữ liệu lưu trữ các cặp key-value

C. Một cấu trúc dữ liệu chỉ lưu trữ các số nguyên.

D. Một cấu trúc dữ liệu chỉ lưu trữ các chuỗi.

ID: 10431

Câu 2: Phương thức nào dưới đây được sử dụng để thêm một cặp key-value vào Dictionary?

Đáp án:

A. append()

B. insert()

C. add()

D. Truy cập key trực tiếp và gán giá trị cho nó, VD: dictionary[key] = value

ID: 10433

Câu 3: Điều gì xảy ra nếu bạn truy cập một key không tồn tại trong Dictionary?

Đáp án:

A. Trả về giá trị None.

B. Tạo ra một key mới với giá trị mặc định.

C. Gây ra một lỗi KeyError

D. Trả về giá trị rỗng.

ID: 10435

Câu 4: Làm thế nào để thêm một cặp key-value mới vào Dictionary?

Đáp án:

A. Dùng phương thức add().

B. Dùng phương thức append().

C. Gán trực tiếp bằng toán tử = với key và value, VD: dictionary[key] = value

D. Dùng phương thức insert().

ID: 10436

Câu 5: Phương thức nào trả về một danh sách các key trong Dictionary?

Đáp án:

A. values()

B. keys()

C. items()

D. list()

ID: 10438

Câu 6: Phương thức items() trong Dictionary trả về gì?

Đáp án:

A. Một danh sách các key.

B. Một danh sách các value.

C. Một danh sách các tuple chứa cặp key-value

D. Một danh sách các dictionary.

ID: 10439

Câu 7: Tính chất nào của Dictionary là đúng?

Đáp án:

A. Dictionary là một cấu trúc dữ liệu có thứ tự.

B. Dictionary không thể thay đổi sau khi tạo ra.

C. Dictionary cho phép các key bị trùng lặp.

D. Dictionary cho phép thay đổi các giá trị của key.

ID: 10442

Câu 8: Kết quả của biểu thức len({1: "one", 2: "two", 3: "three"}) là gì?

Đáp án:

A. 1

B. 2

C. 3

D. 4

ID: 10448

Câu 9: Sử dụng lệnh nào để tạo một dictionary mới rỗng trong Python?

Đáp án:

A. my\_dict = “ ”

B. my\_dict = { }

C. my\_dict = [ ]

D. my\_dict = ( )

ID: 10453

Câu 10: Làm thế nào để truy cập giá trị của key "name" trong dictionary person = {"name": "Binh", "age": 30}?

Đáp án:

A. A) person["name"]

B. B) person.name

C. C) person.get["name"]

D. Cả A) và C) đều đúng

ID: 10454

Câu 11: Làm thế nào để thêm cặp key-value "city": "Hà Nội" vào dictionary person?

Đáp án:

A. person["city"] = "Hà Nội"

B. person.append({"city": "Hà Nội"})

C. person.add({"city": "Hà Nội"})

D. person.insert("city", "Hà Nội")

ID: 10455

Câu 12: Phương thức nào được sử dụng để xóa tất cả các mục trong một dictionary?

Đáp án:

A. clear()

B. remove()

C. delete()

D. discard()

ID: 10457

Câu 13: Làm thế nào để lấy tất cả các giá trị từ một dictionary?

Đáp án:

A. dictionary.keys()

B. dictionary.items()

C. dictionary.values()

D. dictionary.get()

ID: 10459

Câu 14: Để thêm một sinh viên tên "Dũng" với điểm số là 88 vào students, câu lệnh nào sau đây là đúng?students = { "An": 85, "Bình": 90, "Cường": 78}

Đáp án:

A. students["Dũng"] = 88

B. students.add("Dũng", 88)

C. students.insert({"Dũng": 88})

D. students.append({"Dũng": 88})

ID: 10460

Câu 15: Để cập nhật số lượng ‘Chuối’ thành 120 trong ‘hoaqua’, câu lệnh nào sau đây là đúng?hoaqua = { "Táo": 50, "Chuối": 100, "Cam": 75}

Đáp án:

A. hoaqua["Chuối"] = 120

B. hoaqua.update({"Chuối": 120})

C. hoaqua["Chuối"].append(120)

D. hoaqua["Chuối"] = hoaqua["Chuối"] + 120

ID: 10463

Câu 16: Phương pháp nào dưới đây sẽ xóa phần tử có khóa là 'Hoa' khỏi dictionary students? students = { 'Cong': 85, 'Bong': 92, 'Hoa': 78, 'Vinh': 88}

