

1. Is Python case-sensitive when dealing with identifiers?

- a) yes b) no c) machine dependent d) none of the mentioned

Answer: a

Explanation: Case is always significant.

2. Which of the following is invalid?

- a) `_a = 1` b) `a = 1` c) `str = 1` d) none of the mentioned

Answer: d

Explanation: All the statements will execute successfully but at the cost of reduced readability.

3. Which of the following is an invalid variable?

- a) `my_string_1` b) `1st_string` c) `foo` d) `_`

Answer: b

Explanation: Variable names should not start with a number.

4. All keywords in Python are in

- a) lower case b) UPPER CASE c) Capitalized d) None of the mentioned

Answer: d

Explanation: True, False and None are capitalized while the others are in lower case.

5. Which of the following is true for variable names in Python?

- a) unlimited length
 b) all private members must have leading and trailing underscores
 c) underscore and ampersand are the only two special characters allowed
 d) none of the mentioned

Answer: a

Explanation: Variable names can be of any length

6. Which of the following cannot be a variable?

- a) `_init_` b) `in` c) `it` d) `on`

Answer: b

Explanation: `in` is a keyword

7. Which is the correct operator for power(xy)?

- a) `X^y` b) `X**y` c) `X^^y` d) None of the mentioned

Answer: b

Explanation: In python, power operator is `x**y` i.e. `2**3=8`.

8. Which one of these is floor division?

- a) `/` b) `//` c) `%` d) None of the mentioned

Answer: b

Explanation: When both of the operands are integers then Python chops out the fraction part and gives you the round-off value, to get the accurate answer use floor division. This is floor division. For ex, `5/2 = 2.5` but both of the operands are integers so the answer to this expression in Python is 2. To get the 2.5 answer, use floor division.

9. What is the order of precedence in Python?

- i) Parentheses ii) Exponential iii) Multiplication iv) Division
 v) Addition vi) Subtraction

- a) i,ii,iii,iv,v,vi b) ii,i,iii,iv,v,vi c) ii,i,iv,iii,v,vi d) i,ii,iii,iv,vi,v

Answer: a

Explanation: For order of precedence, just remember this PEMDAS (similar to BODMAS)

10. What is the answer to this expression, $22 \% 3$?

- a) 7 b) 1 c) 0 d) 5

Answer: b

Explanation: The modulus operator gives the remainder. So, $22\%3$ gives the remainder, that is, 1.

11. Operators with the same precedence are evaluated in which manner?

- a) Left to Right b) Right to Left c) Can't say d) None of the mentioned

Answer: a

12. What is the output of this expression, $3*13$?**

- a) 27 b) 9 c) 3 d) 1

Answer: c

Explanation: First this expression will solve $1**3$ because exponential has higher precedence than multiplication, so $1**3 = 1$ and $3*1 = 3$. Final answer is 3.

13. The expression `int(x)` implies that the variable `x` is converted to an integer. State whether true or false.

- a) True b) False

Answer: a

14. Which one of the following has the highest precedence in the expression?

- a) Exponential b) Addition c) Multiplication d) Parentheses

Answer: d

Explanation: Just remember: PEMDAS, that is, Parenthesis, Exponentiation, Division, Multiplication, Addition, Subtraction. Note that the precedence order of Division and Multiplication is the same. Likewise, the order of Addition and Subtraction is also the same.

15. Which of these is not a core data type?

- a) Lists b) Dictionary c) Tuples d) Class

Answer: d

Explanation: Class is a user-defined data type.

16. Given a function that does not return any value, What value is thrown by default when executed in the shell?

- a) Int b) bool c) void d) None

Answer: d

Explanation: Python shell throws a NoneType object back

17. Which of the following will run without errors?

- a) `round(45.8)` b) `round(6352.898,2,5)` c) `round()` d) `round(7463.123,2,1)`

Answer: a

Explanation: Execute help(round) in the shell to get details of the parameters that are passed into the round function.

18. What is the return type of function id() ?

- a) int b) float c) bool d) dict

Answer: a

Explanation: Execute help(id) to find out details in the Python shell. The id() returns an integer address location that is unique.

19. What error occurs when you execute?

apple = mango

- a) SyntaxError b) NameError c) ValueErro d) TypeError

Answer: b

Explanation: Mango is not defined hence name error.

