Python Examples

Coding Exercises
Coding Interview

Ray Yao

Coding Exercises

Coding Interview

Python

Examples

Coding Exercises

Coding Interview

Ray Yao

About the Author : Ray Yao's Team

Certified PHP engineer by Zend, USA

Certified JAVA programmer by Sun, USA

Certified SCWCD developer by Oracle, USA

Certified A+ professional by CompTIA, USA

Certified ASP **.** NET expert by Microsoft, USA

Certified MCP professional by Microsoft, USA

Certified TECHNOLOGY specialist by Microsoft, USA

Certified NETWORK+ professional by CompTIA, USA

www.amazon.com/author/ray-yao

Recommended Books by Ray Yao's Team

Advanced C++ in 8 Hours

Advanced Java in 8 Hours

AngularJs in 8 Hours

Asp . net Programming

Awk in 8 Hours

BootStrap in 8 Hours

C# Examples & Interview

C# Programming

C++ Examples & Interview

C++ Programming

Dart in 8 Hours

Django in 8 Hours

Erlang in 8 Hours

Go in 8 Hours

Html Css Examples & Interview

Html Css Programming

Java Examples & Interview

Java Programming

JavaScript Examples & Interview

JavaScript Programming

JQuery Examples & Interview

JQuery Programming

Jsp Servlets Programming

Kotlin in 8 Hours

Linux Command Line

Linux Examples & Interview

Lua in 8 Hours

Matlab in 8 Hours

MySql in 8 Hours

Node . Js in 8 Hours

Numpy in 8 Hours

Perl in 8 Hours

Php Examples & Interview

Php MySql Programming

PowerShell in 8 Hours

Python Examples & Interview

Python Programming

R Programming

React . Js in 8 Hours

Ruby Programming

Rust in 8 Hours

Scala in 8 Hours

Shell Scripting in 8 Hours

Swift in 8 Hours

Tcl in 8 Hours

TypeScript in 8 Hours

Visual Basic Examples & Interview

Visual Basic Programming

Vue . Js in 8 Hours

Xml Json in 8 Hours

Preface

30 Python Useful Examples

This book includes 30 Python examples for beginners .

Through these useful examples, you can study Python programming skills in depth, master skillfully the Python coding knowledge, and became an expert of Python programming .

100 Question & Answers

This book can help you:

Pass the college final exams

Pass the job interview exams

Pass the engineer certification exams

100 Answers for Download

This book includes 100 answers; for your convenience, you can download and print it out to check the questions .

Please download the 100 answers of this book:

https://forms . aweber . com/form/00/638806700 . htm

Table of Contents

Python 30 Examples

Python 30 Examples

01. Get a square root

Source Code:

```
num = float(input('Please input a numbe r :    '))
print (num)
sqrt = num ** 0.5
print('\n The square root of  %0.1f is:  %0.4f'%(num
, sqrt))
```

(Assume that we input a number 81)

Output:

Please input a numbe r :   81.0

The square root of 81.0 is: 9.0000

02. Calculate the triangle area

Source Code:

```
print('Please input three numbers: ')    # three sides of triangle
x = float(input(''))
y = float(input(''))
z = float(input(''))
print('%d ' %x)
print('%d ' %y)
print('%d ' %z)
# calculate the half girth
girth = (x + y + z) / 2
# calculate the triangle area
area = (girth*(girth-x)*(girth-y)*(girth-z)) ** 0.5
print('The triangle area is: %0.3f' %area)
```

Output:

Please input three numbers:
3
4
5
The triangle area is: 6.000

03. Calculate the circle area

Source Code:

```
print('Please input a radius: ')
radius = int(input(''))
print(radius)
def area(r):
    PI = 3.142
    return PI * (r*r)
print("The area of circle is: %.2f" %area(radius))
```

(Assume that we input a radius: 10)

Output:

Please input a radius:

10

The area of circle is: 314.20

04. Change a centigrade to fahrenheit

Source Code:

```
# -*- coding: UTF-8 -*-
print('Please enter a Centigrade:')
C = float(input(''))
print(C)
F = (C * 1.8) + 32
print('Centigrade %0.1f equals Fahrenheit %0.1f '%
(C, F))
```

(Assume that we input 10)

Output:

Please enter a Centigrade:

10.0

Centigrade 10.0 equals Fahrenheit 50.0

05. Confirm an even or odd number.

Source Code:

```
number = int(input("Please input a number: "))
print(number)
if (number % 2) == 0:
   print("{0} is an even number.".format(number))
else:
   print("{0} is an odd number.".format(number))
```

(Assume that we input 10)

Output:

Please input a number: 10

10 is an even number.

06. Decimal & other number conversion

Source Code:

```
num = int(input("Please input a numbe r ：  "))
print(num)
print("Decimal number i s ：  ", num)
print("Binary number i s ：  ", bin(num))
print("Octonary number i s ：  ", oct(num))
print("Hexadecimal number i s ：  ", hex(num))
```

(Assume that we input 10)

Output:

Please input a numbe r ： 10

Decimal number i s ： 10

Binary number i s ： 0b1010

Octonary number i s ： 0o12

Hexadecimal number i s ： 0xa

07. ASCII code & character conversion

Source Code:

```
A = int(input("Please input an ascii cod e :    "))
print(A)
C = input("Please input a characte r :    ")
print(C)
print("The corresponding character of",A,"is:
",chr(A))
print("The corresponding ascii code of",C,"is:
",ord(C))
```

(Assume that we input an ASCII code 70 and a character F)

Output:

