

1. What statement is used to implement a decision?

- A. while
- B. if**
- C. for
- D. import

Section	3.1 The if statement
Title	What is a decision statement?
type	mc
section	3.1 The if Statement
id	testbank-py-1-ch03-01

2. What are the two parts of an if statement?

- A. A condition and a body**
- B. A check and an increment
- C. An increment and a body
- D. An increment and a return value

Section	3.1 The if statement
Title	What are the two parts of an if statement?
type	mc
section	3.1 The if Statement
id	testbank-py-1-ch03-02

3. Which of the following statements is true about the if statement?

- A. The **if** statement can have only one condition that evaluates to an integer value.
- B. The **if** block is optional.
- C. The **else** block is optional.**
- D. The **if** and **else** blocks can be aligned to any indentation level.

Section	3.1 The if statement
Title	Which statement is true about the if statement
type	mc
section	3.1 The if Statement
id	testbank-py-1-ch03-03

4. Which of the following is the correct syntax for an if statement?

A.  
`if (x < 10) size = "small";`

B.  
`if (x < 10)  
 size = "small"  
else (x < 20)  
 size = "medium"`

**C.**  
`if x < 10 :  
 size = "small"  
else :  
 size = "medium"`

D.  
`if x < 10 :  
 size = "small"  
else  
 size = "medium"`

Section	3.1 The if statement
---------	----------------------

<b>Title</b>	What is the correct syntax for an if statement?
<b>type</b>	mc
<b>section</b>	3.1 The if Statement
<b>id</b>	testbank-py-1-ch03-04

5. Which of the following correctly identifies what is wrong with the code snippet below:

```
if y > 300 :
x = y
else :
x = 0
print("x:", x)
```

- A. Nothing, the program runs as intended
- B. The statement after the **if** statement must be indented
- C. The statement after the **if** statement and the statement after the **else** statement must be indented
- D. No colon is needed after the **else** statement

<b>Section</b>	3.1 The if statement
<b>Title</b>	What is the correct syntax for an if statement?
<b>type</b>	mc
<b>section</b>	3.1 The if Statement
<b>id</b>	testbank-py-1-ch03-05

6. Assuming that the user provides 303 as input, what is the output of the following code snippet?

```
y = int(input("Please enter a number: "))
if y > 300 :
    x = y
else :
    x = 0
print("x:", x)
```

- A. **x: 0**
- B. **x: 303**
- C. **x: 300**
- D. There is no output due to a syntax error.

<b>Section</b>	3.1 The if statement
<b>Title</b>	What is the output of an if/else statement given sample input value
<b>type</b>	mc
<b>section</b>	3.1 The if Statement
<b>id</b>	testbank-py-1-ch03-06

7. What is the output of the following code snippet if the **cost** contains 100:

```
if cost > 150 :
    discount = 0.8 * cost
else :
    discount = cost
print("Your cost is:", discount)
```

- A. Nothing, the code contains a syntax error
- B. **Your cost is: 0**
- C. **Your cost is: 80**
- D. **Your cost is: 100**

<b>Section</b>	3.1 The ifstatement
----------------	---------------------

<b>Title</b>	What is the output of a conditional expression?
<b>type</b>	mc
<b>section</b>	3.1 The if Statement
<b>id</b>	testbank-py-1-ch03-10

8. Consider the following code segment:

```
if count > 0 :
    x = x + 1

print(x)
```

If `count` is initialized to -1 and `x` is initialized to 4 then the value displayed by this code segment is:

- A. -1
- B. 0
- C. 4
- D. 5

<b>Section</b>	3.1 The if Statement
<b>Title</b>	Trace an if statement
<b>type</b>	mc
<b>section</b>	3.1 The if Statement
<b>id</b>	testbank-py-1-ch03-11

9. Consider the following code segment:

```
numPizzas = 1
numPeople = 4

if numPeople == 5 :
    numPizzas = 2
```

After this code segment executes, what value is in the variable `numPizzas`?

- A. 1
- B. 2
- C. 4
- D. 5

<b>Section</b>	3.1 The if Statement
<b>Title</b>	Trace an if statement
<b>type</b>	mc
<b>section</b>	3.1 The if Statement
<b>id</b>	testbank-py-1-ch03-12

10. Consider the following code segment:

```
c = 2
b = 1

if b == 0 :
    c = c + 1
else :
    c = c - 1

print(c)
```

What value is printed by this code segment?

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

Section	3.1 The if Statement
Title	Trace an if/else statement
type	mc
section	3.1 The if Statement
id	testbank-py-1-ch03-13

11. Which statement about if statements is **not** correct?

- A. A compound statement requires a colon at the end of the header.
- B. All statements in a statement block must be indented to the same indentation level.
- C. Comments can be indented to any level.
- D. The statements in a statement block must be indented 2 spaces more than the header.

Section	3.1 The if Statement
Title	Which statement about if statements is not correct?
type	mc
section	3.1 The if Statement
id	testbank-py-2-ch03-117

12. The following code snippet contains an error. What is the error?

```
cost = int(input("Enter the cost: "))
if cost > 100
    cost = cost - 10
print("Discounted cost:", cost)
```

- A. Syntax error: missing colon after **if** statement
- B. Logical error: use of an uninitialized variable
- C. Syntax error: missing an **else** statement
- D. Syntax error: incorrect indentation

Section	3.1 The if statement
Title	Find the error in a code snippet containing an if statement
type	mc
from	testbank-py-1-ch03-07
section	3.1 The if Statement
id	testbank-py-2-ch03-07

13. Assuming that the user provides 95 as input, what is the output of the following code snippet?

```
y = int(input("Please enter a number: "))
if y > 300 :
    x = y
else :
    x = 0
print("x:", x)
```

- A. **x: 0**
- B. **x: 95**
- C. **x: 300**
- D. There is no output due to a syntax error

<b>Section</b>	3.1 The if statement
<b>Title</b>	What is the output of an if/else statement for a given input value?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-08
<b>section</b>	3.1 The if Statement
<b>id</b>	testbank-py-2-ch03-08

14. What is printed by the following code snippet if `itemCount` contains a value of 10 and `cost` contains 80:

```
if itemCount > 5 :
    discount = 0.8
    totalCost = cost * discount
    print("Total discounted price is:", totalCost)
```