20. Carefully observe the code and give the answer.

1. def example(a):

2. a = a + '2'

3. a = a*2

4. return a

5. >>>example("hello")

- a) indentation Error
 b) cannot perform a mathematical operation on strings
 c) hello2
 d) hello2hello2

Answer: a

Explanation: Python codes have to be indented properly.

21. What data type is the object below?

L = [1, 23, 'hello', 1].

- a) list b) dictionary c) array d) tuple

Answer: a

Explanation: List data type can store any values within it.

22. In order to store values in terms of key and value we use what core data type.

- a) list b) tuple c) class d) dictionary

Answer: d

Explanation: The dictionary stores values in terms of keys and values.

23. What is the average value of the code that is executed below?

1. >>>grade1 = 80

2. >>>grade2 = 90

3. >>>average = (grade1 + grade2) / 2

- a) 85.0 b) 85.1 c) 95.0 d) 95.1

Answer: a

Explanation: Cause a decimal value of 0 to appear as output.

24. Select all options that print hello-how-are-you

- a) `print('hello-', 'how-', 'are-', 'you')`
 b) `print('hello', 'how', 'are', 'you' + '-' * 4)`
 c) `print('hello-' + 'how-' + 'are-' + 'you')`
 d) `print('hello' + '-' + 'how' + '-' + 'are' + 'you')`

Answer: c

Explanation: + operator concatenates strings without any space between them, where as the comma operator adds a space in between each string.

25. What is the return value of trunc() ?

- a) int b) bool c) float d) None

Answer: a

Explanation: `trunc()` returns int value. Execute `help(math.trunc)` to get details.

26. The value of expressions $4/(3*(2-1))$ and $4/3*(2-1)$ is the same. true or false?

- a) True b) False

Answer: a

27. Which of the following operators has its associativity from right to left?

- a) + b) // c) % d) **

Answer: d

Explanation: All of the operators shown above have associativity from left to right, except the exponentiation operator (`**`) which has its associativity from right to left.

28. What is the value of x if:

`x = int(43.55+2/2)`

- a) 43 b) 44 c) 22 d) 23

Answer: b

Explanation: The expression shown above is an example of explicit conversion. It is evaluated as `int(43.55+1) = int(44.55) = 44`. Hence the result of this expression is 44.

29. Which of the following is the truncation division operator?

- a) / b) % c) // d) |

Answer: c

Explanation: `//` is the operator for truncation division. It is called so because it returns only the integer part of the quotient, truncating the decimal part. For example: `20//3 = 6`.

30. What are the values of the following expressions:

`2**(3**2)`

$(2^{**3})^{**2}$ $2^{**3^{**2}}$

- a) 64, 512, 64 b) 64, 64, 64 c) 512, 512, 512 d) 512, 64, 512

Answer: d

Explanation: Expression 1 is evaluated as: 2^{**9} , which is equal to 512. Expression 2 is evaluated as 8^{**2} , which is equal to 64. The last expression is evaluated as $2^{**(3^{**2})}$. This is because the associativity of $**$ operator is from right to left. Hence the result of the third expression is 512.

31. Which of the following expressions results in an error?

- a) float('10') b) int('10') c) float('10.8') d) int('10.8')

Answer: d

Explanation: All of the above examples show explicit conversion. However, the expression `int('10.8')` results in an error.

32. Python files are saved with the extension as _____ ?

- A .python B .pe C .py D .pi

Answer: C**33. What is the name of the GUI that comes in-built as an interactive shell with Python?**

- A) PGUI B) Pyshell C) IDLE D) PythonSh

Answer: C**34. IDLE stands for?**

- A Indigenous Development Lab
B Integrated Development Environment
C Integrated Developers Local Environment
D Integrated Development and Learning Environment

Answer: D IDLE is Python's Integrated Development and Learning Environment.**35. A user-specified value can be assigned to a variable with _____ function.**

- A user B enter C input D value

Answer: C

Python `input()` function is used to take user input. By default, it returns the user input in form of a string. Ex: `city = input("Enter city name:")`

36. User input is read as _____ ?

- A Floating Decimal B Text String C Boolean Value D Integer

Answer: B

Python `input()` function is used to take user input. By default, it returns the user input in form of a text string.

37. Output displayed by the print function will add this invisible character at the end of the line by default _____

A backslash t B backslash n C backslash s D backslash r

Answer: B

backslash n that is - \n

38. Multiple values specified in parentheses to print function will display each value separated with this by default ...

- A Single Space
- B Double Space
- C A new Line
- D Double Lines

Answer: A

39. Which of the following will provide an ! character as alternative separator for the print function?

- A sep is ! B separate = ! C sep >> '!' D sep = '!'

