|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **No** | **Question** | **A** | **B** | **C** | **D** | **ANS** |
| 1 | which funtion is used to perform input operation in python 3.x | input() | raw\_input() | A only | Both A&B | c |
| 2 | which funtion is used to perform output operation in python 3.x | output() | print " " | print( ) | None | C |
| 3 | what will be the output for the code a=10 A=100 print(a) | 10 | 100 | error | None | A |
| 4 | what will become Data objects in python | mutable | immutable | Both A&B | None | B |
| 5 | a=10 a=100 what will be the output of a now ? | 0 | 100 | 10 | unknownvalue | B |
| 6 | which of the following is invalid | a=100 | \_a=100 | $a=100 | a20=100 | C |
| 7 | which of the following is invalid variable | mydata\_1 | 1mydata\_ | \_ | bar | B |
| 8 | which one is not a keyword | eval | for | pass | assert | A |
| 9 | which one is used to display object type in python | type() | class() | id() | None | A |
| 10 | a=10 type(a) what will be the out put in python 3.x? | <type int> | <class int> | A&B | error | B |
| 11 | How should we perform a\*a | a\*2 | a\*/\*2 | a\*\*2 | a2 | C |
| 12 | How should we got remainder value | a%2 | a%%2 | a/2 | None | A |
| 13 | what will be the Answer of 5//2 | 2.5 | 2 | 2 | 2.2 | B |
| 14 | what will be the Answer of 3 \* 2 \*\* 3 | 18 | 216 | 24 | 18 | C |
| 15 | what will be the Answer of (3\*2 ) \*\* 2 | 36 | 12 | 122 | error | A |
| 16 | a=10 b=20 a,b=b,a what will be the out put of a,b | (10,10) | 10,20 | 20,10 | (20,10) | D |
| 17 | x=(1,2,3,4,5) what will be the output of x x=1,2,3,4,5 | 1 | 12345 | (1,2,3,4,5) | error | C |
| 18 | a=10 a+=10 a\*=2+10 what will be the output of a | 240 | 30 | 50 | error | A |
| 19 |  | syntax error | 11 | 10 | none | A |
| 20 | a=9 b=8 what will be the out put of a & b | TRUE | 1 | 8 | None | A |
| 21 | a="python" print("welcome %d"%a) | welcome python | welcome 6 | Type Error | None | C |
| 22 | l=list('HELLO') print("{0[0]}, {0[1]}".format(l)) | error | HL | HE | H,L | D |
| 23 | what will be the output of hex(32) | 32h | 0x20 | 32 | x32 | B |
| 24 | what will be the output of bin(16) | 0b10000 | b100000 | ob100000 | 100000 | C |
| 25 | what will be the output of '{}'.format([4.56]) | 4 | 4.56 | 4.5 | [4.56] | D |
| 26 |  | equal | equal welcome | welome equal | welcome | B |
| 27 | a=4 b=40 if(a==b):  print("equal")  print("welcome") | equal | error | no output | equal welcome | C |
| 28 | a=4 b=4 if(a==b):  print("equal")  print("welcome") | equal welcome | no output | Indent error | None | C |
| 29 | a=4 b=4 if(a==b): print("equal") print("welcome") | Expected Indent Block | equal welcome | euql | None | A |
| 30 | i = 0 while i < 5:  print(i)  i += 2  if i == 3:  break else:  print(0) | 0 1 2 3 | 0 2 | 0 2 4 0 | 0 2 4 | C |
| 31 | i = 2 while i < 10:  if i % 2 ==0:  if i>4:  break  print(i-1)  i += 2 | 2 4 6 8 | 1 3 | 0 2 4 6 8 10 | None | B |
| 32 | x = "abcdef" i = "a" while i in x:  print(i) | a a a a | a b c d e f | infinite loop | error | C |
| 33 | x = "welcome" i = "c" while i in x:  x = x[:-1]  print(i) | a a a a | c c c c | a a a a a a ……. | c c c c c …… | B |
| 34 | for i in "Abc":  if i.isupper():  print(i.lower())  else:  print(i.upper()) | ABC | Abc | aBC | None | C |
| 35 | x = 'welcome' for i in range(x):  print(i) | error | welcome | 0 1 2 3 4 5 6 | None | A |