- A. Nothing, the program will run but not print any results
- B. Total discounted price is: 64.0
- C. Total discounted price is: 0.0
- D. Total discounted price is: 16.0

<b>Section</b>	3.1 The if statement
<b>Title</b>	What is the output of a compound statement?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-09
<b>section</b>	3.1 The if Statement
<b>id</b>	testbank-py-2-ch03-09

15. What is the error in this statement?

```
if count = max :
    print("You win")
```

- A. Equality is evaluated using two equal signs (`==`), not one.
- B. The `print` function should not be indented
- C. There must be an `else` statement
- D. Nothing, if `count` equals `max`, it would print "You win"

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	What is the error in this statement?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-14

16. What is the opposite of this condition: `count > 10`?

- A. `count >= 10`
- B. `count < 9`
- C. `count <= 10`
- D. `count <= 9`

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	What is the opposite of a conditional statement?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-15

17. What is the output of the following code snippet if `count` contains 56?

```
if count % 2 == 0 :
    print("Count is an even number")
else :
```

```
print("Count is an odd number")
```

- A. Count is an even number
- B. Count is an odd number
- C. Nothing, there is a syntax error
- D. Nothing, the program runs but does not print any output

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	What is the output of a code snippet with relational operators?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-16

18. What is the output of the following code snippet if `count` contains 56?

```
if count % 2 == 0 :  
    print("Count is an even number")  
else :  
    print("Count is an odd number")
```

- A. Count is an even number
- B. Count is an odd number
- C. Nothing, there is a syntax error
- D. Nothing, the program runs but does not print any output

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	What is the output of a code snippet with relational operators?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-17

19. What type of operator is `<=` operator?

- A. Lexicographic
- B. Arithmetic
- C. Inequality
- D. Relational

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	What type of operator is <code>!&lt;</code> operator?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-18

20. The operator `>=` stands for

- A. greater than
- B. greater than or equal to
- C. not equal to
- D. this is not a valid Python operator

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	Which relational operator is this?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-19

21. Which statement correctly tests if the user entered the letter `Y`?

- A. `if userInput = "y" :`
- B. `if userInput = "Y" :`
- C. `if userInput == "Y" :`

D. `if userInput == "y" :`

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	Which statement correctly tests for a user input value?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-20

22. Assuming the user enters 15 as input, what is the output of the following code snippet?

```
number = int(input("Please enter a number: "))
if number >= 20 :
    print("The number is big")
else :
    print("The number is small")
```

- A. There is no output due to a syntax error
- B. `The number is big`
- C. `The number is small`
- D. The program runs successfully but does not print any output

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	What is the output of a given code snippet and given input value?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-21

23. What is the output of the following code snippet if the input is 34?

```
number = int(input("Please enter a number: "))
if number != 20 :
    number = number + 1
else :
    number = number - 1
print(number)
```

- A. `34`
- B. `33`
- C. `35`
- D. `36`

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	What is the output of a given code snippet and given input value?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-22

24. Assuming that the user enters a value of 45, what is the output of the following code snippet?

```
number = int(input("Please enter a number: "))
if number < 100 :
    number = number + 5
if number < 500 :
    number = number - 2
if number > 15 :
    number = number + 1
else :
    number = number - 1
print(number)
```

- A. 105
- B. 45
- C. 43
- D. 49

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	What is the output of a code statement containing multiple if statements and a given input value?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-23

25. In Python, which of the following orderings is used to compare strings?

- A. Semantic
- B. Alphabetic
- C. Syntactic
- D. Lexicographic

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	Which ordering is used to compare strings?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-26

26. Which of the following if statements is problematic because of the limited precision of floating-point numbers?

- A. `if 4 // 3 == 1 :`
- B. `if sqrt(2) * sqrt(2) == 2.0 :`
- C. `if "10" == 5 :`
- D. `if 4 <= 4 :`

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	If statements and the limited precision of floating-point numbers
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-28

27. Consider the following code segment:

```
s1 = "CAT"
s2 = "cat"
```

Which of the following if statements has a condition that evaluates to **True**?

- A. `if s1 == s2 :`
- B. `if s1 = s2 :`
- C. `if s1 < s2 :`
- D. `if s1 >= s2 :`

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	Comparing strings with relational operators
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-29

28. Given the following list of strings, what is the correct order using lexicographic ordering: "Ann", "amy", "Heather", "hanna", "joe", "john", "Leo", "Jim" ?

- A. amy, Ann, hanna, Heather, Jim, joe, john, Leo



- B. Ann, Heather, Jim, Leo, amy, hanna, joe, john
- C. amy, hanna, joe, john, Ann, Heather, Jim, Leo
- D. Leo, john, joe, Jim, Heather, hanna, Ann, amy

<b>Section</b>	3.2 Special Topic: Lexicographic Ordering of Strings
<b>Title</b>	Given a list of strings, what is the correct order using lexicographic ordering?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-1-ch03-31

29. Which of the following is **not** an example of a relational operator?

- A. =
- B. <
- C. <=
- D. !=

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	Which of the following is not an example of a relational operator?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-2-ch03-118

30. Which expression is equivalent to the expression shown below?

`floor - 1 < 13`

- A. `13 < floor - 1`
- B. `13 >= floor - 1`
- C. `floor < 12`
- D. `floor - 1 <= 12`

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	Which is expression is equivalent to the expression shown below?
<b>type</b>	mc
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-2-ch03-119

31. A store provides a 10% discount on items with a price of at least \$100. Otherwise, no discount is applicable. Which of the following DOES NOT correctly compute the discount amount when the item's price is stored in the `price` variable?