Đáp án:

A. students.pop('Hoa')

B. students.del['Hoa']

C. students.remove('Hoa')

D. students.clear('Hoa')

ID: 10465

Câu 17: Để truy cập mức lương của Phương, câu lệnh nào sau đây là đúng?employees = { "Linh": 50000, "Phương": 55000, "Hùng": 60000}

Đáp án:

A. employees["Phương"]

B. employees.gets("Phương")

C. employees["Lương của Phương"]

D. employees.get("Lương của Phương")

ID: 10467

Câu 18: Để kiểm tra xem ‘Xoài’ có trong ‘fruit\_prices’ không, câu lệnh nào sau đây là đúng?fruit\_prices = { "Táo": 30000, "Chuối": 15000, "Cam": 25000}

Đáp án:

A. "Xoài" in fruit\_prices

B. fruit\_prices.has\_key("Xoài")

C. fruit\_prices.contains("Xoài")

D. fruit\_prices.exist("Xoài")

ID: 10469

Câu 19: Cho từ điển product\_prices Câu lệnh nào sau đây sẽ in ra tất cả các giá trị trong từ điển?product\_prices = { "Laptop": 1000, "Điện thoại": 600, "Máy tính bảng": 300}

Đáp án:

A. for value in product\_prices.values(): print(value)

B. for key in product\_prices.keys(): print(key)

C. for item in product\_prices.items(): print(item)

D. print(product\_prices)

ID: 10472

Câu 20: Để in ra tất cả các khóa và giá trị của từ điển sau, câu lệnh nào đúng:person\_info = { "Họ": "Nguyễn", "Tên": "Linh", "Tuổi": 30}

Đáp án:

A. for key, value in person\_info.items(): print(key, value)

B. for key in person\_info.keys(): print(key)

C. for value in person\_info.values(): print(value)

D. print(person\_info.keys())

ID: 10474

Câu 21: Bạn có một danh sách các sinh viên và điểm số của họ như sau:students = { 'An': 85, 'Binh': 92, 'Cường': 78, 'Duy': 88, 'Huy': 68}Làm thế nào để kiểm tra xem sinh viên 'Huy' có trong danh sách không?

Đáp án:

A. 'Huy' in students

B. 'Huy' in students.values()

C. students.has\_key('Huy')

D. students.find('Huy')

ID: 10477

Câu 22: Cho dictionary products chứa tên sản phẩm và giá cả:products = { 'Bàn': 500, 'Ghế': 250, 'Giường': 1000, 'Tủ': 750}Làm thế nào để tăng giá của tất cả các sản phẩm lên 10%?

Đáp án:

A. for key in products: products[key] \*= 1.1

B. for value in products.values(): value \*= 1.1

C. products.update(lambda x: x \* 1.1)

D. products = [v \* 1.1 for v in products.values()]

ID: 10480

Câu 23: Giả sử bạn có dictionary sau:grades = {'Anh': 9, 'Toán': 8, 'Lý': 7, 'Hóa': 6}Làm thế nào để tính điểm trung bình?

Đáp án:

A. sum(grades.values()) / len(grades)

B. sum(grades) / len(grades)

C. sum(grades.keys()) / len(grades)

D. sum(grades) / len(grades.keys())

ID: 10488

Câu 24: Cho một dictionary votes lưu số phiếu bầu của mỗi ứng viên:votes = {'Alice': 50, 'Bob': 75, 'Charlie': 25}Làm thế nào để tìm tên ứng viên có số phiếu cao nhất?

Đáp án:

A. max(votes, key=votes.get)

B. max(votes.keys())

C. max(votes)

D. max(votes.values())

ID: 10490

Câu 25: Cho dictionary population:population = {'Hà Nội': 8000000, 'TPHCM': 9000000, 'Đà Nẵng': 1000000}Làm thế nào để cộng thêm 500000 vào dân số của TPHCM?

Đáp án:

A. population['TPHCM'] += 500000

B. population['TPHCM'].update(500000)

C. population.update({'TPHCM': 500000})

D. population('TPHCM') = population('TPHCM') + 500000

ID: 10492

Câu 26: Cho dictionary employees lưu tên nhân viên và chức vụ:employees = {'Nam': 'Giám đốc', 'Linh': 'Quản lý', 'Hùng': 'Nhân viên'}Làm thế nào để thêm một nhân viên mới tên là 'Tú' với chức vụ 'Nhân viên'?