| 36 | x = 'welcome' for i in range(len(x)):  print(i) | error | welcome | 0 1 2 3 4 5 6 | None | C |
| 37 | x = 12345 for i in x:  print(i) | 1 2 3 4 5 | 12345 | no output | error | D |
| 38 | list1=list() print list1 | [ ] | list( ) | ( ) | error | A |
| 39 | list1=[1,2,3] list1 \* 2 | 2 4 6 | [2,4,6] | [1,2,3,1,2,3] | None | C |
| 40 | list1=[1,2,3] list1 + 4 print(list1 ) | [1,2,3,4] | [1,2,3] | [5,6,7] | error | D |
| 41 | list1=[1,2,3] list1 + [4 ] print(list1 ) | [1,2,3,4] | [1,2,3] | [5,6,7] | error | A |
| 42 | list1=[1,2,3] list1 + [2,2,2] print(list1 ) | [1,2,3,2,2,2] | [3,4,5] | None | error | A |
| 43 | list1=[1,2,3] list1 [0]=100 print(list1 ) | [1,2,3] | [100,2,3] | None | error | B |
| 44 | list1=[1,2,3] list2=[1,2,3] for x,y in zip(list1,list2): print(x+y) | [1,2,3,1,2,3] | [2,4,6] | [1,1,2,2,3,3] | error | B |
| 45 | list1=list("welcome" ) print(list1) | welcome | "welcome" | ['w','e','l','c','o','m','e'] | error | C |
| 46 | How to shuffle the list 1 | list1.shuffle() | shuffle(list1) | random.shuffle(list1) | None | C |
| 47 | states=['Texas','Dallas','newyork'] print[-2][:-2] | as | Dallas | Dall | sa | C |
| 48 | list1=[1,2,3,4]  list2=[i\*\*2 for i in list1] what will be the output of list2 | [1,2,3,4] | [2,4,9,16] | [1,4,9,16] | None | C |
| 49 | list1=[1,2,3,4]  list2=[5,6,7,8] what will be the output of list1.append(list2) | [1,2,3,4,5,6,7,8] | [1,2,3,4,[5,6,7,8]] | [1,5,2,6,3,7,4,8] | None | B |
| 50 | How to insert 10 in 2 position in list1 | list1.add(2,10) | liat1.append(2,10) | list1.insert(2,10) | None | C |
| 51 | what will be the output of "Welcome to Python".split() | ["welcome","to","python"] | welcome to python | welcome,to,python | None | A |
| 52 | numbers = [1, 2, 3, 4]  numbers.append([5,6,7,8])  print(len(numbers)) | 8 | 5 | 13 | None | C |
| 53 | How in operator is used for to search | list | tuple | dict | a,b,c | D |
| 54 | list1=[1,2,3,4] def fun(x): return x\*\*2 what will be the output of map(fun,list1) | [1,2,3,4] | [1,4,9,16] | [2,8,18,32] | None | B |
| 55 | list1=[1,2,3,4,5,6,7,8] def fun(x): if x%2==0: return x what will be the output of map(fun,list1) | [2,4,6,8] | 2,4,6,8 | [None,2,None,4,None,6,None,8] | None | C |
| 56 | list1=[1,2,3,4,5,6,7,8] def fun(x): if x%2==0: return x what will be the output of filter(fun,list1) | [2,4,6,8] | 2,4,6,8 | [None,2,None,4,None,6,None,8] | None | A |
| 57 | x=1,2,3,4,5 print(x) | [1,2,3,4,5] | (1,2,3,4,5) | 1,2,3,4,5 | error | B |
| 58 | x=1,2,3 a,b,c=x is called ? | packing | unpacking | A&B | error | C |
| 59 | p = (4, 5)  x, y, z = p what will be the output of x,y,z | 4,5,0 | (0,4,5) | (4,5) | error | D |
| 60 | x=(1,2,3)  y=(4,5,6) what will be the output of x+y | (1,2,3,4,5,6) | (5,7,9) | [1,2,3,4,5,6] | error | A |
| 61 | tup1=(1,2,3,4) How to add elements in to tuple1 ? | tup1.add(99) | tup1.append(99) | tup1.extend(99) | None | D |
| 62 | std={'id':101,'sname':"robert",'state':"NY"} How to display name of the student? | std('name') | std.get('sname') | std['name'] | std.sname | B |
| 63 | std={'id':101,'sname':"robert",'state':"NY"} what will be the output of std.keys() | (id,sname,state) | ['id','sname'] | id,sname | None | B |
| 64 | prod={'p1':100,'p2':200,'p3':300,'p4':400}  print maximum price value | prod['p4'] | prod('p4') | max(prod) | max(prod.keys()) | D |
