Top of Form

Question 1: **Incorrect**

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

* ​
  1. count=eval(input('Enter count value:'))
  2. print(count+1)

**(Correct)**

* ​
  1. count=input('Enter count value:')
  2. print(count+1)
* ​
  1. count=input('Enter count value:')
  2. print(int(count)+1)

**(Correct)**

* ​
  1. count=int(input('Enter count value:'))
  2. print(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.

Bottom of Form

Top of Form

Question 2: **Correct**

**Consider the code**

1. **count=input('Enter the number of customers of the bank:')**
2. **#Line-1**
3. **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=float(count)+5

* ​

output=count+5

* ​

output=str(count)+5

* ​

output=int(count)+5

**(Correct)**

**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

Bottom of Form

Top of Form

Question 3: **Correct**

**Consider the code**

1. **a=7**
2. **b=3**
3. **c=5**
4. **d=1**

**Which line of the code assigns 9 to the output?**

* ​

output=a+c//d

* ​

output=c\*d-1

* ​

output=a%c+1

* ​

output=a+d\*2

**(Correct)**

**Explanation**

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

Bottom of Form

Top of Form

Question 4: **Incorrect**

**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**

1. **distance=xxx(input('Enter the distance travelled in feet:')) #Line-1**
2. **distance\_miles=distance/5280**
3. **time=yyy(input('Enter the time elapsed in seconds:')) #Line-2**
4. **time\_hours=time/3600**
5. **velocity=distance\_miles/time\_hours**
6. **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 float and yyy should be replaced with float

**(Correct)**

* ​

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 int and yyy should be replaced with float

**Explanation**

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

Bottom of Form

Top of Form

Question 5: **Correct**

**Consider the code**

**a=float('123.456')**

**Which expression evaluates to 2?**

* ​

int(a)+False

* ​

bool(a)

* ​

str(a)

* ​

bool(a)+True

**(Correct)**

**Explanation**

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

Bottom of Form

Top of Form

Question 6: **Correct**

**Consider the code:**

1. **from sys import argv**
2. **print(argv[0])**

**and given the command invocation:**

**py test.py DURGASOFT**

**What is the result?**

* ​

ImportError will be thrown at runtime

* ​

test.py

**(Correct)**

* ​

DURGASOFT

* ​

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. In the above case it is 'test.py'.

Bottom of Form

Top of Form

Question 7: **Correct**

**Consider the python code:**

1. **print(10==10 and 20!=20)**
2. **print(10==10 or 20!=20)**
3. **print( not 10==10)**

**What is the result?**

* ​

True

True

False

* ​

False

False

False

* ​

False

True

False

**(Correct)**

* ​

False

True

True

**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.

Bottom of Form

Top of Form

Question 8: **Correct**

**Which of the following statements are valid?**

* ​

s="Durga Sir's Python Classes are Good"

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

* ​

The followng expression evaluates to 12

   b=False+5-True+35//4

**(Correct)**

* ​

result=456+456.0

   type of result is int

* ​

The following line will print  result:4.5

print('result:',(7/2)+(False or True)+(9%3))

**(Correct)**

**Explanation**

s="Durga Sir's Python Classes are Good"  
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

Bottom of Form

Top of Form

Question 9: **Correct**

**Which expression evaluates to 4?**

* ​

7//2-3

* ​

7-2\*3

* ​

7%2+3

**(Correct)**

* ​

7/2\*3

**Explanation**

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

Bottom of Form

Top of Form

Question 10: **Incorrect**

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

* ​
  1. s1='The Python Course'
  2. s2='The Python Course'.upper()
  3. print(s1 is s2)
* ​

print('is' in 'This IS a Fake News')

**(Correct)**

* ​
  1. a=45
  2. b=45
  3. print(a is not b)

**(Incorrect)**

* ​

print('r' in 'durga')

**(Correct)**

* ​
  1. x=[1,2,3]
  2. y=[1,2,3]
  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

Bottom of Form

Top of Form

Question 11: **Correct**

**Consider the Python Code**

1. **l1=['sunny','bunny','chinny','vinny']**
2. **l2=['sunny','bunny','chinny','vinny']**
3. **print(l1 is l2)**
4. **print(l1 == l2)**
5. **l1=l2**
6. **print(l1 is l2)**
7. **print(l1 == l2)**

**What is the result?**

* ​

False

False

True

True

* ​

False

True

True

False

* ​

False

True

False

True

* ​

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

Bottom of Form

Top of Form

Question 12: **Correct**

**Consider the Python code:**

1. **a=['a','b','c','d']**
2. **for i in a:**
3. **a.append(i.upper())**
4. **print(a)**

**What is the result?**

* ​

['a','b','c','d']

* ​

SyntaxError

* ​

MemoryError thrown at runtime

**(Correct)**

* ​

['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.

