MTA Final Document-2

Microsoft Python Exam Preparation
Mr. Yogesh P Murumkar
Bharat Soft Solutions
Machine Learning Data Science and Big Data

Youtube – https://www.youtube.com/c/yogeshmurumkar

1

Which expression evaluates to 4?

- 7-2*3 (Incorrect)
- 7//2-3
- 7/2*3
- 7%2+3 (Correct)

Explanation

7/2*3==>10.5
7%2+3==>4
7//2-3==>0
7-2*3==>1

2

Consider the code :

```
from sys import argv
sum=0
for i in range(2,len(argv)):
        sum += float(argv[i])
print("The Average for {0} is {1:.2f}".format(argv[1],sum/(len(argv)-2)))
```

Which of the following command invocations will generate the output:

The average for Sachin is 20.00

py test.py Sachin 10 20 30 (correct)

py test.py Sachin 10

py test.py 20

py test.py Sachin 10 20

Explanation

By using argv variable present in sys module,we can access command line arguments. argv[0] represents the name of the file.

In the above code {0} will be replaced with argv[1] which is nothing but Sachin

{1:.2f} will be replaced with sum/(len(argv)-2)) and after decimal point 2 digits will be considered.

Consider the code :

```
from sys import argv
sum=0
for i in range(2,len(argv)):
        sum += float(argv[i])
print("The Average for {0} is {1:.2f}".format(argv[1],sum/(len(argv)-2)))
```

Which of the following command invocations will generate the output:

The Average for Durga is 20.00

- (●) py test.py Durga 10 20 30 **(Correct)**
- ( ) py test.py Durga 10
- ( ) py test.py 20
- ( ) py test.py Durga 10 20

Explanation

By using argv variable present in sys module,we can access command line arguments. argv[0] represents the name of the file.
In the above code {0} will be replaced with argv[1] which is nothing but Durga.
{1:.2f} will be replaced with sum/(len(argv)-2)) and after decimal point 2 digits will be considered.

3

Consider the code

```
x=3
x +=1
#Line-1
```

Which line should be inserted at Line-1 so that x value will become 16?

- x**=2 (Correct)
- x*=2 (Incorrect)
- x+=2
- x-=2

Explanation

x+=2===>6
x-=2===>2
x*=2===>8
x**=2==>16

4

Consider the following code:

```
print(type(input('Enter some value:')))
```

if we enter 10 and 10.0 individually for every run what is the output?

- <class 'float'>
  <class 'float'>
- <class 'int'>
  <class 'float'> (Incorrect)
- <class 'int'>
  <class 'int'>
- <class 'str'>
  <class 'str'> (Correct)

Explanation

input() function always returns string type only.

5

Consider the Python code:

```
a=['a','b','c','d']
for i in a:
    a.append(i.upper())
print(a)
```

What is the result?

- SyntaxError
- MemoryError thrown at runtime (Correct)
- ['A','B','C','D'] (Incorrect)
- ['a','b','c','d']

Explanation

In the above code the content will be added keep on and it won't ends. At certain point memory problem will be raised.

6

Consider the Variable declarations:

```
a='5'
b='2'
```

Which of the following expressions are of type str

- [ ] a*b
- [x] a*2 (Correct)
- [x] a+b (Correct)
- [ ] a-b

Explanation

a+b-->str type
a*b-->TypeError: can't multiply sequence by non-int of type 'str'
a-b-->TypeError: unsupported operand type(s) for -: 'str' and 'str'
a*2-->str type

7

Consider the following code:

```
print(type(eval(input('Enter some value:'))))
```

if we enter 10 and 10.0 individually for every run what is the output?

- ○
```
<class 'int'>
<class 'int'>
```

- ○
```
<class 'str'>
<class 'str'>
```

- ◉
```
<class 'int'>
<class 'float'
```
(Correct)

- ○
```
<class 'float'>
<class 'float'>
```

Explanation

input() function always returns str type, but eval() function converts str type corresponding type automatically.

8

Consider the code

```
a=7
b=3
c=5
d=1
```

Which line of the code assigns 9 to the output?