Please input an ascii cod e : 70

Please input a characte r : F

The corresponding character of 70 is: F

The corresponding ascii code of F is: 70

08. Python calendar

Source Code:

```
import calendar
print("Please input the year with four numbers: ")
yyyy = int(input( "" ))
print(yyyy)
print("Please input the month with two numbers: ")
mm = int(input( "" ))
print(mm)
print(calendar.month(yyyy,mm))
```

(Assume that we input 2021 and 03)

Output:

Please input the year with four numbers:
2021
Please input the month with two numbers:
03
March 2021
Mo Tu We Th Fr Sa Su
1 2 3 4 5 6 7
8 9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31

09. Get yesterday

Source Code:

```
import datetime
def myFun():
    thisDay=datetime.date.today()
    aDay=datetime.timedelta(days=1)
    lastDay=thisDay-aDay
    return lastDay
print('Yesterday was: ')
print(myFun())
```

Output:

Yesterday was:

2021-03-25

10. Search url in an article

Source Code:

```
import re
def myFun(str):
    url = re.findall('https?://(?:[-\w.]|(?:%[\da-fA-F]{2}))+', str)
    return url
str = ' The website of Yahoo i s ： http://www.yahoo.com. The website of Amazon is: https://www.amazon.com. Welcome to these websites! '
print("The original article is: ", str, "\n")
print("The result of searching URL: \n", myFun(str))
```

Output:

The original article is: The website of Yahoo i s ： http://www.yahoo.com. The website of Amazon is: https://www.amazon.com. Welcome to these websites!

The result of searching URL:
['http://www.yahoo.com', 'https://www.amazon.com']

11. Get the greatest common divisor

Source Code

```
def myFun(a, b):
  if a > b:
     smaller = b
  else:
     smaller = a
  for v in range(1,smaller + 1):
     if((a % v == 0) and (b % v == 0)):
        myFun = v
  return myFun
n1 = int(input("Please input the first number: "))
print(n1)
n2 = int(input("Please input the second number: "))
print(n2)
print("The greatest common divisor between",
n1,"and", n2, "is" , myFun(n1, n2))
```

(Assume that we input 6 and 8)

Output:

Please input the first number: 6
Please input the second number: 8
The greatest common divisor between 6 and 8 is 2

12. Get the least common multiple

Source Code:

```
def myFun(a, b):
   if a > b:
      greater = a
   else:
      greater = b
   while(True):
      if((greater % a == 0) and (greater % b == 0)):
         myFun = greater
         break
      greater += 1
   return myFun
n1 = int(input("Please input the first number: "))
print(n1)
n2 = int(input("Please input the second number: "))
print(n2)
print("The least common multiple
between",n1,"and", n2, "is:" , myFun(n1, n2))
```

(Assume that we input 6 and 8)

Output:

Please input the first number: 6
Please input the second number: 8
The least common multiple between 6 and 8 is: 24

13. Miles converts to kilometers

Source Code:

```
miles = float(input("Please input a number in miles: "))
print(miles)
convertor = 0.621371
kilometers = miles / convertor
print('%0.3f miles is equal to %0.3f kilometers' % (miles, kilometers))
```

(Assume that we input 10)

Output:

Please input a number in miles: 10.0

10.000 miles is equal to 16.093 kilometers

14. Multiplication Table

Source Code:

```
for x in range(1, 10):
    print()
    for y in range(1, x+1):
        print ("%d*%d=%d" % (x, y, x*y), end=" " )
```

Output:

```
1*1=1
2*1=2 2*2=4
3*1=3 3*2=6 3*3=9
4*1=4 4*2=8 4*3=12 4*4=16
5*1=5 5*2=10 5*3=15 5*4=20 5*5=25
6*1=6 6*2=12 6*3=18 6*4=24 6*5=30 6*6=36
7*1=7 7*2=14 7*3=21 7*4=28 7*5=35 7*6=42 7*7=49
8*1=8 8*2=16 8*3=24 8*4=32 8*5=40 8*6=48 8*7=56 8*8=64
9*1=9 9*2=18 9*3=27 9*4=36 9*5=45 9*6=54 9*7=63 9*8=72 9*9=81
```

15. Mark and grade

Source Code:

```
mark = float(input('Please input an exam mark: '))
print(mark)
if mark >= 90:
    grade = 'A'
elif mark >= 80:
    grade = 'B'
elif mark >= 60:
    grade = 'C'
else:
    grade = 'D'
print ('%d means %s' % ( mark, grade ))
```

Output:

Please input an exam mark: 100.0

100 means A

16. Show the person who is oldest

Source Code:

```
if __name__ == '__main__':
    person = {"Ann":16,"Lee":28,"Tom":60,"Nel":32}
    oldest = 'Ann'
    for key in person.keys():
        if person[oldest] < person[key]:
            oldest = key
print('The oldest guy is %s, age is %d.' %
(oldest,person[oldest]))
```

Output:

The oldest guy is Tom, age is 60.

17. Print a diamond pattern

Source Code

```
from sys import stdout
for x in range(4):
    for y in range(2 - x + 1):
        stdout.write(' ')
    for num in range(2 * x + 1):
        stdout.write('*')
    print('')
for x in range(3):
    for y in range(x + 1):
        stdout.write(' ')
    for num in range(4 - 2 * x + 1):
        stdout.write('*')
    print('')
```

Output:

```
   *
  ***
 *****
*******
 *****
  ***
   *
```

18. Get local time & Greenwich Time

Source Code:

```
if __name__ == '__main__':
    import time
    print(time.ctime(time.time()))
    print(time.asctime(time.localtime(time.time())))
    print(time.asctime(time.gmtime(time.time())))
```

Output:

Wed Mar 24 21:21:40 2021

Wed Mar 24 21:21:40 2021

Wed Mar 24 21:18:40 2021

19. Count the sum from 1 to 100

Source Code:

```
sum = 0
for num in range(1,101):
    sum += num
print ('The sum is %d' %(sum))
```