Đáp án:

A. employees['Tú'] = 'Nhân viên'

B. employees.append({'Tú': 'Nhân viên'})

C. employees.update({'Tú', 'Nhân viên'})

D. employees['Tú'] = employees.get('Nhân viên')

ID: 10495

Câu 27: Cho dictionary library:library = {'Sách A': 3, 'Sách B': 5, 'Sách C': 2}Làm thế nào để giảm số lượng sách 'Sách B' đi 2?

Đáp án:

A. library['Sách B'] - = 2

B. library.update({'Sách B': library['Sách B'] - 2})

C. library.pop('Sách B')

D. library['Sách B'] = 2

ID: 10497

Câu 28: Cho dictionary expenses:expenses = {'Điện': 100, 'Nước': 50, 'Internet': 70}Làm thế nào để cộng tổng chi phí?

Đáp án:

A. sum(expenses.values())

B. sum(expenses)

C. sum(expenses.keys())

D. sum([key + value for key, value in expenses.items()])

ID: 10498

Câu 29: Giả sử bạn có dictionary fruits lưu tên trái cây và số lượng:fruits = {'Táo': 10, 'Chuối': 20, 'Cam': 30}Làm thế nào để nhân đôi số lượng của mỗi loại trái cây?

Đáp án:

A. for key in fruits: fruits[key] \*= 2

B. for key in fruits.keys(): key \*= 2

C. for value in fruits.values(): value \*= 2

D. fruits.update(lambda x: x \* 2)

ID: 10500

Câu 30: Cho dictionary inventory:inventory = {'Bút': 100, 'Vở': 200, 'Thước': 150}Làm thế nào để xóa tất cả các phần tử trong inventory?

Đáp án:

A. inventory.clear()

B. inventory = None

C. del inventory

D. inventory.remove()

# W6

ID: 10972

Câu 1: Kiểu dữ liệu Tuple trong Python có đặc điểm gì?

Đáp án:

A. Có thể thay đổi kích thước sau khi tạo

B. Không thể thay đổi sau khi tạo

C. Chỉ chứa các phần tử kiểu số

D. Chỉ chứa các phần tử có cùng kiểu dữ liệu

ID: 10980

Câu 2: Lệnh nào dưới đây sẽ tạo ra một Tuple chứa ba phần tử?

Đáp án:

A. t = [1, 2, 3]

B. t = {1, 2, 3}

C. t = (1, 2, 3)

D. t = 1, 2, 3

ID: 10995

Câu 3: Lệnh nào dưới đây sẽ tạo ra một Tuple có một phần tử duy nhất là số 5?

Đáp án:

A. t = (5,)

B. t = (5)

C. t = [5]

D. t = {5}

ID: 11007

Câu 4: Tuple khác với List như thế nào về cú pháp?

Đáp án:

A. Tuple sử dụng ngoặc vuông [ ] còn List sử dụng ngoặc nhọn { }

B. Tuple sử dụng ngoặc nhọn { } còn List sử dụng ngoặc vuông [ ]

C. Tuple sử dụng ngoặc tròn ( ) còn List sử dụng ngoặc vuông [ ]

D. Tuple sử dụng ngoặc vuông [ ] còn List sử dụng ngoặc tròn ( )

ID: 11012

Câu 5: Nếu bạn cố gắng thay đổi một phần tử trong Tuple, điều gì sẽ xảy ra?

Đáp án:

A. Phần tử sẽ được thay đổi thành giá trị mới.

B. Một lỗi sẽ xảy ra vì Tuple là không thể thay đổi

C. Phần tử sẽ bị xóa khỏi Tuple.

D. Giá trị sẽ được thêm vào cuối Tuple.

ID: 11019

Câu 6: Lệnh nào sau đây sẽ tạo ra một Tuple rỗng?

Đáp án:

A. t = ( )

B. t = [ ]

C. t = {}

D. t = None

ID: 11024

Câu 7: Làm thế nào để tạo một Tuple từ một List?