- `output=a%c+1`
- `output=a+d*2` (Correct)
- `output=a+c//d`
- `output=c*d-1`

Explanation

output=a%c+1===>3
output=a+c//d====>12
output=c*d-1====>4
output=a+d*2===>9

9

Consider the code:

```
from sys import argv
print(argv[0])
```

and given the command invocation:

py test.py BHARATSOFT

What is the result?

- ImportError will be thrown at runtime
- `test.py` (Correct)
- IndexError will be thrown at runtime

BHARATSOFT

Explanation

By using argv variable present in sys module,we can access command line arguments. argv[0] represents the name of the file. In the above case it is 'test.py'.

10

Consider the following expression:

```
6//4%5+2**3-2//3
```

This expression results to:

- (●) 9 **(Correct)**
- ( ) -1
- ( ) 25
- ( ) 3

Explanation

6//4%5+2**3-2//3
6//4%5+8-2//3
1%5+8-2//3
1+8-2//3
1+8-0
9

11

`x='TEXT'`

which line of the code will assign 'TT' to the output?

- output=x[0]+x[-1] (Correct)
- output=x[0]+x[2]
- output=x[1]+x[1] (Incorrect)
- output=x[1]+x[4]

Explanation

output=x[0]+x[2]===>TX
output=x[1]+x[1]===>EE
output=x[0]+x[-1]==>TT
output=x[1]+x[4]===>IndexError,because 4 is out of range index

12

Consider the python code:

```
print(10==10  and 20!=20)
print(10==10  or 20!=20)
print( not 10==10)
```

What is the result?

- False
  True (Correct)
  False

- False
  False
  False

- False
  True (Incorrect)
  True

- True
  True
  False

Explanation

If both arguments are True then only 'and' returns True.
If atleast one argument is True then 'or' returns True
not x==>if x is True then it returns False and if x if False then it returns True.
print(10==10 and 20!=20) Here first argument is True and second argument is False.
Hence and operator returns False.
print(10==10 or 20!=20) Here first argument is True and second argument is False.
Hence or operator returns True.
print(not 10==10) prints False to the console.

Consider the code

```
a=float('123.456')
```

Which expression evaluates to 2?

- bool(a)
- int(a)+False
- str(a)
- bool(a)+True **(Correct)**

Explanation

int(a)+False==>123
bool(a)+True==>2
str(a)===>'123.456'
bool(a)===>True

You are intern for XYZ Cars Company.You have to create a function that calculates the average velocity of vehicle on a 2640 foot(1/2 mile)track.

Consider the python code

```
distance=xxx(input('Enter the distance travelled in feet:')) #Line-1
distance_miles=distance/5280
time=yyy(input('Enter the time elapsed in seconds:')) #Line-2
time_hours=time/3600
velocity=distance_miles/time_hours
print('The average Velocity:',velocity,'miles/hour')
```

To generate most precise output, which modifications should be done at Line-1 and at Line-2.

- xxx should be replaced with int and yyy should be replaced with float
- xxx should be replaced with float and yyy should be replaced with int (Incorrect)
- xxx should be replaced with int and yyy should be replaced with int
- xxx should be replaced with float and yyy should be replaced with float (Correct)

Explanation

To get most precise output, we have to typecast into float, so that we won't miss fraction digits also.

15

The XYZ Company has hired you as an intern on the coding team that creates a e-commerce application.You must write a script that asks the user for a value. The value must be used as a whole number in a calculation,even if the user enters a decimal value.

Which of the following meets this requirement?

- `total_items=str(input('How many items you required?'))`
- `total_items=float(input('How many items you required?'))` **(Incorrect)**
- `total_items=int(float(input('How many items you required?')))` **(Correct)**
- `total_items=input('How many items you required?')`

Explanation

The return type of input() function is str by default. If we want to get only whole number from the given string, compulsory we have to type cast to int type. Hence the following is the correct statement we have to use.
If end user provides a float value and it is available in string form,to convert into whole number compulsory first we should convert into float and then into int.
total_items=int(float(input('How many items you required?')))