Answer: D

40. Which of the following will provide a * character as alternative line ending for the print function?

- A end to * B end as * C end = '*' D ending = '*'

Answer: C

41. For which type of error does the interpreter halts and reports the error but does not execute the program?

- A Semantic error B Syntax error C Runtime error D All type of errors

Answer: B Syntax error

42. For which type of error does the interpreter runs the program but halts at error and reports the error as an "Exception"?

- A Semantic error B Syntax error C Runtime error D All type of errors

Answer: C Runtime error

43. For which type of error does the interpreter runs the program and does not report an error?

- A Semantic error B Syntax error C Runtime error D All type of errors

Answer: A Semantic error

44: What do you type to enter the interactive help mode of Python?

- A HELP B save C help() D help

Answer: C help()

45. What type of language is Python?

- A High level B Low level C Top level D Bottom level

Answer: A High level

46. Python language was named after?

A Python - the reptile B Monty Python C A pet D A company

Answer: B Monty Python

47. Who is the creator of Python?

A Bill Gates B Guido Van Rossum C Larry Page D Jeff Bezos

Answer: B Guido Van Rossum

48. Which of the following is identified with Python?

A Dynamic typing B Static typing C Slow typing D Auto typing

Answer: A Dynamic typing

49. Which of the following is used to enclose strings?

A ! symbol B Double quotes C Single quotes D Either single quotes or double quotes

Answer: D

50. In IDLE shell, the output will be the same for all the following statements except one. Which one?

A $4*3$ B $60//5$ C $17-5$ D $12/1$

Answer: D $12/1 = 12.0$ (Results in floating value)

51. In IDLE shell, the output will be an error for one of the following statements. Which one?

A $P = \text{'python'} * \text{int('1')}$ B $P = \text{'python'} + 1$ C $P = \text{'python'} + \text{str}(1)$ D $P = \text{'python'} * 1$

Answer: B

52. What is used for multi-line strings in Python?

A) Three braces `{{{ }}}B) Three Colons ::: ::: C) Three hashes ### ### D) Three Quotes ''' '''`

Answer: D

53. What will be the output after the following statements?

```
x = 90
y = 'I ran for %s minutes'
print(y % x)
```

A) y ran for x minutes B) y ran for 90 minutes C) I ran for 90 minutes D) I ran for x minutes

Answer: C

54. What will be the output after the following statements?

```
x = 'She'
```

```

y = 60
z = 'ran for %s minutes'
print(x, z % y)

```

- A) ran for 60 minutes B) she ran for 60 minutes C) She ran for 60 minutes D) x ran for 60 minutes

Answer: C

55. What will be the output after the following statements?

```

x = ' '
print(x*5)

```

- A Displays a tab B Displays 5 spaces C Displays a newline D Displays 10 quotes

Answer: B

56. What will be the output after the following statements?

```

x = 'no'
y = 'yes'
z = 'may be'
a = [y, z, x]
print(a)

```

- A) 'yes', 'may be', 'no' B) 'no', 'may be', 'yes' C) ['no', 'may be', 'yes'] D) ['yes', 'may be', 'no']

Answer: D

57. Which of the following is used by the interpreter to identify code blocks?

- A Braces B Indentation C Commas D Expressions

Answer: B

58. Which of the following claims about Python's object-oriented programming model is true?

- A Classes are real-world entities while objects are not real
 B Objects are real-world entities while classes are not real
 C Both objects and classes are real-world entities
 D All of the above

Answer: B

In object-oriented programming, an object is an instance of a class. Objects are often used to represent real-world entities, but the class itself is just a blueprint or template for creating objects.

59. What in Python is the name of the function inside the class?

- A Object B Method C Attribute D Argument

Answer: B

In object-oriented programming, a function that is defined inside a class is called a method. The term "method" is often used interchangeably with "function,"

but in the context of object-oriented programming, a method is a function that is associated with a class and operates on instances of that class.

60. Which of the following doesn't qualify as a keyword in Python?

A val B raise C try D with

Answer: A

"val" is not a keyword in Python.

Keywords are reserved words in Python that have a specific meaning and cannot be used as identifiers (i.e., names for variables, functions, etc.)