Đáp án:

A. tuple(list)

B. list(tuple)

C. list.to\_tuple()

D. tuple.from\_list(list)

ID: 11029

Câu 8: Giả sử bạn có một Tuple numbers như sau:numbers = (10, 20, 30)Bạn muốn thêm phần tử có giá trị 40 vào cuối Tuple này. Bạn nên thực hiện các bước nào?

Đáp án:

A. Tạo một Tuple mới bằng cách kết hợp Tuple cũ và Tuple chứa phần tử mới: new\_numbers = numbers + (40,)

B. Sử dụng numbers.append(40)

C. Sử dụng numbers.extend((40,))

D. Sử dụng numbers.insert(-1, 40)

ID: 11036

Câu 9: Nếu bạn có Tuple t= (10, 20, 30) và bạn muốn truy cập phần tử thứ hai, lệnh nào dưới đây là đúng?

Đáp án:

A. t[2]

B. t.get(2)

C. t[1]

D. t(2)

ID: 11043

Câu 10: Khi bạn sử dụng Tuple để lưu trữ dữ liệu, tại sao Tuple không thể dùng phương thức remove như List?

Đáp án:

A. Vì Tuple không hỗ trợ các phương thức thay đổi dữ liệu, chỉ hỗ trợ các phương thức truy cập dữ liệu.

B. Vì Tuple không có phương thức remove.

C. Vì Tuple không thể chứa các kiểu dữ liệu khác nhau.

D. Vì Tuple chỉ chứa các giá trị số nguyên.

ID: 11048

Câu 11: Lệnh nào dưới đây dùng để kiểm tra xem một giá trị có tồn tại trong Tuple không?

Đáp án:

A. value in tuple

B. tuple.contains(value)

C. tuple.has(value)

D. tuple.exists(value)

ID: 11056

Câu 12: Tại sao việc sử dụng Tuple là lựa chọn tốt khi bạn cần lưu trữ một tập hợp các giá trị không thay đổi?

Đáp án:

A. Tuple có thể thay đổi kích thước dễ dàng sau khi tạo.

B. Tuple hỗ trợ thêm và xóa phần tử dễ dàng.

C. Tuple là không thể thay đổi, đảm bảo dữ liệu không bị thay đổi sau khi tạo, giúp bảo vệ tính toàn vẹn của dữ liệu.

D. Tuple chỉ chứa các kiểu dữ liệu số, làm cho việc lưu trữ dữ liệu không thay đổi dễ dàng hơn.

ID: 11060

Câu 13: Bạn có Tuple data = (5, 10, 15) và bạn muốn tạo một Tuple mới chứa phần tử đầu tiên và cuối cùng. Lệnh nào dưới đây là đúng?

Đáp án:

A. new\_data = (data[0], data[-1])

B. new\_data = data[::2]

C. new\_data = data[1:]

D. new\_data = (data[1], data[2])

ID: 11070

Câu 14: Khi bạn muốn hợp nhất hai Tuples tuple1 = (1, 2) và tuple2 = (3, 4) thành một Tuple duy nhất, cách nào là đúng?

Đáp án:

A. merged = tuple1 + tuple2

B. merged = tuple1.merge(tuple2)

C. merged = tuple1.append(tuple2)

D. merged = tuple1.extend(tuple2)

ID: 11072

Câu 15: Nếu bạn có Tuple dimensions = (5, 10) và bạn cần tính diện tích bằng cách nhân hai phần tử, lệnh nào dưới đây là đúng?

Đáp án:

A. area = dimensions[0] \* dimensions[1]

B. area = dimensions.product()

C. area = dimensions.sum()

D. area = dimensions.get(0) \* dimensions.get(1)

ID: 11078

Câu 16: Bạn có Tuple data = (1, 2, 3, 4) và bạn muốn lấy phần tử ở vị trí thứ ba, lệnh nào dưới đây là đúng?

Đáp án:

A. element = data[2]

B. element = data.get(3)

C. element = data[3]

D. element = data(2)

ID: 11081

Câu 17: Bạn có Tuple data = (3, 6, 9, 12, 15) và bạn muốn tính tổng các số chẵn trong Tuple. Lệnh nào dưới đây là đúng?

Đáp án:

A. total = sum(x for x in data if x % 2 == 0)

B. total = data.sum() if x % 2 == 0

C. total = data.filter(lambda x: x % 2 == 0).sum()

D. total = reduce(lambda x, y: x + y if y % 2 == 0 else x, data)

ID: 11086

Câu 18: Cho Tuple scores = (85, 92, 78, 90) và bạn muốn tìm điểm cao nhất và điểm thấp nhất. Lệnh nào dưới đây là đúng?