In which of the following cases, True will be printed to the console ?

- [x] `print('r' in 'durga')` (Correct)

- [ ]
```
a=45
b=45
print(a is not b)
```

- [ ]
```
s1='The Python Course'
s2='The Python Course'.upper()
print(s1 is s2)
```

- [x] `print('is' in 'This IS a Fake News')` (Correct)

- [ ]
```
x=[1,2,3]
y=[1,2,3]
print(x is y)
```

Explanation

a=45
b=45
print(a is not b) #False
Both a and b pointing to the same object
---
s1='The Python Course'
s2='The Python Course'.upper()
print(s1 is s2)#False
s1 and s2 are not pointing to the same object
---
x=[1,2,3]
y=[1,2,3]
print(x is y)#False
x and y are not pointing to the same object
---
print('r' in 'durga') # True
character 'r' present in 'durga' hence 'in' operator returns True
---------- print('is' in 'This IS a Fake News') #True

Consider the code

```
x=2
y=6
x+=2**3
x//=y//2//3
print(x)
```

What is the output?

- 7
- 10 (Correct)
- 9
- 0

Explanation

x+=2**3
x=(x)+(2**3)=10
x//=y//2//3
x=(x)//(y//2//3)
=10//(6//2//3)
=10//(3//3)
=10//1
=10

Given the command invocation:

```
py test.py Durga
```

Which of the following code prints 'Durga' to the console?

- from sys import argv
  `print(argv[1])` **(Correct)**
- from sys import args
  `print(args[1])` **(Incorrect)**
- from sys import argv
  `print(argv[0])`
- from sys import args
  `print(args[0])`

Explanation

By using argv variable present in sys module,we can access command line arguments. argv[0] represents the name of the file. In the above case it is 'test.py'. Hence to access 'Durga', we have to use argv[1]

If the user enters 12345 as input,Which of the following code will print 12346 to the console?

```
count=int(input('Enter count value:'))
print(count+1)
```
(Correct)

```
count=eval(input('Enter count value:'))
print(count+1)
```
(Correct)

```
count=input('Enter count value:')
print(count+1)
```

```
count=input('Enter count value:')
print(int(count)+1)
```
(Correct)

Explanation

The return type of input() function is str type. We have to perform typecasting. As user providing 12345 int value, we have to typecast either by using int() or by using eval() function.

Consider the code

```
t=([10,20],10,False)
```

Which line of the code assigns `<class 'list'>` to x

- `x= type(t[1])`
- `x= type(t[0:])` (Incorrect)
- `x= type(t[0])` (Correct)
- `x= type(t)`

Explanation

x= type(t)===><class 'tuple'>
x= type(t[0])====><class 'list'>
x= type(t[1])===><class 'int'>
x= type(t[0:])===><class 'tuple'>

21

Which expression would evaluate to 2?

- 11/2
- 22%5 (Correct)
- 13//4
- 3**2

Explanation

3**2==>9
22%5===>2
13//4==>3
11/2===>5.5

22

Consider the Code

```
x=3/3+3**3-3
print(x)
```

What is the output?

- 25.0 (Correct)
- 32 (Incorrect)
- 0.11
- 25

Explanation

x=3/3+3**3-3
=3/3+27-3
=1.0+27-3
=25.0

23

Question 24: Skipped

Consider the python code:

```
result=str(bool(1) + float(10)/float(2))
print(result)
```

What is the output?

- 6.0 **(Correct)**
- TypeError
- SyntaxError
- 6

Explanation

/ operator has more precedence than +. Hence float(10)/float(2) will be evaluated first and its result is 5.0. bool(1) is considered as True and again will be considered as 1 whenever we are performing + operator. Hence result is 6.0. str(bool(1) + float(10)/float(2))=str(bool(1) + 10.0/2.0)=str(bool(1) + 5.0)=str(True + 5.0)=str(1 + 5.0)='6.0'

Consider the code:

```
from sys import argv
print(argv[1]+argv[2])
```

and given the command invocation:

```
py test.py 10 20
```

What is the result?