Output:

```
The sum is 5050
```

20. Count letters, spaces and characters

Source Code:

```
import string
str = "Shell Scripting in 8 Hours!"
letters = 0
spaces = 0
number = 0
symbol = 0
i=0
while i < len(str):
    c = str[i]
    i += 1
    if c.isalpha():
        letters += 1
    elif c.isspace():
        spaces += 1
    elif c.isdigit():
        number += 1
    else:
        symbol += 1
print('chars = %d, spaces = %d, number = %d,
symbol = %d' % (letters, spaces, number, symbol))
```

Output:

```
chars = 21, spaces = 4, number = 1, symbol = 1
```

21. Output the values in reverse order

Source Code:

```
mylist = ['one', 'two', 'three', 'four', 'five']
for num in mylist[::-1]:
    print(num)
```

Output:

five

four

three

two

one

22. Print out the Yang Hui triangle

Source Code:

```
#!/usr/bin/python
# -*- coding: UTF-8 -*-
if __name__ == '__main__':
    arr = []
    for x in range(10):
        arr.append([])
        for y in range(10):
            arr[x].append(0)
    for x in range(10):
        arr[x][0] = 1
        arr[x][x] = 1
    for x in range(2,10):
        for y in range(1,x):
            arr[x][y] = arr[x - 1][y-1] + arr[x - 1][y]
    from sys import stdout
    for x in range(10):
        for y in range(x + 1):
            stdout.write(str(arr[x][y]))
            stdout.write(' ')
        print()
```

Output:

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

1 7 21 35 35 21 7 1

1 8 28 56 70 56 28 8 1

1 9 36 84 126 126 84 36 9 1

23. Get the sum of the '1! + 2! ... + 10!’

Source Code:

```
#!/usr/bin/python
# -*- coding: UTF-8 -*-
num = 0
sum = 0
item = 1
for num in range(1,11):
    item *= num
    sum += item
print ('1! + 2! + 3! + ... + 10! = %d' % sum)
```

Output:

```
1! + 2! + 3! + ... + 10! = 4037913
```

24. Input three numbers, output in order

Source Code:

```
print( 'Please input three numbers.' )
if __name__ == '__main__':
    v1 = int(input(''))
    v2 = int(input(''))
    v3 = int(input(''))
    def swap(p1,p2):
        return p2,p1
    if v1 > v2 : v1,v2 = swap(v1,v2)
    if v1 > v3 : v1,v3 = swap(v1,v3)
    if v2 > v3 : v2,v3 = swap(v2,v3)
    print(v1, v2, v3)
```

(Assume that we input three numbers: 6, 9, 3)

Output:

Please input three numbers.

3 6 9

25. Sort contents alphabetically

Source Code:

```
if __name__ == '__main__':
    str1 = "MySQL in 8 Hours"
    str2 = "C# in 8 Hours"
    str3 = "Python in 8 Hours"
    str4 = "PHP in 8 Hours"
    str5 = "JAVA in 8 Hours"

    print('Before being sorted.')
    print(str1)
    print(str2)
    print(str3)
    print(str4)
    print(str5)

    if str1 > str2 : str1,str2 = str2,str1
    if str1 > str3 : str1,str3 = str3,str1
    if str1 > str4 : str1,str4 = str4,str1
    if str1 > str5 : str1,str5 = str5,str1

    if str2 > str3 : str2,str3 = str3,str2
    if str2 > str4 : str2,str4 = str4,str2
    if str2 > str5 : str2,str5 = str5,str2

    if str3 > str4 : str3,str4 = str4,str3
    if str3 > str5 : str3,str5 = str5,str3
    if str4 > str5 : str4,str5 = str5,str4
```

```
print(' ')
print('After being sorted.')
print(str1)
print(str2)
print(str3)
print(str4)
print(str5)
```

Output:

Before being sorted.
MySQL in 8 Hours
C# in 8 Hours
Python in 8 Hours
PHP in 8 Hours
JAVA in 8 Hours

After being sorted.
C# in 8 Hours
JAVA in 8 Hours
MySQL in 8 Hours
PHP in 8 Hours
Python in 8 Hours

26. Exchange the value of variables

Source Code:

```
def swap(a,b):
    a,b = b,a
    return (a,b)
if __name__ == '__main__':
    x = 100
    y = 200
    print('Before exchange: x = %d, y = %d' % (x,y))
    x,y = swap(x,y)
    print('After exchanged: x = %d, y = %d' % (x,y))
```

Output:

Before exchange: x = 100, y = 200

After exchanged: x = 200, y = 100

27. Output the largest number

Source Code

```
print('Please input three numbers:')
value1 = int(input('Input the 1st number:'))
print(value1)
value2 = int(input('Input the 2nd number:'))
print(value2)
value3 = int(input('Input the 3rd number:'))
print(value3)
if (value1 >= value2) and (value1 >= value3):
   largest = value1
elif (value2 >= value1) and (value2 >= value3):
   largest = value2
else:
   largest = value3
print("The largest number is", largest)
```

Output:

Please input three numbers:
Input the 1st number:20
Input the 2nd number:90
Input the 3rd number:60
The largest number is 90

28. Count the number of each letter

Source Code:

```
letters = 'okey'
str = 'Thank you for reading the series books of Ray Yao team!'
print(str)
str = str.casefold()
num = {}.fromkeys(letters,0)
for ch in str:
   if ch in num:
      num[ch] += 1
print(num)
```

Output:

Thank you for reading the series books of Ray Yao team!