- A.
 

```
discount = 0
if price >= 100 :
    discount = 0.10 * price
```
- B.
 

```
discount = 0.10 * price
if price <= 100 :
    discount = 0
```
- C.
 

```
discount = 0
if price >= 100 :
    discount = price / 10
```
- D.
 

```
discount = 10
if price >= 100 :
```

```

        discount = 0.1 * price
    else :
        discount = 0

```

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	Which statement DOES NOT correctly compute the discount?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-24
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-2-ch03-24

32. Which of the following conditions is true, given that `num1` contains 3 and `num2` contains 4?

- A. `num1 + 1 < num2`
- B. `num1 + 1 > num2`
- C. `num1 + num2 != 7`
- D. `num1 - num2 <= 0`

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	Which of the following conditions evaluates to True given two variables?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-25
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-2-ch03-25

33. Which condition will cause the statement block of the if statement to execute only when `count` is 0?

- A. `if count = 0 :`
- B. `if count < 0 :`
- C. `if count =< 0 :`
- D. `if count == 0 :`

<b>Section</b>	3.2 Relational Operators
<b>Title</b>	Which condition will cause the statement block of the if statement to execute?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-27
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-2-ch03-27

34. Which statement evaluates to `True` when comparing the two strings:

```

name1 = "Heather"
name2 = "hanna"

```

- A. `name1 == name2`
- B. `name1 > name2`
- C. `name1 < name2`
- D. Relational operators cannot be used to compare strings

<b>Section</b>	3.2
<b>Title</b>	Which statement evaluates to true when comparing two strings?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-30
<b>section</b>	3.2 Relational Operators
<b>id</b>	testbank-py-2-ch03-30

35. Assuming a user enters 30, 20, and 10 as the input values, what is the output of the following code snippet?

```

num1 = int(input("Enter a number: "))
num2 = int(input("Enter a number: "))

```

```

num3 = int(input("Enter a number: "))
if num1 > num2 :
    if num1 > num3 :
        print(num1)
    else :
        print(num3)
else :
    if num2 > num3 :
        print(num2)
    else :
        print(num3)

```

- A. 0
- B. 10
- C. 20
- D. 30

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	What is the output of the nested if code snippet given three input values?
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-33

36. What is the output of the following code snippet?

```

num1 = 100
if num1 < 100 :
    if num1 < 50 :
        num1 = num1 - 5
    else :
        num1 = num1 - 10
else :
    if num1 > 150 :
        num1 = num1 + 5
    else :
        num1 = num1 + 10
print(num1)

```

- A. 95
- B. 100
- C. 105
- D. 110

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	What is the output of an if/else code snippet?
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-35

37. Which of the following options refers to the technique of simulating program execution on a sheet of paper?

- A. Compiling
- B. Prototyping
- C. Debugging
- D. Tracing

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Which refers to the technique of simulating program execution on a sheet of paper?
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches

38. Which of the following code segments is an example of a nested `if` statement?

A.

```
if a == b :  
    print(a)
```

B.

```
if a == b :  
    print(a)  
else :  
    print(b)
```

C.

```
if a == b :  
    print(a)  
    if c == d :  
        print(c)
```

D. `a = a - 1 if a > 0 else a = a + 1`

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Which of the following is an example of a nested if statement?
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-38

39. Consider the following code segment:

```
if a > b :  
    print("X")  
    if a == b :  
        print("Y")
```

What is displayed if `a` is 1 and `b` is 0?

- A. `X`
- B. `Y`
- C. `X` followed by `Y` on the next line
- D. Nothing

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Trace a nested if statement
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-39

40. Consider the following code segment:

```
if a > b :  
    print("X")  
    if a == b :  
        print("Y")
```

What is displayed if `a` is 0 and `b` is 0?

- A. `X`
- B. `Y`
- C. `X` followed by `Y` on the next line
- D. Nothing

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Trace a nested if statement
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-40

41. Consider the following code segment:

```
if a > b :
    print("X")
    if a == b :
        print("Y")
```

What is displayed if **a** is 1 and **b** is 2?

- A. **X**
- B. **Y**
- C. **X** followed by **Y** on the next line
- D. Nothing

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Trace a nested if statement
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-41

42. Consider the following code segment:

```
if a == b :
    print("W")
else :
    print("X")
    if b == c :
        print("Y")
    else :
        print("Z")
```

If **a**, **b** and **c** are all 0 then the output generated by this code segment is:

- A. **W**
- B. **W** followed by **Y** on the next line
- C. **X** followed by **Y** on the next line
- D. **W** followed by **X** on the next line, followed by **Y** on the next line

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Trace nested if/else statements
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-42

43. Consider the following code segment:

```
if a == b :
    print("W")
else :
    print("X")
    if b == c :
        print("Y")
    else :
```

```
print("Z")
```

If **a** is 0, **b** is 1 and **c** is 0 then the output generated by this code segment is:

- A. **W**
- B. **X**
- C. **X** followed by **Y** on the next line
- D. **X** followed by **Z** on the next line

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Trace nested if/else statements
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-43

44. Consider the following code segment:

```
if a == b :  
    print("W")  
else :  
    print("X")  
    if b == c :  
        print("Y")  
    else :  
        print("Z")
```

If **a** is 0, **b** is 1 and **c** is 1 then the output generated by this code segment is:

- A. **W**
- B. **X**
- C. **X** followed by **Y** on the next line
- D. **X** followed by **Z** on the next line

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Trace nested if/else statements
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-44

45. What error will Python display when it attempts to execute the following if/else statement?

```
if a == b :  
    print("Equal")  
else :  
    print("Not Equal")  
    if a > b :  
        print("a is larger")  
    else :  
        print("b is larger")
```

- A. Python will display an error indicating that **==** should be replaced with **=**
- B. Python will display an error indicating that an if statement cannot reside inside the body of an else
- C. Python will display an error indicating that there is a problem with the indentation
- D. No error will be displayed

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Find the error in an if statement

<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-45

46. What error will Python display when it attempts to execute the following if/else statement?

```
if a = b :
    print("Equal")
else :
    print("Not Equal")
    if a > b :
        print("a is larger")
    else :
        print("b is larger")
```

- A. Python will display an error indicating that = is invalid syntax
- B. Python will display an error indicating that an if statement cannot reside inside the body of an else
- C. Python will display an error indicating that there is a problem with the indentation
- D. No error will be displayed

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Find the error in an if statement
<b>type</b>	mc
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-1-ch03-46

47. What is the definition of a nested statement?

- A. A decision statement that is contained inside the statement block of another decision statement
- B. A compound statement that consists of a header and a statement block
- C. A decision statement that immediately follows another decision statement at the same indentation level
- D. A statement that is used to validate user input

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	What is the definition of a nested statement?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-32
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-2-ch03-32

48. Assuming that a user enters 25 for the price of an item, which of the following hand-trace tables is valid for the given code snippet?

```
price = 0
status = ""
price = float(input("Enter the price for your item: "))
if price >= 50 :
    status = "reasonable"
    if price >= 75 :
        status = "costly"
else :
    status = "inexpensive"
    if price <= 25 :
        status = "cheap"
```