| 65 | prod={'p1':100,'p2':200,'p3':300,'p4':400} maxp=max(prod.values()) for k in prod: if(prod.get(k)==maxp: print(k) | 400 | p4,400 | p4 | error | C |
| 66 | prod={'p1':100,'p2':200,'p3':300,'p4':400} what will be the output of prod.has\_key('p1') | TRUE | FALSE | 1 | 0 | A |
| 67 | prod={'p1':100,'p2':200,'p3':300,'p4':400} what will be the output of prod.has\_key('p10') | TRUE | FALSE | 1 | 0 | B |
| 68 | m = [[x, x \*\*2, x + 2] for x in range(1, 3)] what will be the output ? | [[1, 1, 3], [2, 4, 4]] | [1,1,2,2,3,4] | [[1,1,3],[2,4,4],[3,9,5]] | error | A |
| 69 | x=[1,2,3] y=[4,5,6] for x,y in zip(x,y): print(x+y) | [5,7,9] | [1,2,3,4,5,6] | [1,4,2,5,3,6] | None | A |
| 70 | list1 = [[1,2,3,4], [5,6,7,8]]   for row in range(0, len(list1)):  for column in range(0, len(list1[row])):  print(list1[row][column]) | [2,4,6,8] | [1,3,5,7] | [1,2,3,4,5,6,7,8] | None | C |
| 71 | data = [[[1, 2], [3, 4]], [[5, 6], [7, 8]]] print element 5 in this data list | print(data[1,0,0] | print(data[1][0][0]) | print(data[1][0]) | error | B |
| 72 | a=[10,23,56,[78]] b=list(a) a[3][0]=95  print(b) | [10,23,56,78,95] | [10,23,56,[95]] | [10,23,56,[95,78]] | None | B |
| 73 | print(zip([1,2],["a","b"],[3,4])) | [1,2,"a","b",3,4] | [(1,'a',3),(2,'b',4)] | [(1,2),("a","b"),(3,4)} | error | B |
| 74 | a=[1,2,3,4] b=[sum(a[0:x+2]) for x in range(0,len(a))] print(b) |  |  |  |  |  |
| 75 | a="wel" b=list((x,len(a)) for x in range(0,len(a))) print(b) | [(w,3),(e,3),(l,3)] | [(0,3),(1,3),(2,3)] | [(0,1),(1,2),(2,3)] | None | B |
| 76 | what will be the output of "a"+"ert" | "aert" | (aert) | aert | aert' | D |
| 77 | str1="welcome"  what will be the output of str1[1:] | "w" | "elcome" | elcome' | elcome | B |
| 78 | stra="welcome"  what will be the output of str1[-2:] | "we" | we' | "me" | "ome" | C |
| 79 | which arithmetic operator not allowed in string | + | - | \* | None | B |
| 80 | what will be the output of print("\r\thello") | hello | hello | hello | error | B |
| 81 | str1="abc"  what will be the output of str1.capitalize () | Abc | ABC | Abc' | "ABC" | C |
| 82 | str1="welcome"  what will be the output of str1.center (10,'\*') | \*\*welcome\*\*' | welcome\*\* | \*\*\*\*\*wel\*\*\*\* | welcome | A |
| 83 | str1="aabbbcccdef"  what will be the output of str1.count("c",0,len(str1)) | 2 | 5 | 3 | 4 | C |
| 84 | str1="aabbbcccdef" what will be the output of str1.count("w",0,len(str1)) | -1 | 0 | Null | error | B |
| 85 | str1="welcome"  what will be the output of str1.isalnum() | 1 | 0 | TRUE | FALSE | C |
| 86 | str1="welcome"  what will be the output of str1.islower() | 1 | -1 | TRUE | FALSE | C |
| 87 | str1="welcome"  what will be the output of str1.isupper() | 0 | 1 | TRUE | FALSE | D |
| 88 | str1="welcome"  what will be the output of str1.isdigit() | 0 | 1 | TRUE | FALSE | D |
| 89 | str1=" "  what will be the output of str1.isspace() | -1 | 1 | 0 | TRUE | D |
| 90 | str1="Wel"  what will be the output of str1.lower() | 1 | TRUE | FALSE | None | D |
| 91 | str1=" wel "  what will be the output of str1.strip() | w,e,l | wel | "wel" | None | C |
| 92 | str1="WeL"  what will be the output of str1.swapcase() | "wel" | "WEL" | "wel" | "wEl" | D |
| 93 | what will be the output of max("what r u") | u | y | w | None | C |
| 94 | str1="qwer"  what will be the output ofstr1.find('e') | 1 | 0 | -1 | 2 | A |