Đáp án:

A. max\_score = max(scores), min\_score = min(scores)

B. max\_score = scores.max(), min\_score = scores.min()

C. max\_score = scores[0], min\_score = scores[-1]

D. max\_score, min\_score = scores.sort()

ID: 11132

Câu 19: Giả sử bạn có Tuple coordinates = (2, 4, 6) và bạn muốn tính tổng của tất cả các giá trị trong Tuple, lệnh nào dưới đây là đúng?

Đáp án:

A. total = sum(coordinates)

B. total = coordinates.sum()

C. total = reduce(lambda x, y: x + y, coordinates)

D. total = coordinates.aggregate(sum)

ID: 11142

Câu 20: Nếu bạn có Tuple elements = (3, 6, 9) và bạn cần nhân mỗi phần tử với 2 để tạo một Tuple mới, lệnh nào dưới đây là đúng?

Đáp án:

A. new\_elements = tuple(x \* 2 for x in elements)

B. new\_elements = elements.map(lambda x: x \* 2)

C. new\_elements = list(map(lambda x: x \* 2, elements))

D. new\_elements = elements \* 2

ID: 11149

Câu 21: Nếu bạn có Tuple info = ('John', 30, 'New York'), để tạo Tuple mới có giá trị ('John', 31, 'New York'), cách nào dưới đây là đúng?

Đáp án:

A. new\_info = (info[0], 31, info[2])

B. new\_info = info.update(1, 31)

C. new\_info = info.replace(30, 31)

D. new\_info = info[0:1] + (31,) + info[2:]

ID: 11158

Câu 22: Bạn có Tuple data = (5, 10, 15, 20) và bạn muốn lấy các phần tử có giá trị lớn hơn 10, lệnh nào dưới đây là đúng?

Đáp án:

A. filtered\_data = tuple(x for x in data if x>10)

B. filtered\_data = filter(lambda x: x>10, data)

C. filtered\_data = data.filter(lambda x: x>10)

D. filtered\_data = data[2:]

ID: 11165

Câu 23: Để tìm chỉ số đầu tiên của phần tử 'banana' trong Tuple fruits = ('apple', 'banana', 'apple', 'cherry'), câu lệnh nào đúng?

Đáp án:

A. index = fruits.index('banana')

B. index = fruits.find('banana')

C. index = fruits.search('banana')

D. index = fruits.position('banana')

ID: 11169

Câu 24: Giả sử bạn có Tuple data = (1, 2, 3) và bạn cần sao chép Tuple này 3 lần để tạo thành một Tuple mới. Lệnh nào dưới đây là đúng?

Đáp án:

A. new\_data = data \* 3

B. new\_data = data.copy(3)

C. new\_data = data.repeat(3)

D. new\_data = data.concat(data, data)

ID: 11173

Câu 25: Bạn có Tuple data = (1, 2, 3, 4, 5) và bạn muốn lấy phần tử ở vị trí thứ hai đến cuối cùng. Lệnh nào dưới đây là đúng?

Đáp án:

A. sub\_data = data[1:]

B. sub\_data = data[:2]

C. sub\_data = data[2:]

D. sub\_data = data[2:4]

ID: 11177

Câu 26: Cho Tuple data = (1, 2, 3, 4), làm thế nào để chuyển đổi nó thành một List?

Đáp án:

A. data\_list = list(data)

B. data\_list = data.to\_list()

C. data\_list = convert\_to\_list(data)

D. data\_list = list.convert(data)

ID: 11181

Câu 27: Bạn có Tuple numbers = (4, 7, 2, 9, 5) và bạn muốn sắp xếp các số theo thứ tự giảm dần. Lệnh nào dưới đây là đúng?

Đáp án:

A. sorted\_numbers = numbers.reverse()

B. sorted\_numbers = numbers.sort(reverse=True)

C. sorted\_numbers = numbers.order(reverse=True)

D. sorted\_numbers = tuple(sorted(numbers, reverse=True))

ID: 11187

Câu 28: Giả sử bạn có Tuple data = (5, 10, 15, 20) và bạn cần tạo một Tuple mới với các phần tử chỉ chia hết cho 5. Câu lệnh nào dưới đây là đúng?