61. Which of the following selections serves as the literal hexadecimal start?

A 00 B 0x C 0X D Both 0x and 0X

Answer: D

x = 0xFF; # x is 255

y = 0XFF; # y is also 255

62. What will be the output after the following statements?

x = 7

if x > 5:

 print(20)

A 20 B 5 C x D 7

Answer: A

63. What will be the output after the following statements?

x = 8

if x > 8:

 print(20)

else:

 print(10)

A) 20 B) x C) 10 D) 8

Answer: C

64. Write a list comprehension to produce the list: [1, 2, 4, 8, 16.....212].

a) [(2**x) for x in range(0, 13)]

b) [(x**2) for x in range(1, 13)]

c) [(2**x) for x in range(1, 13)]

d) [(x**2) for x in range(0, 13)]

Answer: a

Explanation: The required list comprehension will print the numbers from 1 to 12, each raised to 2. The required answer is thus, [(2**x) for x in range(0, 13)].

65. What will be the output of the following code?

x, a, b = 0, 5, 5

if a > 0:

```

if b < 0:
    x = x + 5
elif a > 5:
    x = x + 4
else:
    x = x + 3
else:
    x = x + 2
print(x)

```

- a. 0 b. 4 c. 2 d. 3

Answer: D

67. What will be the output of the following code?

```

var = 10
for i in range(10):
    for j in range(2, 10, 1):
        if var % 2 == 0:
            continue
        var += 1
    var+=1
else:
    var+=1
print(var)

```

- a. 20 b. 21 c. 10 d. 30

Answer: B

68. What will be the output of the following Python code?

```

x = ['ab', 'cd']
for i in x:
    i.upper()
print(x)

```

- a) ['ab', 'cd'] b) ['AB', 'CD'] c) [None, None] d) none of the mentioned

Answer: a

Explanation: The function upper() does not modify a string in place, it returns a new string which isn't being stored anywhere.

69. What will be the output of the following Python code?

```

x = ['ab', 'cd']
for i in x:
    x.append(i.upper())

```

`print(x)`
 a) ['AB', 'CD'] b) ['ab', 'cd', 'AB', 'CD'] c) ['ab', 'cd'] d) Infinite loop

Answer: d

Explanation: The loop does not terminate as new elements are being added to the list in each iteration.

70. What will be the output of the following Python code?

```
i = 1
while True:
    if i%3 == 0:
        break
    print(i)
    i += 1
```

a) 1 2 b) 1 2 3 c) error d) none of the mentioned

Answer: c

Explanation: SyntaxError, there shouldn't be a space between + and = in +=.

71. What will be the output of the following Python code?

```
i = 1
while True:
    if i%2 == 0:
        break
    print(i)
    i += 2
```

a) 1 b) 1 2 c) 1 2 3 4 5 6 ... d) 1 3 5 7 9 11 ...

Answer: d

Explanation: The loop does not terminate since i is never an even number.

72. What will be the output of the following Python code?

```
x = "abcdef"
while i in x:
    print(i, end=" ")
```

a) a b c d e f b) abcdef c) i i i i i ... d) error

Answer: d

Explanation: NameError, i is not defined.

73. What will be the output of the following Python code?

```
x = "abcdef"
i = "a"
while i in x:
```

```
print(i, end = " ")
```

a) no output b) i i i i i ... c) a a a a a a ... d) a b c d e f

Answer: c

Explanation: As the value of i or x isn't changing, the condition will always evaluate to True.

74. What will be the output of the following Python code?

```
x = "abcdef"
i = "a"
while i in x:
    print('i', end = " ")
```

a) no output b) i i i i i ... c) a a a a a a ... d) a b c d e f

Answer: b

Explanation: Here i i i i i ... printed continuously because as the value of i or x isn't changing, the condition will always evaluate to True. But also here we use a citation marks on "i", so, here i treated as a string, not like a variable.

75. What will be the output of the following Python code?

```
for i in range(10):
    if i == 5:
        break
    else:
        print(i)
else:
    print("Here")
```

a) 0 1 2 3 4 Here b) 0 1 2 3 4 5 Here c) 0 1 2 3 4 d) 1 2 3 4 5

Answer: c

Explanation: The else part is executed if control doesn't break out of the loop.

76. What will be the output of the following Python code?

```
x = (i for i in range(3))
for i in x:
    print(i)
```

a) 0 1 2 b) error c) 0 1 2 0 1 2 d) none of the mentioned

Answer: a

Explanation: The first statement creates a generator object.