{'o': 6, 'k': 2, 'e': 5, 'y': 3}

29. Calculate the power of a number

Source Code:

```
num = int(input("Please input a number: "))
print(num)
rows = 8
print("The total rows are:", rows)
result = list(map(lambda x: num ** x, range(rows)))
for n in range(rows):
   print(num, "raised to power",n,"is",result[n])
```

(Assume that we input a number 3)

Output:

Please input a number: 3
The total rows are: 8
3 raised to power 0 is 1
3 raised to power 1 is 3
3 raised to power 2 is 9
3 raised to power 3 is 27
3 raised to power 4 is 81
3 raised to power 5 is 243
3 raised to power 6 is 729
3 raised to power 7 is 2187

30. A simple calculator

Source Code:

```
def add(n1, n2):
    return n1 + n2
def subtract(n1, n2):
    return n1 - n2
def multiply(n1, n2):
    return n1 * n2
def divide(n1, n2):
    return n1 / n2
print("One number represents one of the
operations:\n" \
        "Number 1 represents Add\n" \
        "Number 2 represents Subtract\n" \
        "Number 3 represents Multiply\n" \
        "Number 4 represents Divide\n")
choose = int(input("Please choose one of the
operations in 1 / 2 / 3 / 4 : "))
print(choose)
value1 = int(input("Enter the first number: "))
print(value1)
value2 = int(input("Enter the second number: "))
print(value2)
if choose == 1:
    print(value1, "+", value2, "=", add(value1,
value2))
elif choose == 2:
    print(value1, "-", value2, "=", subtract(value1,
```

```
value2))
elif choose == 3:
    print(value1, "*", value2, "=", multiply(value1,
value2))
elif choose == 4:
    print(value1, "/", value2, "=", divide(value1,
value2))
else:
    print("Your input is invalid!")
```

Output:

One number represents one of the operations:

Number 1 represents Add

Number 2 represents Subtract

Number 3 represents Multiply

Number 4 represents Divide

Please choose one of the operations in 1 / 2 / 3 / 4 : 3

Enter the first number: 50

Enter the second number: 2

50 * 2 = 100

Python 100

Questions & Answers

100 Questions

Fill in the blank below, make the program complete **.** The answers are on the last page **.**

(1)

```
age = 15
ticket = "Child Fare" if (age < 16 ) fill in here "Adult Fare"
# conditional expression
print (ticket)
```

A **.** ? B **.** **:** C **.** then D **.** else

(2)

```
trafficLight = fill in here ("Please input traffic light -- red, green or yellow: ")  # user inputs data
if trafficLight == "red":
    print ("The traffic light is " + trafficLight)
elif trafficLight == "green":
    print ("The traffic light is " + trafficLight)
```

```
else:
    print ("The traffic light is " + trafficLight)
```

A . user_input B . input C . user_enter D . enter

(3)

```
import math
r = input("Please enter a radius: ")
def circleArea():
    fill in here   math . pi*pow(r, 2)  # send back result to caller
print ("The circle area is: ", circleArea())
```

A . back B . send C . return D . param

(4)

```
color ={0:"red", 1:"yellow", 2:"green", 3:"white"}
```

```
v = color . values()
for c fill in here v:     # iterate through elements
    print (c)
```

A . in B . at C . on D . by

(5)

```
name = input("Please enter your last name:  ")
isLetter =  name . fill in here    # check if all characters are letters
if isLetter:
    print ("OK ! Valid Last Name ! ")
else:
    print ("No Good ! Invalid Last Name ! ")
```

A . isCharacter() B . isChar() C . isLetter() D . isalpha()

(6)

```
f = fill in here ("tryFile . txt", "w")   # open a file
f . write("I am learning Python programming ! ")
f . close
f = fill in here ("tryFile . txt", "r")     # open a file
print (f . read())
f . close
```

A . unwrap B . unlock C . untie D . open

(7)

```
# program001.py
def red():
    print ("This flower is red")
def yellow():
    print ("This flower is yellow")
def green():
    print ("This flower is green")
```

```
# program002.py
```

fill in here program001 # import a module

program001 . red()

program001 . yellow()

program001 . green()

A . get B . import C . obtain D . acquire

(8)

class Flower:

  def __init__(**fill in here** , name, color): # a keyword represents the current object

    self . name = name

    self . color = color

f = Flower("rose", "red")

print ("The flower's name is " + f . name)

print ("The flower's color is " + f . color)

A . this B . which C . self D . object

(9)

```
try:
    int("ten")
fill in here ValueError as message:  # handle the exception
    print ("Exception occurs ! ", message)
```

A . except B . exception C . catch D . finally

(10)

```
class Dog:              # define a class
    fill in here cry(self):      # define a cry() method
        print ("Dog cries: Wou ! Wou ! ")
class Cat:              # define a class
    fill in here cry(self):      # define a cry() method
        print ("Cat cries: Meo ! Meo ! ")
d = Dog()
d . cry()
c = Cat()
```

c . cry()

A . function B . method C . define D . def

(11)

num1 = **fill in here** (8 . 67) # convert data type

print (num1) # returns 8

num2 = **fill in here** (8 . 67) # convert data type

print (num2) # returns 9 . 0

num3 = **fill in here** (5) # convert data type

print (num3) # returns 5 . 0

A . float round int

B . int float round

C . round int float

D . int round float

(12)

```
n = 0
fill in here n <  9:    # loop statement
    print (n)
    n = n + 1
```

A . switch B . while C . for D . do

(13)

```
import math
print ("ceil(9 . 5) : ", fill in here . ceil(9 . 5))    # math function
print ("floor(9 . 5) : ", fill in here . floor(9 . 5))    # math function
```

A . math B . mathematics C . function D . method

(14)

Structures	Descriptions

fill in here	# store multiple changeable values
fill in here	# store multiple unchangeable values
fill in here	# store multiple unique values
fill in here	# store multiple key **:** value pairs