| 95 | str1="qwer"  what will be the output of str1.find('p') | 1 | 0 | -1 | 2 | C |
| 96 | To concatenate two strings to a third what statements are applicable ? | s3 = s1 . S2 | s3=s1+s2 | s3=s1.\_add\_.s2 | None | C |
| 97 | what will be the output of ord('a') | 95 | 96 | 97 | 98 | C |
| 98 | what will be the output of str(ord('b')) | "b" | b' | 97 | 98' | D |
| 99 | what will be the output of chr(ord('b')) | "b" | b' | 97 | 98 | A |
| 100 | what will be the output of "abc"+1 | "abc1" | ABC1 | abc1 | error | A |
| 101 | what will be the output of "abcdef".center() | "cd" | abcdef | "abcdef" | error | D |
| 102 | what will be the output of  " a\tb".expandtabs() | a b | a b | ab | "ab" | B |
| 103 | what will be the output of a in "abc" | 1 | 0 | TRUE | FALSE | D |
| 104 | what will be the output of "a" in "abc" | 1 | 0 | TRUE | FALSE | C |
| 105 | what will be the output of "abcccc".count("c) | 3 | 4 | 2 | None | B |
| 106 | what will be the output of "b" not in "abc" | 1 | 0 | -1 | None | D |
| 107 | what will be the output of "while".isidentifier() | 1 | 0 | TRUE | FALSE | C |
| 108 | what will be the output of "qw".isidentifier() | -1 | 0 | TRUE | FALSE | D |
| 109 | what will be the output of set1={} | {} | [] | set([ ]) | None | A |
| 110 | what will be the output of set1=set([]) | {} | [] | set([]) | None | C |
| 111 | what will be the output of et1={1,2,2,3} | {1,2,2,3} | {1,2,3} | [1,2,3,4] | error | B |
| 112 | set1={1,2,3,4} set2={1,2,7,8} what will be the output of set1|set2 | {1,2} | {3,4,7,8} | {1,2,3,4,7,8} | {1,2,1,2,3,4,7,8} | C |
| 113 | set1={1,2,3,4} set2={1,2,7,8} what will be the output of set1&set2 | {1,2} | {3,4,7,8} | {1,2,3,4,7,8} | {1,2,1,2,3,4,7,8} | A |
| 114 | set1={1,2,3,4} set2={1,2,7,8} what will be the output of set1^set2 | {1,2} | {3,4,7,8} | {1,2,3,4,7,8} | {1,2,1,2,3,4,7,8} | B |
| 115 | set1={1,2,3,4} set2={1,2,7,8} what will be the output of set1>set2 | 1 | 0 | -1 | FALSE | D |
| 116 | set1={1,2,3,4} set2={1,2,7,8} what will be the output ofset1.isdisjoint(set2) | {1,2} | {3,4,7,8} | {1,2,3,4,7,8} | none | D |
| 117 | set1={1,2,3,4} what will be the output of set1[2]=45 | {1,2,3,4} | {3,4,7,8} | {1,2,3,45} | none | D |
| 118 | set1={1,2,3,4} what will be the output of print(set1[0]) | {1} | (2} | 0 | none | D |
| 119 | set1={1,2,3,4} what will be the output of del set1[0] | {2,3,4} | {1} | {1,2,3,4} | error | D |
| 120 | set1={1,2,3,4} set2={1,2,7,8} what will be the output of set1+set2 | {1,2,3,4,1,2,7,8} | {1,2,3,4,7,8} | {1,2,3,4,{1,2,7,8}} | error | D |
| 121 | set1={1,2,3} what will be the output of set1.add(3) | {1,2,3,3} | {1,2,3} | {1,2,3,{3}} | None | B |
| 122 | set1={1,2,3} what will be the output of set1.append(3) | {1,2,3,3} | {1,2,3} | {1,2,3,{3}} | error | D |
| 123 | def welcome():  print('welcome')  welcome() welcome() what will be the output code? | welcome welcome | welcome | welcome,welcome | welcomewelcome | A |
| 124 | def printMax(a, b):  if a > b:  print(1)  elif a == b:  print(2)  else:  print(3) printMax(3, 4) | 1 | 2 | 3 | 4 | C |
| 125 | x=10 def fun(): x=20 print(x) fun() what will be the output of code ? | 10 | 20 | 0 | None | B |
| 126 | x=10 def fun(): print(x) fun() | 10 | 20 | 0 | None | A |
| 127 | x-what will be the output of code ? | 10 | 20 | 50 | 30 | A |
| 128 | x=50 def abc():  global x  print(x)  x=10  print(x) abc() what will be the output of code ? | 10 | 20 | 50 | 30 | A |