77. What will be the output of the following Python code?

```
string = "my name is x"
for i in string:
    print (i, end=", ")
```

- a) m, y, , n, a, m, e, , i, s, , x, b) m, y, , n, a, m, e, , i, s, , x
c) my, name, is, x, d) error

Answer: a

Explanation: Variable i takes the value of one character at a time.

78. What will be the output of the following Python code?

```
string = "my name is x"
```

```
for i in string.split():
```

```
    print(i, end=" ")
```

- a) m, y, , n, a, m, e, , i, s, , x, b) m, y, , n, a, m, e, , i, s, , x
c) my, name, is, x, d) error

Answer: c

Explanation: Variable i takes the value of one word at a time.

79. What will be the output of the following Python code snippet?

```
a = [0, 1, 2, 3]
```

```
for a[-1] in a:
```

```
    print(a[-1])
```

- a) 0 1 2 3 b) 0 1 2 2 c) 3 3 3 3 d) error

Answer: b

Explanation: The value of a[-1] changes in each iteration.

80. What will be the output of the following Python code snippet?

```
a = [0, 1, 2, 3]
```

```
for a[0] in a:
```

```
    print(a[0])
```

- a) 0 1 2 3 b) 0 1 2 2 c) 3 3 3 3 d) error

Answer: a

Explanation: The value of a[0] changes in each iteration. Since the first value that it takes is itself, there is no visible error in the current example.

81. What will be the output of the following Python code snippet?

```
string = "my name is x"
```

```
for i in ''.join(string.split()):
```

```
    print(i, end=" ")
```

- a) m, y, , n, a, m, e, , i, s, , x, b) m, y, , n, a, m, e, , i, s, , x
c) my, name, is, x, d) error

Answer: a

Explanation: Variable i takes the value of one character at a time.

82. What will be the output after the following statements?

```
x = '%s MID %s' %('Python', 'Exams')
print(x)
```

- A) Python MID B) MID Exams C) Exams MID Python D) Python MID Exams

Answer: D

83: What will be the output after the following statements?

```
x = 'Monday'
print('Mon' in x)
```

- A 'Mon' in x B 'Monday' in x C True D False

Answer: C

84: What will be the output after the following statements?

```
x = 'Today is a nice day' + \
' I will go for a walk today'
print(x)
```

- A) SyntaxError B) Today is a nice day
C) I will go for a walk today D) Today is a nice day I will go for a walk today

Answer: D

85: What will be the output after the following statements?

```
x = 'Today is a nice day'
y = x[:9] + 'not ' + x[9:]
print(x)
```

- A) TypeError B) Today is a nice day C) SyntaxError D) Today is not a nice day

Answer: D

86. What will be the output after the following statements?

```
x = ('Python')
print(x)
```

- A. ('P', 'y', 't', 'h', 'o', 'n') B. Python C. P y t h o n D. ('Python')

Answer: B

87. What will be the output after the following statements?

```
x = ('Python',)
print(x)
```

- A ('Python',) B Python C P y t h o n D ('Python')

Answer: A

88. What will be the output after the following statements?

```
x = 'Python'
print(list(x))
```

A ('P', 'y', 't', 'h', 'o', 'n') B (Python) C ['P', 'y', 't', 'h', 'o', 'n'] D ['Python']

Answer: C

89. What will be the output after the following statements?

```
x = 'Python'
print(tuple(x))
```

A ('P', 'y', 't', 'h', 'o', 'n') B (Python) C ['P', 'y', 't', 'h', 'o', 'n'] D ['Python']

Answer: A

90. What will be the output of the python code shown below for various styles of format specifiers?

```
x=1234
res='integers:...%d...%-6d...%06d' %(x, x, x)
res
```

- a) 'integers:...1234...1234 ...001234'
- b) 'integers...1234...1234...123400'
- c) 'integers:... 1234...1234...001234'
- d) 'integers:...1234...1234...001234'

Answer: a

Explanation: The code shown above prints 1234 for the format specified %d, '1234 ' for the format specifier %-6d (minus '-' sign signifies left justification), and 001234 for the format specifier %06d. Hence the output of this code is: 'integers:...1234...1234 ...001234'

91. What will be the output of the following Python code snippet?

```
x=3.3456789
'%f | %e | %g' %(x, x, x)
```

- a) Error
- b) '3.3456789 | 3.3456789+00 | 3.345678'
- c) '3.345678 | 3.345678e+0 | 3.345678'
- d) '3.345679 | 3.345679e+00 | 3.34568'

Answer: d

Explanation: The %f %e and %g format specifiers represent floating point numbers in different ways. %e and %E are the same, except that the exponent is in lowercase. %g chooses the format by number content. Hence the output of this code is: '3.345679 | 3.345679e+00 | 3.34568'.