A **.** Dictionary Set Tuple List

B **.** Tuple Dictionary List Set

C **.** Set Tuple Dictionary List

D **.** List Tuple Set Dictionary

(15)

Functions	**Returned Strings**
fill in here	# replace every old with new
fill in here	# count the number of the characters
fill in here	# change the first letter to uppercase

A **.** count() capitalize() replace()

B **.** replace() capitalize() count()

C **.** replace() count() capitalize()

D . capitalize() count() replace()

(16)

```
import webbrowser
url = "http://www . amazon . com"
webbrowser . fill in here (url)      # open a specified web page
print ("You are visiting "+ url)
```

A . open B . redirect C . href D . link

(17)

```
import math
from fill in here import *      # imports a built-in module
print (math . sqrt(100))
d = datetime . today()
print (d)
```

A . date　　B . time　　C . timedate　　D . datetime

(18)

fill in here BaseClass:　　# define a base class

……

fill in here DerivedClass (BaseClass):　　# define a derived class

……

A . define　　B . class　　C . base　　D . derived

(19)

```
while True:
    try:
        num = int(raw_input("Please enter your ID:  "))
    except ValueError as message:
        print (message)
    fill in here :     # This statement must be executed
```

print ("Remind: please input number only . ")

A . catch B . throw C . throws D . finally

(20)

class BaseClass

def **methodName():** # base method

……

class DerivedClass(BaseClass):

def **fill in here :** # derived method **overrides** base method

……

A . functionName()

B . functionID()

C . methodName()

D . methodID()

(21)

```
# display multiple lines of text .
multiString = fill in here Python
                        is a very
                        good language ! fill in here
print (multiString)
```

A . ‘ ’ B . “ ” C . “ “ ” ” D . “ “ “ ” ” ”

(22)

```
num=200
if num < 100:
    print ("num is less than 100")
fill in here 100 < num < 150:    # run when expression is true
    print ("num is between 100 and 150")
else:
    print ("num is greater than 150")
```

A . elif B . then C . if D . else

(23)

```
def tryFunction( ):
   fill in here tryVar     # defines a global inside the function
   tryVar = "This variable can be referenced in everywhere ."
tryFunction( )    # call a function
print ("tryVar: " + tryVar )     # reference tryVar
```

A . str B . String C . var D . global

(24)

```
lst1 = [0, 1, 2]
lst2 = [3, 4, 5]
myList = lst1 fill in here lst2      # concatenates two lists
print ("myList: ", myList)
print ("myList[5]: ",myList[5])
print ("len(myList): ", len(myList))
```

A . concatenates B . + C . concat D . join

(25)

```
s1 = "JavaScript"     # return the index of first occurrence or -1
print (s1 . fill in here ("a"))    # Output:  1
```

A . index B . search C . find D . seek

(26)

```
import os
print ( os. fill in here ())      # return current  working directory
```

A . cd B . getcwd C . get D . cwd

(27)

```
import math
from fill in here import *      # import a built-in module
d = fill in here . today()
print (d)
```

A . datetime B . date C . time D . now

(28)

```
class Animal:       # define a class
count = 0
def __init__( fill in here ) :    # define a constructor
   self . name = value 1
   self . size = value2
def show(self) :
   print (self . name)
   print (self . size)
```

A . constructor B . this C . arg D . self

(29)

```
# define a function that is a start point of the whole program
def fill in here ():
    pwd = input ("Please enter your password: ")
    if pwd == "12345":
        print ("Password is correct ! ")
    else:
        print ("Password is incorrect ! ")
```

A . start B . initial C . main D . begin

(30)

“open(filename, “argument”)”

arguments	**actions**
+	open file for **fill in here** mode

b	open file in binary mode
t	open file in text mode

A **.** joining

B **.** concatenating

C **.** appending

D **.** reading & writing

(31)

Regular Expressions

\w	Matches word characters **.**
fill in here	Matches non-word characters **.**
\s	Matches space **.**
fill in here	Matches non-space **.**
\d	Matches digitals **.**
fill in here	Matches non-digitals **.**

A . \S \D \W

B . \W \S \D

C . \D \W \S

D . \W \D \S

(32)

myArr = ["a", "b", "c", "d", "e"]

def show(key):

fill in here (key > 5), "Index out of range **!** "

assertion statement

print (key)

key = 5

show (key)

A . debug B . insert C . assertion D . assert

(33)

Which following statement is **not** correct?

A **.** Python is a computer programming language with Python software tools and libraries **.**

B **.** Python is a human-readable programming language which is processed by its interpreter **.**

C **.** Python is a client-side programming language whose interpreter is embedded in web browser software **.**

D **.** Python is a widely used high-level programming language used for general-purpose programming **.**

(34)

Which following is the Python file extension name?

A **. .** pn B **. .** py C **. .** ph D **. .** python

(35)

_____ function outputs the text specified within its parentheses **.**

A **.** alert() B **.** echo() C **.** show() D **.** print()

(36)

Which following is the Python comment symbol?

A . # B . // C . \\ D . < ! - - - - ->

(37)

_____ function outputs the text specified within its parentheses, and waits for the input from user .

A . user_input() B . raw-input() C . input() D . enter()

(38)

Which following statement is **not** correct?

A . The value of Python variable can be referenced by its name .

B . A Python variable may be given an initial value when declaring .

C . A Python variable contains any data type .

D **.** A Python variable is something that holds a changeable value **.**

(39)

Which following line is **not** correct?

+	addition	line 0
-	subtraction	line 1
*	multiplication	line 2
%	Division	line 3
**	Exponent	line 4

A **.** line 1 B **.** line 2 C **.** line 3 D **.** line 4

(40)

Which following expression is **not** correct?