Đáp án:

A. filtered\_data = tuple(x for x in data if x % 5 == 0)

B. filtered\_data = data.filter(lambda x: x % 5 == 0)

C. filtered\_data = data.select(lambda x: x % 5 == 0)

D. filtered\_data = tuple(x for x in data if x 5)

ID: 11199

Câu 29: Nếu bạn có Tuple data = (1, 2, 3, 4, 5) và bạn muốn lấy phần tử thứ ba và thứ tư, lệnh nào dưới đây là đúng?

Đáp án:

A. sub\_data = data[2:4]

B. sub\_data = data[3:5]

C. sub\_data = data[2:5]

D. sub\_data = data[1:3]

ID: 11203

Câu 30: Bạn có Tuple data = (7, 14, 21, 28) và bạn muốn tìm các số chia hết cho 7. Lệnh nào dưới đây là đúng?

Đáp án:

A. filtered\_data = tuple(x for x in data if x % 7 == 0)

B. filtered\_data = data.filter(lambda x: x % 7 == 0)

C. filtered\_data = data.select(lambda x: x % 7 == 0)

D. filtered\_data = tuple(x for x in data if x>7)

# W7

ID: 11274

Câu 1: Câu lệnh nào dưới đây dùng để tạo một Set rỗng trong Python?

Đáp án:

A. empty\_set = set()

B. empty\_set = {}

C. empty\_set = []

D. empty\_set = ()

ID: 11276

Câu 2: Câu lệnh nào dưới đây dùng để thêm phần tử vào Set?

Đáp án:

A. my\_set.add(element)

B. my\_set.append(element)

C. my\_set.insert(element)

D. my\_set.extend(element)

ID: 11331

Câu 3: Khi bạn sử dụng phương thức add() trên một Set, điều gì sẽ xảy ra nếu bạn cố gắng thêm một phần tử đã tồn tại?

Đáp án:

A. Phần tử sẽ không được thêm và Set không thay đổi.

B. Phần tử sẽ được thêm vào nhiều lần.

C. Phần tử sẽ thay thế phần tử cũ với giá trị mới.

D. Một lỗi sẽ được phát sinh.

ID: 11337

Câu 4: Điều gì xảy ra khi bạn sử dụng phương thức discard() để xóa một phần tử không có trong Set?

Đáp án:

A. Phương thức sẽ không làm gì và Set không thay đổi.

B. Phần tử sẽ được xóa nếu tồn tại.

C. Một lỗi sẽ được phát sinh.

D. Phương thức sẽ thêm phần tử vào Set.

ID: 11392

Câu 5: Khi bạn gọi phương thức union() trên hai Set, kết quả là gì?

Đáp án:

A. Tập hợp các phần tử xuất hiện trong ít nhất một trong hai Set.

B. Tập hợp các phần tử xuất hiện trong cả hai Set.

C. Tập hợp các phần tử chỉ xuất hiện trong Set đầu tiên.

D. Tập hợp các phần tử chỉ xuất hiện trong Set thứ hai.

ID: 11393

Câu 6: Câu lệnh nào dưới đây sẽ trả về tập hợp các phần tử xuất hiện trong Set đầu tiên nhưng không xuất hiện trong Set thứ hai?

Đáp án:

A. set1.difference(set2)

B. set1.intersection(set2)

C. set1.union(set2)

D. set1.symmetric\_difference(set2)

ID: 11394

Câu 7: Khi bạn gọi phương thức pop() trên một Set, phần tử nào sẽ được xóa?

Đáp án:

A. Phần tử đầu tiên theo thứ tự của Set.

B. Một phần tử ngẫu nhiên trong Set.

C. Phần tử cuối cùng theo thứ tự của Set.

D. Phần tử đầu tiên được thêm vào Set.

ID: 11395

Câu 8: Phương thức nào dưới đây sẽ trả về một Set chứa các phần tử chỉ xuất hiện trong một trong hai Set nhưng không xuất hiện trong cả hai?

Đáp án:

A. set1.symmetric\_difference(set2)

B. set1.union(set2)

C. set1.intersection(set2)

D. set1.difference(set2)

ID: 11396

Câu 9: Bạn có hai Set A = {1, 2, 3} và B = {2, 3, 4}. Lệnh nào dưới đây sẽ trả về Set chứa tất cả các phần tử từ cả hai Set mà không có phần tử nào lặp lại?