A. 

price	status
0	"inexpensive"

	25	"cheap"
B.	price	status
	0	"inexpensive"
	25	"reasonable"
C.	price	status
	0	"inexpensive"
	25	"reasonable"
		"costly"
D.	price	status
	0	"inexpensive"
	25	"costly"

<b>Section</b>	3.3 Nested Branches
<b>Title</b>	Which hand-trace table is valid for this snippet?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-37
<b>section</b>	3.3 Nested Branches
<b>id</b>	testbank-py-2-ch03-37

49. What is the output of the following code snippet when the user enters 75 as the grade?

```
grade = int(input("Enter student grade: "))
if grade >= 90 :
    letterGrade = "A"
if grade >= 80 :
    letterGrade = "B"
if grade >= 70 :
    letterGrade = "C"
if grade >= 60 :
    letterGrade = "D"
else :
    letterGrade = "E"
print(letterGrade)
```

- A. A
- B. B
- C. C
- D. D

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is the output of an if statement with multiple alternatives?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-47

50. What is the wrong with the following code snippet?

```
grade = int(input("Enter student grade: "))
if grade >= 90 :
    letterGrade = "A"
if grade >= 80 :
    letterGrade = "B"
if grade >= 70 :
    letterGrade = "C"
if grade >= 60 :
```



```

    letterGrade = "D"
else :
    letterGrade = "E"
print(letterGrade)

```

- A. Everyone will get an "E"
- B. Anyone with a grade higher than 60 will receive a "D"
- C. Nothing is wrong, students will get the correct grade
- D. The code block will not compile

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is the wrong with the code snippet that has multiple alternatives?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-48

51. What is the output of the following code snippet?

```

x = 20
if x <= 20 :
    print("1", end="")
if x <=40 :
    print("2", end="")
if x <= 30 :
    print("3", end="")

```

- A. 1
- B. 2
- C. 3
- D. 123

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is the output of the following code snippet?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-50

52. Consider the following code snippet:

```

number = int(input("Enter a number: "))
if number > 30 :
    ...
elif number > 20 :
    ...
elif number > 10 :
    ...
else :
    ...

```

Assuming that the user input is 40, which block of statements is executed?

- A. `if number > 30 : ...`
- B. `else if number > 20 : ...`
- C. `else if number > 10 : ...`
- D. `else : ...`

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	Which statement is executed when the user enters 40 for the input value?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives

53. Consider the following code snippet:

```
number = int(input("Enter a number: "))
if number < 10 :
    print("Too small")
elif number < 50 :
    print("Intermediate")
elif number < 100 :
    print("High")
else :
    print("Too high")
```

Assuming that the user input is 60, what is the output of the this code snippet?

- A. Too high
- B. High
- C. Intermediate
- D. Too small

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is output of the if/elif/else snippet with this input?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-52

54. Consider the following code snippet.

```
num1 = 0
num2 = 0
num3 = 0
num4 = 0
num5 = 0
num1 = int(input("Enter a number: "))
num2 = int(input("Enter a number: "))
if num1 < num2 :
    num3 = num1
else :
    num3 = num2

if num1 < num2 + 10 :
    num4 = num1
elif num1 < num2 + 20 :
    num5 = num1
print("num1 =", num1, "num2 =", num2, "num3 =", num3,
      "num4 =", num4, "num5 =", num5)
```

Assuming that the user enters the numbers 20 and 12 as the two input values, what is the output of the code snippet?

- A. num1 = 20 num2 = 12 num3 = 20 num4 = 0 num5 = 20
- B. num1 = 20 num2 = 12 num3 = 12 num4 = 20 num5 = 0
- C. num1 = 20 num2 = 12 num3 = 12 num4 = 0 num5 = 20
- D. num1 = 20 num2 = 12 num3 = 20 num4 = 20 num5 = 0

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is output of the if/else and if/elif snippet with two input values?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-53

55. What is the value of the price variable after the following code snippet is executed?

```
price = 42
if price < 40 :
    price = price + 10
if price > 30 :
    price = price * 2
if price < 100 :
    price = price - 20
```

- A. 42
- B. 52
- C. 84
- D. 64

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is value of a variable after (if/if/elif) snippet is executed?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-54

56. Consider the following code snippet:

```
age = int(input("Enter your age: "))
if age < 10 :
    print("Child")
if age < 30 :
    print("Young Adult")
if age < 70 :
    print("Old")
if age < 100 :
    print("Impressively old")
```

Assuming that the user inputs 80 as the age, what is the output?

- A.  
Child  
Young adult  
Old
- B.  
Young adult  
Old
- C.  
Impressively old
- D.  
Child  
Young adult  
Old  
Impressive old

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is output of the code snippet with this input?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-55

57. Consider the follow code segment. It is supposed to convert numeric marks to letter grades. However,

it may contain a bug. Examine the program, and identify what bug, if any, is present.

```
grade = "F"
if mark >= 80 :
    grade = "A"
if mark >= 70 :
    grade = "B"
if mark >= 60 :
    grade = "C"
if mark >= 50 :
    grade = "D"
```

- A. The greater than or equal signs need to be replaced with equal signs
- B. All instances of if, except the first, need to be replaced with elif
- C. All instances of if, except the first, need to be replaced with else
- D. There is nothing wrong with the code segment (it works as intended)

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is wrong with this multiple alternatives statement?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-58

58. Consider the follow code segment. It is designed to classify widgets as too small if they are less than 10mm in diameter or too large if they are 15mm in diameter or more. Otherwise they should be classified as just right. However, this code may contain a bug. Examine the program, and identify what bug, if any, is present.

```
if size >= 0 :
    print("Too small")
elif size >= 10 :
    print("Just right")
elif size >= 15 :
    print("Too big")
```

- A. The greater than or equal signs need to be replaced with greater than signs
- B. All instances of elif need to be replaced with else
- C. The order of the conditions (and bodies) must be changed
- D. There is nothing wrong with the code segment (it works as intended)

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is wrong with this multiple alternatives statement?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-59

59. Consider the following code segment. It is designed to convert letter grades to grade points. Examine the program, and identify what bug, if any, is present.

```
if letter == "A" :
    gradePoints = 4.0
elif letter == "B" :
    gradePoints = 3.0
elif letter == "C" :
    gradePoints = 2.0
elif letter == "D" :
    gradePoints = 1.0
else :
```

```
gradePoints = 0.0
```

- A. The double equal signs need to be replaced with greater than or equal signs
- B. All instances of elif need to be replaced with else
- C. The order of the conditions (and bodies) must be changed
- D. There is nothing wrong with the code segment (it works as intended)

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What is wrong with this multiple alternatives statement?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-1-ch03-60

60. Assume that the following import statements appear at the beginning of your program:

```
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
from email.mime.image import MIMEImage
from email.mime.application import MIMEApplication
```

Which statement creates a new email message that can contain both text and images?