A. a += b means a = (a + b)

B. a *= b means a = (a * b)

C. a %= b means a = (a % b)

D **.** a **!** = b means a = (a **!** b)

(41)

Which following line is **not** correct?

```
a = 100                          # line 1
b = 200                          # line 2
max = a ? (a > b) : b            # line 3
print (max)                      # line 4
```

A **.** line 1 B **.** line 2 C **.** line 3 D **.** line 4

(42)

What is the output according to the following code?

```
result = 10-2*2**2
print (result)
```

A . 2　　B . -6　　C . 256　　D . 16

(43)

Which following statements are **not** correct? (two choices)

A . float(n) converts n to a floating point number .

B . integer(n) converts n to an integer number .

C . string(n) converts n to a normal string .

D . hex(n) converts n to a hexadecimal string .

(44)

Which following statement is **not** correct?

A . list . pop(n) removes an item at index n and returns it .

B . list . number(n) returns the number of times n shown in the list .

C . list . sort() returns the sorted items in order .

D . list . reverse() returns the reverse order items .

(45)

Which following code is correct to define a Tuple?

A . myTuple = [‘Mon’, ‘Tue’, ‘Wed’, ‘Thu’]

B . myTuple = <‘Mon’, ‘Tue’, ‘Wed’, ‘Thu’>

C . myTuple = {‘Mon’, ‘Tue’, ‘Wed’, ‘Thu’}

D . myTuple = (‘Mon’, ‘Tue’, ‘Wed’, ‘Thu’)

(46)

Which following line is **not** correct?

```
color = {'Red', 'Yellow', 'Green'}       # line 1
color . add('Blue')                      # line 2
color . pop()                            # line 3
print (count(color))                     # line 4
```

A . line 1 B . line 2 C . line 3 D . line 4

(47)

Which following statement is **not** correct?

A **.** set **.** add(n) adds one item n to the set **.**

B **.** set **.** copy() copies the whole set **.**

C **.** set **.** discard(n) removes one item at index n **.**

D **.** set **.** pop() removes one random item from the set **.**

(48)

Which following statement is **not** correct?

A **.** List stores multiple unfixed values in an unordered index **.**

B **.** Tuple stores multiple fixed values in an ordered index **.**

C **.** Set stores multiple unique values in an unordered index **.**

D **.** Dictionary stores multiple key:value pairs in an unordered index **.**

(49)

What is the output according to the following code?

```
x = [3,2]
y = 0
y, x[y] = 1,8
print (x)
```

A . [3,2] B . [1,8] C . [1,2] D . [3,8]

(50)

Which following line is **not** correct?

```
num =  8          # line 1
if num > 10:      # line 2
   print ('The number is greater than 10 . ')
then num < 10:            # line 3
   print ('The number is less than 10 . ')
else:                     # line 4
   print ('The number is 10 . ')
```

A . line 1 B . line 2 C . line 3 D . line 4

(51)

What is the output according to the following code?

```
for num in range (1):
    num = num - 1
    print (num)
```

A . -1 B . 0 C . 1 D . 2

(52)

Which following statement is **not** correct?

A . int(-8 . 5) returns -8

B . round(-8 . 5) returns -9 . 0

C . ceil(-8 . 5) returns -8 . 0

D . floor(-8 . 5) returns -8 . 0

(53)

What is the output according to the following code?

```
result = False and (False or (True and (not False)))
print (result)
```

A . True B . False C . 1 D . 0

(54)

What is the output according to the following code?

```
def myFuntion(x, y):
    result =  x + y
    return result
myFuntion(100, 200)
print (result)
```

A . 100 B . 200 C . 300 D . error message

(55)

Which following statement is **not** correct?

A . \’ means single quote

B . \” means double quote

C . \t means table

D . \n means new line

(56)

What is the output according to the following code?

```
myString = 'JavaScript Programming'
print (myString[-22:-17])
```

A . Hours

B . Programming

C . JavaScript

D . Java

(57)

Which following statement is **not** correct?

A **.** str **.** isalnum(): str contains only numbers **.**

B **.** str **.** isalpha(): str contains only letters **.**

C **.** str **.** isdigit(): str contains only digits **.**

D **.** str **.** isidentifier(): str contains only Python identifiers **.**

(58)

Which following statement is **not** good?

A **.** The global variable is created outside functions **.**

B **.** The local variable is created inside functions **.**

C **.** Try to use Global variables as more as possible **.**

D **.** Try to use Local variables as more as possible **.**

(59)

What is the output according to the following code?

```
def function01(num):return num*10
def function02(num):return num*20
def function03(num):return num*30
result = function01(1) + function02(2) + function03(3)
print (result)
```

A . 0　　B . 60　　C . 140　　D . error message

(60)

_____ returns an index of a character in myString, or -1 if the character is not found in myString .

A . myString . find(char)

B . myString . index(char)

C . myString . search(char)

D . myString . seek(char)

(61)

Which following is illegal variable name of Python?

A. mistakeName = 10
B. 100%Correct = 10
C. wrong_Name = 10
D. error100percent =10

(62)

Which following line is **not** correct?

```
bool = false       # line 1
if bool:           # line 2
   print ('Python Programming ! ')       # line 3
else:              # line 4
   print ('JavaScript Programming ! ')
```

A . line 1 B . line 2 C . line 3 D . line 4

(63)

Which following statement is correct?

A . Justify-align code is required in Python code blocks .

B . Right-align code is required in Python code blocks .

C . Left-align code is required in Python code blocks .

D . Indentation of code is required in Python code blocks .