Đáp án:

A. A.union(B)

B. A.intersection(B)

C. A.difference(B)

D. A.symmetric\_difference(B)

ID: 11397

Câu 10: Nếu bạn có Set data = {10, 20, 30} và bạn muốn loại bỏ số 20 khỏi Set, lệnh nào dưới đây là đúng?

Đáp án:

A. data.remove(20)

B. data.discard(20)

C. data.pop(20)

D. data.delete(20)

ID: 11398

Câu 11: Bạn có Set set1 = {1, 2, 3} và set2 = {2, 3, 4}. Bạn cần tìm các phần tử chỉ có trong set1 nhưng không có trong set2. Lệnh nào dưới đây là đúng?

Đáp án:

A. result = set1 - set2

B. set1.intersection(set2)

C. set1.union(set2)

D. set1.symmetric\_difference(set2)

ID: 11404

Câu 12: Khi bạn gọi phương thức symmetric\_difference() trên hai Set, kết quả sẽ là gì?

Đáp án:

A. Tập hợp các phần tử chỉ xuất hiện trong một trong hai Set nhưng không xuất hiện trong cả hai.

B. Tập hợp các phần tử xuất hiện trong cả hai Set.

C. Tập hợp các phần tử chỉ xuất hiện trong Set đầu tiên.

D. Tập hợp các phần tử chỉ xuất hiện trong Set thứ hai.

ID: 11405

Câu 13: Giả sử bạn có Set numbers = {1, 2, 3, 4, 5} và bạn muốn tạo một Set mới chỉ chứa các số chẵn. Lệnh nào dưới đây là đúng?

Đáp án:

A. even\_numbers = {x for x in numbers if x % 2 == 0}

B. even\_numbers = numbers.filter(lambda x: x % 2 == 0)

C. even\_numbers = numbers.select(lambda x: x % 2 == 0)

D. even\_numbers = {x if x % 2 == 0}

ID: 11406

Câu 14: Giả sử bạn có một tập hợp A = {1, 2, 3}. Điều gì xảy ra nếu bạn thêm phần tử 2 vào tập hợp A bằng cách sử dụng lệnh A.add(2)?

Đáp án:

A. Tập hợp A sẽ trở thành {1, 2, 3, 2}.

B. Tập hợp A sẽ trở thành {1, 2}.

C. Tập hợp A sẽ không thay đổi.

D. Tập hợp A sẽ trở thành {1, 2, 3, 4}.

ID: 11407

Câu 15: Kết quả của chương trình sau là gì?set1 = set([ 4, 5, (6, 7)]) set1.update([10, 11])print(set1)

Đáp án:

A. {4, 5, (6, 7), 10, 11}

B. {4, 5, 6, 7, 10, 11}

C. {4, 5, 10, 11}

D. None

ID: 11409

Câu 16: Kết quả của chương trình sau là gì?sets = {1, 2, 3, 4, 4} print(sets)

Đáp án:

A. {1, 2, 3, 4}

B. {1, 2, 3}

C. {1, 2, 3, 4, 4}

D. Error

ID: 11410

Câu 17: Giả sử bạn có Set data = {1, 2, 3, 4, 5} và bạn muốn xóa tất cả các phần tử không nằm trong tập hợp {2, 4, 6}. Viết lệnh nào dưới đây là đúng?

Đáp án:

A. data.intersection\_update({2, 4, 6})

B. data.difference\_update({2, 4, 6})

C. data.remove({2, 4, 6})

D. data.discard({2, 4, 6})

ID: 11411

Câu 18: Bạn có Set numbers = {1, 2, 3, 4, 5} và bạn muốn tạo một Set mới chỉ chứa các số lớn hơn 3. Viết lệnh nào dưới đây là đúng?

Đáp án:

A. filtered\_numbers = {x for x in numbers if x>3}

B. filtered\_numbers = numbers.filter(lambda x: x 3)

C. filtered\_numbers = {x 3 for x in numbers}

D. filtered\_numbers = numbers.remove(x for x in numbers if x = 3)

ID: 11412

Câu 19: Kết quả của chương trình sau là gì?sets = {3, 4, 5} sets.update([1, 2, 3]) print(sets)

Đáp án:

A. {1, 2, 3, 4, 5}

B. {3, 4, 5, 1, 2, 3}

C. {1, 2, 3, 3, 4, 5}

D. Error

ID: 11413

Câu 20: Giả sử bạn có Set data = {10, 20, 30, 40} và bạn muốn sao chép Set này sang một Set mới. Viết lệnh nào dưới đây là đúng?