Bottom of Form

Top of Form

Question 13: **Correct**

**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])

**(Correct)**

* ​

x= type(t)

* ​

x= type(t[0:])

**Explanation**

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

Bottom of Form

Top of Form

Question 14: **Correct**

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

* ​

argv

**(Correct)**

* ​

arguments

* ​

argsv

* ​

args

**Explanation**

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

Bottom of Form

Top of Form

Question 15: **Correct**

**Consider the Code**

1. **x=3/3+3\*\*3-3**
2. **print(x)**

**What is the output?**

* ​

0.11

* ​

32

* ​

25

* ​

25.0

**(Correct)**

**Explanation**

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

Bottom of Form

Top of Form

Question 16: **Correct**

**Consider the code :**

1. **from sys import argv**
2. **sum=0**
3. **for i in range(2,len(argv)):**
4. **sum += float(argv[i])**
5. **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 20

* ​

py test.py Durga 10 20

* ​

py test.py Durga 10

**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.

Bottom of Form

Top of Form

Question 17: **Correct**

**Which expression would evaluate to 2?**

* ​

11/2

* ​

13//4

* ​

3\*\*2

* ​

22%5

**(Correct)**

**Explanation**

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

Bottom of Form

Top of Form

Question 18: **Incorrect**

**Consider the code**

1. **x=2**
2. **y=6**
3. **x+=2\*\*3**
4. **x//=y//2//3**
5. **print(x)**

**What is the output?**

* ​

10

**(Correct)**

* ​

0

**(Incorrect)**

* ​

9

* ​

7

**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

Bottom of Form

Top of Form

Question 19: **Incorrect**

**Consider the code:**

1. **print(not 0)**
2. **print(not 10)**
3. **print(not '')**
4. **print(not 'durga')**
5. **print(not None)**

**What is the result?**

* ​

True

False

True

False

True

**(Correct)**

* ​

True

False

False

False

True

**(Incorrect)**

* ​

False

False

True

False

True

* ​

True

False

True

False

False

**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

Bottom of Form

Top of Form

Question 20: **Correct**

**x='TEXT'**

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

* ​

output=x[0]+x[2]

* ​

output=x[1]+x[4]

* ​

output=x[1]+x[1]

* ​

output=x[0]+x[-1]

**(Correct)**

**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

Bottom of Form

Top of Form

Question 21: **Correct**

**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?**

* ​
  1. <class 'int'>
  2. <class 'float'>
* ​
  1. <class 'float'>
  2. <class 'float'>
* ​
  1. <class 'str'>
  2. <class 'str'>

**(Correct)**

* ​
  1. <class 'int'>
  2. <class 'int'>

**Explanation**

input() function always returns string type only.

Bottom of Form

Top of Form

Question 22: **Correct**

**Consider the code**

1. **x='10'**
2. **y='20'**

**The type of x+y ?**

* ​

str

**(Correct)**

* ​

float

* ​

complex

* ​

int

**Explanation**

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

Bottom of Form

Top of Form

Question 23: **Correct**

**Consider the code:**

**s='DURGA SOFT'**

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

* ​

result = len(s.lstrip())

* ​

result = len(s.rstrip())

* ​

result = len(s.strip())

* ​

result = len(s)

* ​

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.

Bottom of Form

Top of Form

Question 24: **Incorrect**

**Consider the code:**

1. **lst = [7, 8, 9]**
2. **b = lst[:]**
3. **print(b is lst)**
4. **print(b == lst)**

**What is the result?**

* ​

True

False

* ​

False

False

* ​

False

True

**(Correct)**

* ​

True

True

**(Incorrect)**

**Explanation**

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

Bottom of Form

Top of Form

Question 25: **Correct**

**Consider the code**

1. **x=3**
2. **x +=1**
3. **#Line-1**

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

* ​

x\*=2

* ​

x+=2

* ​

x-=2

* ​

x\*\*=2

**(Correct)**

**Explanation**

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

Bottom of Form

Top of Form

Question 26: **Correct**

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

* ​

x='47'

print(type(x))

* ​

x=2\*\*2\*\*2

print(type(x))

**(Correct)**

* ​

x=10+20j

print(type(x))

* ​

x=47.0

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' >

Bottom of Form

Top of Form

Question 27: **Correct**

**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?**

* ​
  1. <class 'str'>
  2. <class 'str'>
* ​
  1. <class 'int'>
  2. <class 'int'>
* ​
  1. <class 'float'>
  2. <class 'float'>
* ​
  1. <class 'int'>
  2. <class 'float'