| 129 | x=50 def abc():   print(x)  x=10  print(x) abc() | 10 | 20 | 50 | error | D |
| 130 | x=lambda p:p+1  what will be the output of x(9) | 9 | 10 | 11 | error | B |
| 131 | a={1:"A",2:"B",3:"C"}  what will be the output of print(a.get(3,4)) | A | B | C | None | C |
| 132 | a={1:"A",2:"B",3:"C"}  what will be the output of print(a.get(4,4)) | A | B | C | None | D |
| 133 | a={1:"A",2:"B",3:"C"}  what will be the output of a.setdeafault(4,"d") | set key,value | {4:"d"} | no change | None | D |
| 134 | x=["a","b"]  what will be the output of map(upper,x) | [A,B] | A,B | a,b | error | D |
| 135 | what will be the output of math.ceil(4.5) | 4 | 5 | 4.5 | None | B |
| 136 | what will be the output of math.floor(4.6) | 4 | 5 | 4.6 | None | A |
| 137 | what will be the output of math.pow(3,3) | 6 | 9 | 27 | None | C |
| 138 | from math import pow what will be the output of pow(pow(2,2),2) | 8 | 4 | 16 | 6 | C |
| 139 | what will be the output of math.fabs(-1) | 1 | 0 | -1 | error | A |
| 140 | what will be the output of math.fact(3) | 3 | 6 | 9 | error | D |
| 141 | what will be the output of math.trunc(3.1) | 1 | 0.1 | 3 | 3.1 | C |
| 142 | what will be the output of math.sqrt(16) | 4 | 4.0 | 4.23 | None | B |
| 143 | which one converts into pattern object ? | re.create(str) | re.compile(str) | re.assemble(str) | None | B |
| 144 | which one used to match starting of string ? | re.create(str) | re.compile(str) | re.assemble(str) | re.match(str) | D |
| 145 | which one used to any position of the string? | re.create(str) | re.compile(str) | re.assemble(str) | None | D |
| 146 | '.' will be used to match except | \n | & | % | @ | A |
| 147 | which will be use to match start and enc of string | carrat | dollar | dollar,carrat | carrat,dollar | D |
| 148 | what will be the output of re.split('[a-c]', 'who am i', re.I) | ['who','am','I'] | ['who','mi'] | ['who'] | None | B |
| 149 | what will be the output of re.sub('morning', 'evening', ' morning') | morning | evening | morning evening | None | B |
| 150 | whilte spaces and comments | re.L | re.X | re.M | re.W | B |
| 151 | which mode is used for Reading Text files | r | w | rb | a | A |
| 152 | which mode is used for Adding to Text files | r | w | rb | a | D |
| 153 | which mode is used for Reading object files | r | w | rb | a | C |
| 154 | which method will be used to read line by line | read() | readline() | line() | readlines() | B |
| 155 | which method will be used to read lines | read() | readline() | line() | readlines() | D |
| 156 | which method will be used to write mutiple line | write() | readline() | writelines() | writeline() | C |
| 157 | which method will be used read csv files | Reader() | read() | readlines() | readrow() | A |
| 158 | which method will be used to read object files | load() | read() | dump() | None | A |
| 160 | which method will be used to write object files | load() | read() | dump() | None | C |
| 161 | which method will be used to read 10 characters | load(10) | read(10) | dump(10) | None | B |
| 162 | which method will be used to returns current position | pos() | read() | tell() | tellpos() | B |
| 163 | which method will be used to place starting position | seek(0,0) | seek(0,1) | seek(0,2) | None | A |
| 164 | which method will be used to place ending position | seek(0,0) | seek(0,1) | seek(0,2) | None | C |