- A. `msg = MIMEApplication()`
- B. `msg = MIMEImage()`
- C. `msg = MIMEMultipart()`
- D. `msg = MIMEText()`

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	Which statement creates a new email message that can contain both text and images?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-2-ch03-113

61. Which part of an email message includes information about the sender and the recipient?

- A. The application
- B. The attachment
- C. The footer
- D. The header

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	Which part of an email message includes information about the sender and the recipient?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-2-ch03-114

62. What type of object needs to be created to attach a PDF file to an email message?

- A. `MIMEApplication`
- B. `MIMEDocument`
- C. `MIMEImage`
- D. `MIMEPDF`

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What type of object needs to be created to attach a PDF file to an email message?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-2-ch03-115

63. What library needs to be imported to send a message after it has been created?

- A. `email`
- B. `login`

- C. `mimelib`
- D. `smtplib`

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	What library needs to be imported to send a message after it has been created?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-2-ch03-116

64. Which type of statement should be used to choose exactly one of several alternatives?

- A. `if`
- B. `if-elif`
- C. `if-else`
- D. `if-elif-else`

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	Which type of statement should be used to choose exactly one of several alternatives?
<b>type</b>	mc
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-2-ch03-120

65. Given that the following code is incorrect, what code would fix the following code snippet?

```
grade = int(input("Enter student grade: "))
if grade >= 90 :
    letterGrade = "A"
if grade >= 80 :
    letterGrade = "B"
if grade >= 70 :
    letterGrade = "C"
if grade >= 60 :
    letterGrade = "D"
else :
    letterGrade = "E"
print(letterGrade)
```

- A. Change the `if` statements to `elif` statements (except the first one)
- B. Change the `if` statements to `else` statements (except the first one)
- C. Reverse the order of the `if` statements
- D. Change the last statement to `if` instead of `else`

<b>Section</b>	3.4 Multiple Alternatives
<b>Title</b>	How can you correct a code snippet that has multiple alternatives?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-49
<b>section</b>	3.4 Multiple Alternatives
<b>id</b>	testbank-py-2-ch03-49

66. Consider the following code snippet:

```
age = int(input("Enter your age:"))
if age < 10 :
    print("Child", end="")
if age < 30 :
    print("Young Adult", end="")
if age < 70 :
    print("Old", end="")
if age < 100 :
    print("Impressively old", end="")
```

Assuming that the user inputs 30 as the age, what is the output?

- A. ChildYoung adultOldImpressively old
- B. Young adultOldImpressively old
- C. OldImpressively old
- D. Impressively old

Section	3.4 Multiple Alternatives
Title	What is output of the code snippet with this input?
type	mc
section	3.4 Multiple Alternatives
id	testbank-py-1-ch03-56

67. Consider the following code snippet:

```
age = int(input("Enter your age: "))
if age < 10 :
    print("Child", end="")
if age < 30 :
    print("Young Adult", end="")
if age < 70 :
    print("Old", end="")
if age < 100 :
    print("Impressively old", end="")
```

Assuming that the user inputs 5 as the age, what is the output?

- A. Child
- B. ChildYoung Adult
- C. ChildYoung AdultOld
- D. ChildYoung adultOldImpressively old

Section	3.4 Multiple Alternatives
Title	What is output of the code snippet with this input?
type	mc
from	testbank-py-1-ch03-57
section	3.4 Multiple Alternatives
id	testbank-py-2-ch03-57

68. Flowcharts are made up of all the following elements, EXCEPT:

- A. elements for tasks
- B. elements for input/output
- C. elements for pseudocode
- D. elements for decisions

Section	3.5 Problem Solving: Flowcharts
Title	What are the parts of a flowchart?
type	mc
section	3.5 Problem Solving: Flowcharts
id	testbank-py-1-ch03-61

69. The flowchart shows the order in which steps should be executed, and the diamond-shaped boxes indicate:

- A. input
- B. algorithms
- C. tasks
- D. decision statements

Section	3.5 Problem Solving: Flowcharts
Title	What are the parts of a flowchart?
type	mc

**section** 3.5 Problem Solving: Flowcharts  
**id** testbank-py-1-ch03-62

70. A messy network of possible pathways through a program is referred to as:

- A. knotted logic
- B. spaghetti code
- C. twisted conditions
- D. zigzag functions

**Section** 3.5 Problem Solving: Flowcharts  
**Title** A messy network of possible pathways through a program is referred to as:  
**type** mc  
**section** 3.5 Problem Solving: Flowcharts  
**id** testbank-py-2-ch03-121

71. When testing code for correctness, it always makes sense to

- A. Aim for complete coverage of all decision points
- B. Identify boundary cases and test them
- C. Check all cases using hand-tracing
- D. Assume invalid input will never occur

**Section** 3.6 Problem Solving: Test Cases  
**Title** What is reasonable for checking/testing code?  
**type** mc  
**section** 3.6 Problem Solving: Test Cases  
**id** testbank-py-1-ch03-63

72. Consider the following code segment:

```
if a == 0 :  
    print("a is 0")  
elif a < 0 :  
    print("a is less than 0")  
else :  
    print("a is greater than 0")
```

What is the minimum number of test cases needed to test every line of code in this segment?