- 30
- ImportError will be thrown at runtime
- 1020 (Correct)
- IndexError will be thrown at runtime

Explanation

By using argv variable present in sys module,we can access command line arguments. argv[0] represents the name of the file. The command line arguments are always considered as str type. Hence + operator meant for concatenation. In this case the output is: 1020

You develop a Python application for your company. You required to accept input from the user and print that information to the user screen.

Consider the code:

```
1  print('Enter Your Name:')
2  #Line-1
3  print(name)
```

At Line-1 which code we have to write?

- `name=input`
- `input(name)`
- `input('name')`
- `name=input()` **(Correct)**

Explanation

To get input from the keyboard, we have to use input() function. Hence the correct statement is: name=input()

Which of the following are valid statements?

- [ ] `type('') is <class 'bool'>`
- [ ] True and False evaluates to False **(Correct)**
- [ ] True or False evaluates to False
- [ ] 5+False evaluates to False
- [ ] True+1 evaluates to 2 **(Correct)**

Explanation

5+False evaluates to 5 but not False
True+1 evaluates to 2
True and False evaluates to False
True or False evaluates to True but not False
type('') is but not

Consider the code:

```
print(not 0)
print(not 10)
print(not '')
print(not 'durga')
print(not None)
```

What is the result?

- True
  False
  False
  False
  True

- True
  False
  True (Correct)
  False
  True

- True
  False
  True
  False
  False

- False
  False
  True
  False
  True

Explanation

In boolean expressions:
0 is treated as False, non-zero treated as True
empty string is treated as False and non-empty string treated as True
None is always treated as False

Consider the Python code:

```
l1=['sunny','bunny','chinny','vinny']
l2=['sunny','bunny','chinny','vinny']
print(l1 is not l2)
print(l1 != l2)
l1=l2
print(l1 is not l2)
print(l1 != l2)
```

What is the result?

- True
  False
  True
  False

- True
  True
  False
  False

- True
  False
  False
  True

- True
  False
  False
  False
  (Correct)

Explanation

if l1 and l2 are not pointing to the same object then only 'l1 is not l2' returns True.
If l1 and l2 are not having same content then only 'l1 != l2' returns True

In which of the following cases we will get `<class 'int'>` as output?

- x=2**2**2
  print(type(x)) **(Correct)**
- x='47'
  print(type(x))
- x=47.0
  print(type(x))
- x=10+20j
  print(type(x))

Explanation

x=47.0
print(type(x))#< class 'float' >
x='47'
print(type(x))#< class 'str' >
x=10+20j print(type(x))#< class 'complex' >
x=2**2**2 print(type(x))#< class 'int' >

Consider the code:

```
s='DURGA SOFT'
```

Which of the following lines will assign 9 to variable result?

- `result = len(s.lstrip())`
- `result = len(s)`
- `result = len(s.rstrip())`
- `result = len(s.strip())`
- `result = len(s.replace(' ',''))` (Correct)

Explanation

strip()==>It will remove spaces present at left and right sides of the string
lstrip()==>It will remove spaces present at only left side of the string
rstrip()==>It will remove spaces present at only right side of the string
Hence lstrip(),rstrip() and strip()methods won't remove the space.
Only replace() method replaces space character with empty string. Hence in this case the result will become 9.

31

Question 32: Skipped

From sys module, by using which variable we can access command line arguments?

- arguments
- argsv
- args
- argv (Correct)

Explanation

By using argv variable present in sys module,we can access command line arguments.

32

Consider the variable declaration

```
b = 'BANANA'
```

Which of the following lines will print 'AA' to the console?

- print(b[3]+b[5]) (Correct)
- print(b[1]+b[5]) (Correct)
- print(b[1]+b[2])
- print(b[1]+b[3]) (Correct)

Explanation

print(b[1]+b[2])==>AN
print(b[1]+b[3])==>AA
print(b[1]+b[5])==>AA
print(b[3]+b[5])==>AA

33

Consider the Python Code

```
l1=['sunny','bunny','chinny','vinny']
l2=['sunny','bunny','chinny','vinny']
print(l1 is l2)
print(l1 == l2)
l1=l2
print(l1 is l2)
print(l1 == l2)
```

What is the result?