| 165 | which method will be used to place cureent position | seek(0,0) | seek(0,1) | seek(0,2) | None | B |
| 167 | which module is imported for performing object load and dump operation | csv | load | pickle | object | C |
| 168 | which statement is used for Rename a file name | rename(old,new) | rename(new,old) | rename(new) | None | A |
| 169 | which one used Representation of binary data | w | r | + | b | D |
| 170 | what is Deserialiazation | pickling | unpickling | A&B | None | B |
| 171 | what is serialiazation | pickling | unpickling | A&B | None | A |
| 172 | which one is wrong for Read and Write operation | r+ | w+ | a+ | rw | D |
| 173 | How to close file object | close(fp) | fclose(fp) | fp.close() | None | C |
| 174 | How to remove file | remove(fp) | del(filename) | remove(filename) | None | D |
| 175 | next() method of an iterator does not point to any object | StopIteration | Iteration | stop | ALL | A |
| 176 | what is Error in Arithmetic calculations | ArithmeticError | ArithmeticException | Arithmetic | ALL | A |
| 177 | what is Error in Maximum limit | OverFlow | OverFlowError | OverFlowException | None | B |
| 178 | what is the error when division or modulo by zero | ZeroDivisionErr | Zerodivisionerroe | ZeroDivisionError | ALL | C |
| 179 | Import Error occurs due to | No Module Found | Module Not installes | wrong installation | ALL | D |
| 180 | if key is not found then error is | KeyError | IndexErr | KeyException | ALL | A |
| 181 | what is the error trying to access a local variable in a function or method but no value has been assigned to it. | LocalError | UnBoundedLocalError | BoundedLocalError | ALL | B |
| 182 | what is the when input nad output operation is fails | IOError | InputOutputError | Ioerror | ALL | A |
| 183 | list1=[1,2,3]  what will be the output of list1[3] | NameError | ValueError | Indexerror | TypeError | C |
| 184 | what will be the error a[5] | NameError | ValueError | Indexerror | TypeError | A |
| 185 | what will be the error 12 + '10' | NameError | ValueError | Indexerror | TypeError | D |
| 186 | what will be the error int("abc") | NameError | ValueError | Indexerror | TypeError | B |
| 187 | parsing Errors are | syntax error | Module | logical | None | A |
| 188 | what will be the output of 10+2%10 | 2 | 0 | 12 | 10 | C |
| 189 | what will be the output of 1%2//3 | 0 | 1 | 3 | None | A |
| 190 | Right to left associativity | // | >> | << | \*\* | D |
| 191 | which one is Truncation division operator symbol | // | / | % | \ | A |
| 192 | what will be the output of (2\*\*2)\*\*4 | 16 | 256 | 64 | None | B |
| 193 | what will be the output of aB!2.swapcase() | error | Ab!2 | ab!2 | None | A |
| 194 | what will be the output of 'ab cd'.title() | Ab Cd | Ab | Cd | ABCD | A |
| 195 | what will be the output of 'ab cd'.capitalize() | Ab cd | Ab | Cd | ABCD | A |
| 196 | what will be the output of 'abc'.center() | error | abc | \*\*abc\*\* | None | A |
| 197 | what will be the output of "12rN".lower() | 12rN | error | 12rn | None | C |
| 198 | what will be the output of " sdf".lstrip() | " sdf" | "sdf " | "sdf" | error | C |
| 199 | what will be the output of "abc".encode() | "abc" | b"abc" | "123hjk78lp" | None | B |
| 200 | Defualt value in encoding is | utf-6 | utf-8 | utf-16 | utf-32 | B |