- A. 2
- B. 3
- C. 4
- D. 5

**Section** 3.6 Problem Solving: Test Cases  
**Title** How many test cases are needed to test every line of code in this segment?  
**type** mc  
**section** 3.6 Problem Solving: Test Cases  
**id** testbank-py-1-ch03-64

73. Which operators listed below are considered boolean operators:

- A. < / >
- B. and / or
- C. == / !=
- D. <= / >=

**Section** 3.7 Boolean Variables and Operators  
**Title** What are boolean operators?  
**type** mc  
**section** 3.7 Boolean Variables and Operators



74. Consider the following code snippet:

```
emp = int(input("Enter Celsius temperature: "))
if temp > 0 and temp < 100 :
    print("Liquid")
if temp <= 0 or temp >= 100 :
    print("Not liquid")
```

Assuming the user enters a value of 120, what will be the output:

- A. Nothing is printed
- B. Liquid
- C. Not Liquid
- D. LiquidNotLiquid

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Given a code snippet and an input value, what output is produced?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-67

75. Which of the following variables is used to store a condition that can be either **True** or **False**?

- A. Logical
- B. Boolean
- C. Algebraic
- D. Conditional

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	What kind of variable is used to store a true/false condition?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-68

76. Rewrite the following algebraic expression to an equivalent Python expression:

$32 \leq \text{temp} \leq 100$

- A. `if temp <= 32 and temp <= 100`
- B. `if temp <= 32 or temp <= 100`
- C. `if temp >= 32 and temp <= 100`
- D. `if temp >= 32 or temp <= 100`

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Rewrite an algebraic expression into the equivalent Python expression?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-71

77. What value causes the following logical expression to 'short-circuit'?

```
if temp >= 32 and temp <= 100
```

- A. `temp = 0`
- B. `temp = 32`
- C. `temp = 100`
- D. `temp = 75`

<b>Section</b>	3.7 Boolean Variables and Operators
----------------	-------------------------------------

<b>Title</b>	What value causes a given expression to short-circuit?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-72

78. Using De Morgan's law, what is the equivalent to this statement?

```
if not (state == "PA" or state == "OH")
```

- A. `if state != "PA" and state != "OH"`
- B. `if state != "PA" or state != "OH"`
- C. `if state == "PA" and state == "OH"`
- D. `if state == "PA" or state == "OH"`

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Using De Morgan's law, identify an equivalent expression?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-74

79. Using De Morgan's law, what is the equivalent to this statement?

```
if not (state == "PA" and state == "OH")
```

- A. `if state != "PA" and state != "OH"`
- B. `if state != "PA" or state != "OH"`
- C. `if state == "PA" and state == "OH"`
- D. `if state == "PA" or state == "OH"`

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Using De Morgan's law, identify an equivalent expression?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-75

80. Consider the following code snippet:

```
attendance = True
failed = False
```

Which of the following `if` statements include a condition that evaluates to `True`?

- A. `if attendance == "true" :`
- B. `if attendance :`
- C. `if failed :`
- D. `if attendance == failed :`

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	What if statements include a condition that evaluates to true?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-76

81. Consider the following code snippet:

```
age = int(input("Enter your age: "))
if age < 13 :
    print("Child", end="")
if age >= 13 and age <= 19 :
    print("Teen", end="")
```

```

if age > 19 and age < 30 :
    print("Young adult", end="")
if age >= 30 and age <= 50 :
    print("Adult", end="")
if age > 50 :
    print("Young at heart", end="")

```

Assuming that the user enters 55 as the age, what is the output?

- A. Teen
- B. Young at heart
- C. Child
- D. Adult

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	What is output of the code snippet with this input?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-77

82. Given the following code snippet:

```

grade = int(input("Enter student grade: "))
if grade >= 90 :
    letterGrade = "A"
elif grade >= 80 and grade < 90 :
    letterGrade = "B"
elif grade >= 70 and grade < 80 :
    letterGrade = "C"
elif grade >= 60 and grade < 70 :
    letterGrade = "D"
else :
    letterGrade = "E"
print(letterGrade)

```

what is value of `grade` when the user enters 75?

- A. "A"
- B. "B"
- C. "C"
- D. "D"

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	What is the output of a code snippet given a specific input value?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-80

83. Which of the following operators is used to invert a conditional statement?

- A. or
- B. and
- C. not
- D. equal

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Which operator is used to invert a conditional statement?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-81

84. Given that the following code snippet:

```
isFelon = False
answer = input("have you ever committed a felony? ")
if answer == "Yes" or answer == "yes" :
    isFelon = True
age = int(input("what is your age? "))
```

which statement assigns the variable `mayVote` a value of `True` if a person may vote if they are 18 or older and not a felon?

- A. `mayVote = age > 18 or not isFelon`
- B. `mayVote = not ( age >= 18 and isFelon)`
- C. `mayVote = age >= 18 and not isFelon`
- D. `mayVote = not ( age >= 18 or isFelon)`

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Which of the following statements assigns the Boolean variable correctly?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-83

85. Given the following code snippet:

```
MIN_SPEED = 45
MAX_SPEED = 65
speed = 55
if not (speed < MAX_SPEED) :
    speed = speed - 10
if not (speed > MIN_SPEED) :
    speed = speed + 10
print(speed)
```

what output is produced?

- A. 45
- B. 55
- C. 65
- D. 50

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	What is the output of a Boolean not code snippet?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-84

86. Given the following code snippet:

```
score = 0
price = 100.0
if score > 0 and price < 200 and price / score > 10 :
    print("buy!")
```

which of the following statements is true?

- A. The output is `buy!`
- B. The code snippet runs, but there is no output
- C. The code snippet has syntax errors
- D. The code snippet causes a divide-by-zero exception

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Which statement is true on the basis of this code snippet?