(64)

Which following line is **not** correct?

```
while True:         # line 1
    try:          # line 2
        num = int(input('How old are you?'))        # line 3
        print (num)
    catch NameError:                  # line 4
        print ('Please enter an integer number . ')
```

A . line 1 B . line 2 C . line 3 D . line 4

(65)

The user input is always read as a _____data type .

A . int B . integer C . str D . string

(66)

_____ returns number of times char occurs within myString .

A . myString . size(char)

B . myString . count(char)

C . myString . length(char)

D . myString . number(char)

(67)

Debugging code can be added to the program using _____ keyword to return error messages .

A . assert

B . debug

C . check

D . test

(68)

Which following statement is **not** correct?

About the argument of the open()

A **.** “r+” opens a text file for reading or writing **.**

B **.** “w+” opens a text file for reading or writing **.** (overwrite text)

C **.** “a+” opens a text file for reading or writing **.** (append text)

D **.** “b” opens a text file for both reading and writing from beginning **.**

(69)

Which following line is **not** correct?

```
while True:       # line 1
    try:          # line 2
        num = input('What is your student number?')
        print (num)
    except NameError:       # line 3
        print ('Please do not enter any characters . ')
```

```
    final:          # line 4
        print ('Thank you ! ')
```

A . line 1 B . line 2 C . line 3 D . line 4

(70)

What is the output according to the following code?

```
def flodiv(a,b):
    return 1- a // b ** 1
result = flodiv(100, 100)
print (result)
```

A . -1 B . 0 C . 1 D . 2

(71)

Which following line is **not** correct to declare a class?

```
class Animal:       # line 1
```

```
count = 0              # line 2
def __init__() :       # line 3
  self . name = value 1        # line 4
  self . size = value2
```

A . line 1 B . line 2 C . line 3 D . line 4

(72)

Which following is correct to declare a derived class?

A . class DerivedClass (BaseClass):

B . class DerivedClass extends BaseClass:

C . class DerivedClass inherits BaseClass):

D . class DerivedClass : BaseClass:

(73)

“self” is a variable that refers to the _____ .

A . current class

B . current object

C . global variable

D . local variable

(74)

Which following statement is **not** correct about formatted string?

A . print ("String value is: **%s** " %num) returns a string .

B . print ("Float value is: **%.3f** " %num) returns a floating point number .

C . print ("Octal value is: **%o** " %num) returns an octal number .

D . print ("Integer value is: **%i** " %num) returns an integer .

(75)

Which following is **not** correct about Regular Expression?

A . **?** matches zero or one repetition .

B . **+** matches one or more repetitions .

C . ***** matches zero or more repetitions .

D . **$** matches any characters .

(76)

Which following statement will return **false** ?

A． print ('G' in 'Good')

B． print ('A' not in 'Good')

C． print ('g' in 'Good')

D． print ('g' not in 'Good')

(77)

What is the output according to the following code?

print ('JavaScript'[2: 5])

A． avaSc

B． vaScr

C． ava

D． vaS

(78)

Which following code can return **true** ?

A **.** print ('Java8' **.** isalnum())

B **.** print ('Java8' **.** isalpha())

C **.** print ('Java 8' **.** isalnum())

D **.** print ('java 8' **.** isalpha())

(79)

Which following statement is **not** correct?

A **.** String must be enclosed within either single quote marks or double quote marks **.**

B **.** str object can convert a value to the string data type **.**

C **.** rjust() method adds padding to the left of the string **.**

D **.** ljust() method adds padding to the right of the string **.**

(80)

What is the output according to the following code?

```
myList = ['N', 'B','A', 'C', 'B', 'A']
mySet = set(myList)
myTuple = tuple(mySet)
print (myTuple)
```

A . ('N', 'B', 'A', 'C', 'B', 'A')

B . ('N', 'B', 'A')

C . ('C', 'B', 'A')

D . ('A', 'C', 'B', 'N')

(81)

Which following statement is **not** correct about file open() mode?

A . If open a non-existing file in **r** mode, the interpreter will report an error .

B . If open a non-existing file in **w** mode, a new file will be created .

C . If open an existing file in **w** mode, the original contents will be deleted .

D . If open an existing file in **a** mode, the new contents will be added to the first line of the original contents .

(82)

Which following line is **not** correct?

```
class MyClass:
Greeting = "          # line 1
def __init__(self, Name="My friend"):        # line 2
   self . Greeting = Name + " ! "
function SayHi(self):                        # line 3
   print ("Hi, "+ self . Greeting)
MyObject = MyClass()                # line 4
MyObject . SayHi()
```

A . line 1 B . line 2 C . line 3 D . line 4 .

(83)

Which following statement is **not** correct?

A . In Python, method overloading happens in the same name methods and different parameters

B . The class name should be uppercase and object name should be lowercase .

C . For overriding a base method, the method declaration in the derived class must be exactly the same as that in the base class .

D . You must not pass an argument value to the self argument .

(84)

Which following code is correct to import a module?

A . from myFile . py import myFunction .

B . from myFile import *

C . from myFile . py import*

D . from * . py import myFunction

(85)

Which following line is **not** correct according to the code?

```
name = 'Smith'        # line 1
def myFunction(Nickname):        # line 2
    local name          # line 3
```

```
    name = Nickname              # line 4
    print (name)
myFunction('FatGuy')
```

A . line 1 B . line 2 C . line 3 D . line 4 .

(86)

Which following statement is **not** correct?

A . myString . strip(ch) removes all ch character at the beginning or the end of myString .

B . myString . split(sp) returns a list of substrings of myString which are separate by sp separators .

C . myString . encode() sets the encoding of myString .

D . myString . join(sp) concatenates the myString by sp separators .

(87)

Which following statement is **not** correct?