**(Correct)**

**Explanation**

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

Bottom of Form

Top of Form

Question 28: **Correct**

**Consider the following expression:**

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

**This expression results to:**

* ​

3

* ​

9

**(Correct)**

* ​

-1

* ​

25

**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

Bottom of Form

Top of Form

Question 29: **Correct**

**Which of the following are valid statements?**

* ​

type('') is <class 'bool'>

* ​

5+False evaluates to False

* ​

True+1 evaluates to 2

**(Correct)**

* ​

True and False evaluates to False

**(Correct)**

* ​

True or False evaluates to False

**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

Bottom of Form

Top of Form

Question 30: **Incorrect**

**Consider the python code:**

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

**What is the output?**

* ​

TypeError

**(Incorrect)**

* ​

SyntaxError

* ​

6.0

**(Correct)**

* ​

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'

Bottom of Form

Top of Form

Question 31: **Correct**

**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[3])

**(Correct)**

* ​

print(b[1]+b[2])

* ​

print(b[1]+b[5])

**(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

Bottom of Form

Top of Form

Question 32: **Incorrect**

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

* ​
  1. s1='durga\n
  2. software\n
  3. solutions'

**(Incorrect)**

* ​
  1. s1='''durga
  2. software
  3. solutions'''

**(Correct)**

* ​
  1. s1="durga
  2. software
  3. solutions"
* ​
  1. s1='durga
  2. software
  3. solutions'

**Explanation**

Multi line string literals should be enclosed within triple quotes.

Bottom of Form

Top of Form

Question 33: **Correct**

**Consider the code:**

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

**and given the command invocation:**

**py test.py 10 20**

**What is the result?**

* ​

IndexError will be thrown at runtime

* ​

30

* ​

1020

**(Correct)**

* ​

ImportError 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

Bottom of Form

Top of Form

Question 34: **Correct**

**Consider the Python code:**

1. **l1=['sunny','bunny','chinny','vinny']**
2. **l2=['sunny','bunny','chinny','vinny']**
3. **print(l1 is not l2)**
4. **print(l1 != l2)**
5. **l1=l2**
6. **print(l1 is not l2)**
7. **print(l1 != l2)**

**What is the result?**

* ​

True

True

False

False

* ​

True

False

True

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

Bottom of Form

Top of Form

Question 35: **Correct**

**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()

Bottom of Form

Top of Form

Question 36: **Incorrect**

**Consider the following code:**

1. **v1 = 1**
2. **v2 = 0**
3. **v1 = v1 ^ v2**
4. **v2 = v1 ^ v2**
5. **v1 = v1 ^ v2**
6. **print(v1)**

**What is the result?**

* ​

3

* ​

0

**(Correct)**

* ​

1

**(Incorrect)**

* ​

2

**Explanation**

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

Bottom of Form

Top of Form

Question 37: **Incorrect**

**Consider the Variable declarations:**

1. **a='5'**
2. **b='2'**

**Which of the following expressions are of type str**

* ​

a\*2

**(Correct)**

* ​

a-b

**(Incorrect)**

* ​

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

Bottom of Form

Top of Form

Question 38: **Correct**

**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=input('How many items you required?')

* ​

total\_items=int(float(input('How many items you required?')))

**(Correct)**

* ​

total\_items=str(input('How many items you required?'))

* ​

total\_items=float(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?')))

Bottom of Form

Top of Form

Question 39: **Correct**

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

**Welcome to DURGASOFT in Hyderabad**

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

* ​
  1. college\_name=eval('Enter Your College Name:')
  2. city=eval('Enter Your City:')
  3. print('Welcome to {} in {}'.format(college\_name,city))
* ​
  1. college\_name=read('Enter Your College Name:')
  2. city=read('Enter Your City:')
  3. print('Welcome to {} in {}'.format(college\_name,city))
* ​
  1. college\_name=input('Enter Your College Name:')
  2. city=input('Enter Your City:')
  3. print('Welcome to {} in {}'.format(college\_name,city))

**(Correct)**

* ​
  1. college\_name=str('Enter Your College Name:')
  2. city=str('Enter Your City:')
  3. print('Welcome to {} in {}'.format(college\_name,city))

**Explanation**

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

Bottom of Form

Top of Form

Question 40: **Incorrect**

**Given the command invocation:**

**py test.py Durga**

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

* ​

from sys import args

print(args[1])

**(Incorrect)**

* ​

from sys import argv

print(argv[0])

* ​

from sys import argv

print(argv[1])

**(Correct)**

* ​

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]

Bottom of Form