<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-85

87. Which of the following options checks that `city` is neither Atlanta or Philadelphia?

- A. `if not city == "Atlanta" or not city == "Philadelphia"`
- B. `if not (city == "Atlanta" or city == "Philadelphia")`
- C. `if not (city == "Atlanta" and city == "Philadelphia")`
- D. `if not city == "Atlanta" or city == "Philadelphia"`

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Which of the following options checks that city is neither Atlanta or Philadelphia?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-86

88. Assuming a user enters 30, 55, and 10 as the input, what is the output of the following code snippet?

```
num1 = int(input("Enter a number: "))
num2 = int(input("Enter a number: "))
num3 = int(input("Enter a number: "))
if not (num1 > num2 and num1 > num3) :
    print(num1)
elif not (num2 > num1 and num2 > num3) :
    print(num2)
elif not (num3 > num1 and num3 > num2) :
    print(num3)
```

- A. 55
- B. 10
- C. 0
- D. 30

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	What is the output of the nested if code snippet given three input values?
<b>type</b>	mc
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-1-ch03-87

89. Assuming a user enters 30, 55, and 10 as the input, what is the output of the following code snippet?

```
num1 = int(input("Enter a number: "))
num2 = int(input("Enter a number: "))
num3 = int(input("Enter a number: "))
if num1 > num2 and num1 > num3 :
    print(num1)
elif num2 > num1 and num2 > num3 :
    print(num2)
elif num3 > num1 and num3 > num2 :
    print(num3)
```

- A. 55
- B. 10
- C. 0
- D. 30

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	What is the output of the nested if code snippet given three input values?
<b>type</b>	mc

section  
id

3.7 Boolean Variables and Operators  
testbank-py-1-ch03-88

90. Which of the following conditions is **True** only when the variables **a**, **b**, and **c** contain three different values?

- A. `if a != b and a != c and b != c :`
- B. `if a != b or a != c or b != c :`
- C. `if not (a == b and b == c and a == c) :`
- D. `if a != b != c :`

Section

3.7 Boolean Variables and Operators

Title  
different values?

Which of the following conditions is true only when the variables a, b, and c contain three

type

mc

section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-89

91. Consider the following code segment. It should display a message only if the cost is between 50 and 75 dollars. The message should also be displayed if the cost is exactly 50 dollars or exactly 75 dollars.

```
if _____ :  
    print("The cost is in the desired range")
```

What condition should be placed in the blank to achieve the desired behavior?

- A. `cost > 50`
- B. `cost < 75`
- C. `cost >= 50 and cost <= 75`
- D. `cost >= 50 or cost <= 75`

Section

3.7 Boolean Variable and Operators

Title

Complete an if statement with an appropriate condition

type

mc

section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-90

92. Water is liquid between 0 and 100 degrees Celsius. The following code segment should display a message if the water is **not** liquid. For this question, we will assume that water is liquid if it is exactly 0 degrees or exactly 100 degrees.

```
if _____ :  
    print("The water is not liquid")
```

What condition should be placed in the blank to achieve the desired behavior?

- A. `temp < 0`
- B. `temp > 100`
- C. `temp < 0 and temp > 100`
- D. `temp < 0 or temp > 100`

Section

3.7 Boolean Variables and Operators

Title

Complete an if statement with an appropriate condition

type

mc

section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-91

93. Suppose that **b** is **False** and **x** is 0. Which of the following expressions evaluates to **True**?

- A. `b or x == 1`
- B. `b and x == 0`
- C. `not b and x == 1`
- D. `not b or x == 1`

**Section** 3.7 Boolean Variables and Operators  
**Title** Which of the following expressions evaluates to True?  
**type** mc  
**section** 3.7 Boolean Variables and Operators  
**id** testbank-py-1-ch03-92

94. Suppose that `b` is `False` and `x` is `0`. Which of the following expressions evaluates to `True`?

- A. `not b and x == 1`
- B. `b or x == -1`
- C. `not b or b`
- D. `x == 1 or x == -1`

**Section** 3.7 Boolean Variables and Operators  
**Title** Which of the following expressions evaluates to True?  
**type** mc  
**section** 3.7 Boolean Variables and Operators  
**id** testbank-py-1-ch03-93

95. Which operator has the lowest precedence?

- A. `!=`
- B. `*`
- C. `**`
- D. `and`

**Section** 3.7 Boolean Variables and Operators  
**Title** Which operator has the lowest precedence?  
**type** mc  
**section** 3.7 Boolean Variables and Operators  
**id** testbank-py-2-ch03-122

96. Which of the following values make the expression `not x == y and z > x` true?

- A. `x = 10, y = 10, z = 15`
- B. `x = 10, y = 20, z = 15`
- C. `x = 10, y = 2, z = 5`
- D. `x = 10, y = 20, z = 10`

**Section** 3.7 Boolean Variables and Operators  
**Title** Which of the following values make the expression `not x == y and z > x` true?  
**type** mc  
**from** testbank-py-1-ch03-34  
**section** 3.7 Boolean Variables and Operators  
**id** testbank-py-2-ch03-34

97. What two values does the Boolean (`bool`) data type have in Python?

- A. Yes / No
- B. True / False
- C. 0 / 1
- D. -1 / 1

**Section** 3.7 Boolean Variables and Operators  
**Title** What are the values of a boolean variable?  
**type** mc  
**from** testbank-py-1-ch03-65  
**section** 3.7 Boolean Variables and Operators  
**id** testbank-py-2-ch03-65

98. Given two variables x and y, how do you test whether exactly one of them is zero?

- A. `if x == 0 or y == 0 :`
- B. `if x = 0 or y = 0 :`
- C. `if x == 0 and y != 0 or y == 0 and x != 0 :`
- D. `if x == 0 and y != 0 and y == 0 and x != 0 :`

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Given a code snippet, determine the correct boolean expression to test for a given condition?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-69
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-2-ch03-69

99. Given two variables x and y, how do you test whether at least one of them is zero?

- A. `if x == 0 or y == 0 :`
- B. `if x = 0 or y = 0 :`
- C. `if x == 0 and y != 0 or y == 0 and x != 0 :`
- D. `if x == 0 and y != 0 and y == 0 and x != 0 :`

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Given a code snippet, determine the correct boolean expression to test for a given condition?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-70
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-2-ch03-70

100. The following logical expression will 'short-circuit'...

`quantity > 0 and price/quantity < 10`

- A. When `quantity` is equal to 0
- B. When `quantity` is equal to 5
- C. When `price/quantity` is less than 10
- D. When `price/quantity` is greater than 10

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	What value causes a given expression to short-circuit?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-73
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-2-ch03-73

101. Which of the following expressions represents a legal way of checking whether a value assigned to the `num` variable falls within the range 0 to 150 (inclusive)?

- A. `if num >= 150 and num <= 0 :`
- B. `if num >= 0 and num <= 150 :`
- C. `if num >= 0 or num <= 150 :`
- D. `if num >= 150 or num <= 0 :`

<b>Section</b>	3.7 Boolean Variables and Operators
<b>Title</b>	Which expression checks whether a value falls between 0 and 150 inclusive?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-78
<b>section</b>	3.7 Boolean Variables and Operators
<b>id</b>	testbank-py-2-ch03-78

102. Which of the following expressions represents a legal way of checking whether a value assigned to the `num` variable is either less than 100 or more than 200?