Đáp án:

A. new\_data = data.copy()

B. new\_data = data.clone()

C. new\_data = data.duplicate()

D. new\_data = data.replicate()

ID: 11414

Câu 21: Kết quả của chương trình sau là gì?set1 = {1, 2, 3} set2 = set1.copy() set2.add(4) print(set1)

Đáp án:

A. {1, 2, 3}

B. {1, 2, 3, 4}

C. Invalid Syntax

D. Error

ID: 11415

Câu 22: Nếu bạn có Set data = {5, 10, 15} và bạn muốn kiểm tra xem số 10 có tồn tại trong Set hay không, viết lệnh nào dưới đây là đúng?

Đáp án:

A. exists = 10 in data

B. exists = data.has(10)

C. exists = data.contains(10)

D. exists = data.exists(10)

ID: 11416

Câu 23: Giả sử bạn có Set numbers = {1, 2, 3, 4, 5} và bạn muốn tìm phần tử duy nhất trong numbers có thể chia hết cho cả 2 và 3. Viết lệnh nào dưới đây là đúng?

Đáp án:

A. result = {x for x in numbers if x % 2 == 0 and x % 3 == 0}

B. result = {x for x in numbers if x % 2 == 0 or x % 3 == 0}

C. result = numbers.filter(lambda x: x % 2 == 0 and x % 3 == 0)

D. result = numbers.intersection({6})

ID: 11417

Câu 24: Nếu bạn có Set data = {10, 12, 25, 50} và bạn muốn loại bỏ tất cả các phần tử có giá trị 25, viết lệnh nào dưới đây là đúng?

Đáp án:

A. data = {x for x in data if x> 25 or x<25}

B. data.remove(x for x in data if x = 25 or x =25)

C. data.discard(x for x in data if x = 25)

D. data = data.difference({x for x in data if x = 25})

ID: 11418

Câu 25: Nếu bạn có Set data = {10, 12, 30, 50} và bạn muốn loại bỏ tất cả các phần tử nhỏ hơn 25, viết lệnh nào dưới đây là đúng?

Đáp án:

A. data = {x for x in data if x >= 25}

B. data.remove(x for x in data if x 25)

C. data.discard(x for x in data if x 25)

D. data = data.difference({x for x in data if x 25})

ID: 11419

Câu 26: Nếu bạn có Set data = {10, 20, 30, 40} và bạn muốn loại bỏ tất cả các phần tử lớn hơn 25, viết lệnh nào dưới đây là đúng?

Đáp án:

A. data = {x for x in data if x <= 25}

B. data.remove(x for x in data if x 25)

C. data.discard(x for x in data if x 25)

D. data = data.difference({x for x in data if x 25})

ID: 11421

Câu 27: Kết quả của chương trình sau là gì?set1 = {1, 2, 3} set2 = set1.add(4) print(set2)

Đáp án:

A. None

B. {1, 2, 3, 4}

C. {1, 2, 3}

D. Invalid Syntax

ID: 11422

Câu 28: Giả sử bạn có Set students = {'Alice', 'Bob', 'Charlie'} và bạn muốn thêm 'David' vào Set này. Viết lệnh nào dưới đây là đúng?

Đáp án:

A. students.add('David')

B. students.append('David')

C. students.insert('David')

D. students.extend('David')

ID: 11423

Câu 29: Bạn có hai Set A = {1, 2, 3, 4, 5} và B = {4, 5, 6, 7, 8}. Viết lệnh để tạo một Set mới chứa các phần tử xuất hiện trong cả hai Set.

Đáp án:

A. intersection\_set = A B

B. intersection\_set = A.union(B)

C. intersection\_set = A.difference(B)

D. intersection\_set = A.symmetric\_difference(B)

ID: 11432

Câu 30: Bạn có hai Set A = {1, 2, 3, 4, 5} và B = {4, 5, 6, 7, 8}. Viết lệnh để tạo một Set mới chứa các phần tử chỉ xuất hiện trong Set A.

Đáp án:

A. difference\_set = A - B

B. difference\_set = A B

C. difference\_set = A.union(B)

D. difference\_set = A.symmetric\_difference(B)