92. What will be the output of the following Python code snippet?

```
'%(qty)d more %(food)s' %({'qty':1, 'food': 'spam'})
```

- a) Error
- b) No output
- c) '1 more foods'
- d) '1 more spam'

Answer: d

Explanation: String formatting also allows conversion targets on the left to refer to the keys in a dictionary coded on the right and fetch the corresponding values. In the code shown above, (qty) and (food) in the format string on the left refers to keys in the dictionary literal on the right and fetch their assorted values. Hence the output of the code shown above is: 1 more spam.

93. Which of these about a dictionary is false?

- a) The values of a dictionary can be accessed using keys
- b) The keys of a dictionary can be accessed using values
- c) Dictionaries aren't ordered
- d) Dictionaries are mutable

Answer: b

Explanation: The values of a dictionary can be accessed using keys but the keys of a dictionary can't be accessed using values.

94. Which of the following is not a declaration of the dictionary?

- a) {1: 'A', 2: 'B'}
- b) dict([[1,"A"],[2,"B"]])
- c) {1,"A",2"B"}
- d) { }

Answer: c

Explanation: Option c is a set, not a dictionary.

95. What will be the output of the following Python code snippet?

```
a={1:"A",2:"B",3:"C"}
```

```
for i,j in a.items():
```

```
    print(i,j,end=" ")
```

- a) 1 A 2 B 3 C
- b) 1 2 3
- c) A B C
- d) 1:"A" 2:"B" 3:"C"

Answer: a

Explanation: In the above code, variables i and j iterate over the keys and values of the dictionary respectively.

96. What will be the output of the following Python code snippet?

```
a={1:"A",2:"B",3:"C"}
```

```
print(a.setdefault(3))
```

- a) {1: 'A', 2: 'B', 3: 'C'}
- b) C
- c) {1: 3, 2: 3, 3: 3}
- d) No method called setdefault() exists for dictionary

Answer: b

Explanation: `setdefault()` is similar to `get()` but will set `dict[key]=default` if key is not already in the dictionary.

97. Read the information given below carefully and write a list comprehension such that the output is: ['e', 'o']

`w="hello"`

`v=('a', 'e', 'i', 'o', 'u')`

- a) `[x for w in v if x in w]` b) `[x for x in w if x in v]`
 c) `[x for x in v if w in w]` d) `[x for v in w for x in w]`

Answer: b

Explanation: The tuple 'v' is used to generate a list containing only vowels in the string 'w'. The result is a list containing only vowels present in the string "hello". Hence the required list comprehension is: `[x for x in w if x in v]`.

98. What will be the output of the following Python code?

`[ord(ch) for ch in 'abc']`

- a) `[97, 98, 99]` b) `['97', '98', '99']` c) `[65, 66, 67]` d) Error

Answer: a

Explanation: The list comprehension shown above returns the ASCII value of each alphabet of the string 'abc'. Hence the output is: `[97, 98, 99]`. Had the string been 'ABC', the output would be: `[65, 66, 67]`.

99. Write a list comprehension for producing a list of numbers between 1 and 1000 that are divisible by 3.

- a) `[x in range(1, 1000) if x%3==0]` b) `[x for x in range(1000) if x%3==0]`
 c) `[x%3 for x in range(1, 1000)]` d) `[x%3=0 for x in range(1, 1000)]`

Answer: b

Explanation: The list comprehension `[x for x in range(1000) if x%3==0]` produces a list of numbers between 1 and 1000 that are divisible by 3.

100. Write a list comprehension equivalent for the Python code shown below.

`for i in range(1, 101):`

`if int(i*0.5)==i*0.5:`

`print(i)`

- a) `[i for i in range(1, 100) if int(i*0.5)==(i*0.5)]` b) `[i for i in range(1, 101) if int(i*0.5)==(i*0.5)]`
 c) `[i for i in range(1, 101) if int(i*0.5)=(i*0.5)]` d) `[i for i in range(1, 100) if int(i*0.5)=(i*0.5)]`

Answer: b

Explanation: The code shown above prints the value 'i' only if it satisfies the condition: `int(i*0.5)` is equal to `(i*0.5)`. Hence the required list comprehension is: `[i for i in range(1, 101) if int(i*0.5)==(i*0.5)]`.