- False
  False
  True
  True

- False
  True
  False
  True

- False
  True
  True
  False

- False
  True
  True
  True
  (Correct)

Explanation

== operator is always meant for content comparison
is operator is always meant for reference(address) comparison
if l1 and l2 are pointing to the same object then only 'l1 is l2' returns True.
If l1 and l2 are having same content then only 'l1 == l2' returns True

Consider the following code:

```
v1 = 1
v2 = 0
v1 = v1 ^ v2
v2 = v1 ^ v2
v1 = v1 ^ v2
print(v1)
```

What is the result?

- 0 (Correct)
- 2
- 1
- 3

Explanation

^ is XOR operator.
If both bits are same then result is 0,otherwise result is 1

Consider the code

```
count=input('Enter the number of customers of the bank:')
#Line-1
print(output)
```

Which code inserted at Line-1 will print 20 to the console if we pass 15 as count value from the console?

- `output=int(count)+5` (Correct)
- `output=str(count)+5`
- `output=count+5`
- `output=float(count)+5`

Explanation

output=int(count)+5===>20
output=count+5===>Error,because we can not apply + operator between str and int
output=str(count)+5===>Error,because we can not apply + operator between str and int
output=float(count)+5===>20.0

Consider the code

```
x='10'
y='20'
```

The type of x+y ?

- float
- complex
- str (Correct)
- int

Explanation

If we use + operator between 2 string types the result is always string type

37

Which of the following string declarations spans more than one line and considers whitespace properly when the string is printed to the console?

a. S1='Bharat\n
Software\n
Solutions'

b. S1= ''' Bharat
Software
Solutions'''

c. S1=' Bharat
Software
Solutions''

d. S1=" Bharat
Software
Solutions'"

Explanation

Multi line string literals should be enclosed within triple quotes.

38

Consider the code:

```
lst = [7, 8, 9]
b = lst[:]
print(b is lst)
print(b == lst)
```

What is the result?

- True
  False
- False
  True (Correct)
- True
  True
- False
  False

Explanation

slice operator will create a new object.
== operator is always meant for content comparison
is operator is always meant for reference(address) comparison

39

Which of the following statements are valid?

- The followng expression evaluates to 12
  b=False+5-True+35//4 (Correct)

S=”Sachin Tendulkar is cricket’s great player”

- It causes error because we cannot use double quotes and single quotes simultaneously

The following line will print result:4.5
print('result:',(7/2)+(False or True)+(9%3)) (Correct)

result=456+456.0
type of result is int

Explanation

S=”Sachin Tendulkar is cricket’s great player”

It won't cause any error because we can take single and double quotes simultaneously
---
result=456+456.0
type of result is float
---------- The followng expression evaluates to 12
b=False+5-True+35//4=False+5-True+8=0+5-1+8=12

The following line will print result:4.5
print('result:',(7/2)+(False or True)+(9%3))
(7/2)+(False or True)+(9%3)
=(3.5)+(True)+(0)
=(3.5)+(1)+(0)
=4.5

40

We are developing an app in which students will provide college name and city as input.If the student provides college name as BHARATSOFT and city as Pune,then our application has to provide the following greeting message.

Welcome to BHARATSOFT in Pune

Which of the following code can be used for this requirement?

```
college_name=input('Enter Your College Name:')
city=input('Enter Your City:')
print('Welcome to {} in {}'.format(college_name,city))
```

(Correct)

```
college_name=read('Enter Your College Name:')
city=read('Enter Your City:')
print('Welcome to {} in {}'.format(college_name,city))
```

```
college_name=str('Enter Your College Name:')
city=str('Enter Your City:')
print('Welcome to {} in {}'.format(college_name,city))
```

(Incorrect)

```
college_name=eval('Enter Your College Name:')
city=eval('Enter Your City:')
print('Welcome to {} in {}'.format(college_name,city))
```

Explanation

We should use input() function to read input from the keyboard.