A . type(100) returns <class ‘int’>

B . type(100 . 00) returns <class ‘float’>

C . type(‘100’) returns <class ‘str’>

D . type(true) returns <class ‘boolean’>

(88)

_____ can visit a specified website .

A . web . hyperlink(url)

B . webbrowser . open(url)

C . browser . link(url)

D . webbrowser . link(url)

(89)

What is the output according to the following code?

```
LuckyNumber = (1, 6, 8, 8)
LuckyNumber = LuckyNumber . append(100)
print (LuckyNumber)
```

A . (1,6,8,8,100)

B . 0

C . error message

D . 100

(90)

Which following line is **not** correct according to the code?

```
nums =[10, 20, 30, 40, 50]       # line 1
foreach n in nums:        # line 2
    if n > 30:          # line 3
        print (n)         # line 4
```

A . line 1 B . line 2 C . line 3 D . line 4 .

(91)

Which following statement is **not** correct?

A . The random module provides a random() method which can produce an integer number between 0 and 1 .

B . The math module provides various math methods such as ceil(), floor(), pow(), sqrt(), sin(), cos() and exp() for programming .

C . The time module provides a time() method which returns the elapsed time since January 1, 1970 .

D . The keyword module provides a “kwlist” attribute which includes a list of all Python keywords .

(92)

What is the output according to the following code?

```
a = 10
b = 20
c = 30
result = a or b and c
print (result)
```

A . true B . false C . 10 D . 20

(93)

Which following line is **not** correct according to the code?

```
num = 100        # line 1
if (num==100):          # line 2
    result = 200          # line 3
    break                  # line 4
print (result)
```

A . line 1 B . line 2 C . line 3 D . line 4 .

(94)

What is the output according to the code?

```
num = 100
for i in range (1, 10):
    num = num + 1
    continue
print (num)
```

A . 108 B . 109 C . 110 D . error message

(95)

Which following line is **not** correct according to the code?

```
num = 10          # line 1
while (num < 10):          # line 2
    return num - 1          # line 3
print (num)          # line 4
```

A . line 1 B . line 2 C . line 3 D . line 4 .

(96)

Which following statement is **not** correct?

A . seek() can specify the position in a file .

B . dir() can show the directory of the file .

C . format() can format the string .

D . zfill() can add zeros to the left of the string .

(97)

Which following code is **not** correct?

```
def myFunction(num):
    try:            # line 1
        return 10/num          # line 2
    except  ZeroDivisionError:
        throws 'error'            # line 3
    finally:          # line 4
        print ("The result is ")
print (myFunction(0))
```

A . line 1 B . line 2 C . line 3 D . line 4 .

(98)

Which following statement is **not** correct about constructor?

A . __init__ is called as a constructor, since it constructs the object .

B . A constructor is called automatically whenever a new object is created .

C . __init__ must have a keyword **self** as its first parameter .

D . __init__ can be overloaded in Python .

(99)

Which following statement is **not** correct?

A . You must call gc() function to execute garbage collection .

B . Class declaration starts with the class keyword .

C . An instance variable is initialized when an object is created .

D . Polymorphism describes the ability to perform different method for different object if a program has one more classes .

(100)

What is the output according to the following code?

```
bool = 'The End'
if (bool == 'The End'):
    print ('Thank you very much ! ')
else:
```

pass

A . The End

B . See You !

C . My Friend !

D . Thank you very much !

100 Answers

01. D	26. B	51. A	76. C
02. B	27. A	52. D	77. D
03. C	28. D	53. B	78. A
04. A	29. C	54. D	79. B
05. D	30. D	55. C	80. D
06. D	31. B	56. D	81. D
07. B	32. D	57. A	82. C
08. C	33. C	58. C	83. A
09. A	34. B	59. C	84. B
10. D	35. D	60. A	85. C
11. D	36. A	61. B	86. D
12. B	37. C	62. A	87. D
13. A	38. B	63. D	88. B
14. D	39. C	64. D	89. C
15. C	40. D	65. C	90. B
16. A	41. C	66. B	91. A
17. D	42. A	67. A	92. C
18. B	43. BC	68. D	93. D
19. D	44. B	69. D	94. D
20. C	45. D	70. B	95. C
21. C	46. D	71. C	96. B
22. A	47. C	72. A	97. C
23. D	48. A	73. B	98. D
24. B	49. D	74. D	99. A
25. C	50. C	75. D	00. D

Recommended Books

Recommended Books by Ray Yao's Team

Advanced C++ in 8 Hours
Advanced Java in 8 Hours
AngularJs in 8 Hours
Asp.net Programming
Awk in 8 Hours
BootStrap in 8 Hours
C# Examples & Interview
C# Programming
C++ Examples & Interview
C++ Programming
Dart in 8 Hours
Django in 8 Hours
Erlang in 8 Hours
Go in 8 Hours
Html Css Examples & Interview
Html Css Programming
Java Examples & Interview
Java Programming
JavaScript Examples & Interview
JavaScript Programming
JQuery Examples & Interview
JQuery Programming
Jsp Servlets Programming
Kotlin in 8 Hours
Linux Command Line
Linux Examples & Interview
Lua in 8 Hours
Matlab in 8 Hours
MySql in 8 Hours
Node.Js in 8 Hours
Numpy in 8 Hours
Perl in 8 Hours
Php Examples & Interview
Php MySql Programming
PowerShell in 8 Hours
Python Examples & Interview
Python Programming
R Programming
React.Js in 8 Hours
Ruby Programming
Rust in 8 Hours
Scala in 8 Hours
Shell Scripting in 8 Hours
Swift in 8 Hours
Tcl in 8 Hours
TypeScript in 8 Hours
Visual Basic Examples & Interview
Visual Basic Programming
Vue.Js in 8 Hours
Xml Json in 8 Hours