- A. `if num < 100 and num > 200 :`
- B. `if num < 100 and num > 200 :`
- C. `if num < 100 or num > 200 :`
- D. `if num <= 100 or num >= 200 :`

**Section** 3.7 Boolean Variables and Operators  
**Title** Which of the following expressions represents a legal way of checking whether a value assigned to the num variable is either less than 100 or more than 200?  
**type** mc  
**from** testbank-py-1-ch03-79  
**section** 3.7 Boolean Variables and Operators  
**id** testbank-py-2-ch03-79

103. Which of the following conditions is true only when the integer variable middle is between 0 and 10 inclusive?

- A. `middle >= 0 and middle <= 10`
- B. `0 < middle < 10`
- C. `0 <= middle or middle <= 10`
- D. `middle > 0 and middle < 10`

**Section** 3.7 Boolean Variables and Operators  
**Title** Which of the following conditions is true only when the integer variable middle is between 0 and 10?  
**type** mc  
**from** testbank-py-1-ch03-82  
**section** 3.7 Boolean Variables and Operators  
**id** testbank-py-2-ch03-82

104. What string method can be used to determine if the string contained in the variable `text` only consists of numbers?

- A. `text.isalnum()`
- B. `text.isalpha()`
- C. `text.isdigit()`
- D. `text.islower()`

**Section** 3.8 Analyzing Strings  
**Title** What String method checks for all numbers in a variable?  
**type** mc  
**section** 3.8 Analyzing Strings  
**id** testbank-py-1-ch03-100

105. What will be printed by the following code snippet?

```
name = "Ravi Avalon"
counter = name.count("av")
print(counter)
```

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

**Section** 3.8 Analyzing Strings  
**Title** What value is returned when using the String method count?  
**type** mc  
**section** 3.8 Analyzing Strings  
**id** testbank-py-1-ch03-101

106. What will be printed by the following code snippet?

```
name = "Dino the Dinosaur"
counter = name.count("Di")
print(counter)
```

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

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	What value is returned when using the String method count?
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-102

107. Which of the following statements returns the number of blank spaces contained in the string `sentence`?

- A. `sentence.count(" ")`
- B. `" " in sentence`
- C. `sentence.find(" ")`
- D. `count(sentence)`

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	What method is used to identify the number of spaces in a string variable?
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-103

108. Review the code snippet below:

```
name1 = "Betty joe"
name2 = "Betty Jean"
name3 = "Betty Jane"
if name1 < name2 :
    if name1 < name3 :
        print(name1, "is first")
    else :
        print(name3, "is first")
else :
    if name2 < name3 :
        print(name2, "is first")
    else :
        print(name3, "is first")
```

what output is produced?

- A. Betty joe is first
- B. Betty Jean is first
- C. Betty Jane is first
- D. Betty joe is firstBetty Jean is firstBetty Jane is first

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	Review the code snippet below; what output is produced?
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-105

109. How do you test if a filename (given as a string) has an extension of ".png", ".jpg" or ".gif"?

- A. `if filename.endswith(".png" or ".jpg" or ".gif") :`
- B. `if filename.endswith(".png") or filename.endswith(".jpg") or filename.endswith(".gif")`

```
:  
C. if ".png" in filename or ".jpg" in filename or ".gif" in filename :  
D. if filename.contains(".jpg", ".gif", ".png") :
```

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	How do you test if a filename (given as a string) has a valid extension?
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-106

110. What value is displayed by the following code segment?

```
s = "Computer Science"  
x = s.find("TER")  
print(x)
```

- A. -1
- B. 0
- C. 5
- D. 6

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	Search for a substring within a string
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-107

111. What value is displayed by the following code segment?

```
name = "John Smith"  
print(name.startswith("john"))
```

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

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	Determine what is output by the starts with method
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-108

112. Which of the following checks to see if the string variable `sentence` starts with the string `"Dear"`?

- A. `if "Dear" in sentence :`
- B. `if sentence.find("Dear") :`
- C. `if "Dear" not in sentence :`
- D. `if sentence.startswith("Dear") :`

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	Which statement tests if a string contains a substring at the beginning?
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-95

113. What value is printed by the following code snippet?

```
name = "John R. Johnson"
```

```

firstName = "John"
location = name.find(firstName)
print(location)

```

- A. -1
- B. 0
- C. 8
- D. 1

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	What value is returned when using the find substring command?
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-96

114. What value is printed by the following code snippet?

```

name = "John R. Johnson"
firstName = "Joe"
location = name.find(firstName)
print(location)

```

- A. -1
- B. 0
- C. 8
- D. 1

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	What value is returned when using the find substring command?
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-97

115. What string method can be used to determine if the string contained in the variable `text` only consists of letters?

- A. `text.isalnum()`
- B. `text.isalpha()`
- C. `text.isdigit()`
- D. `text.islower()`

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	What String method checks for all characters in a variable?
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-98

116. What string method can be used to determine if all characters within a string are lowercase?

- A. `text.isalnum()`
- B. `text.isalpha()`
- C. `text.isdigit()`
- D. `text.islower()`

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	What String method checks for all lowercase characters?
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-1-ch03-99

117. Review the code snippet below:

```
sentence = input("Enter some text: ")
firstCh = sentence[0]
```

Which of the following statements correctly determines if the first letter of the string contained in `sentence` is an uppercase letter?

- A. `if firstCh.isupper() :`
- B. `if not (firstCh.isupper()) :`
- C. `if firstCh.isspace() :`
- D. `if not (firstCh.isspace()) :`

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	Which of the following statements correctly determines if the first letter of the string is an uppercase letter?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-104
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-2-ch03-104

118. What value is displayed when the following code segment is executed?

```
s = "Jonathan"
print(s.endswith("\n"))
```

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

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	Trace code involving the <code>endswith</code> method
<b>type</b>	mc
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-2-ch03-123

119. Which of the following checks to see if there is a comma anywhere in the string variable `name`?

- A. `if "," in name :`
- B. `if name.contains(",") :`
- C. `if "," not in name :`
- D. `if name.startswith(",") :`

<b>Section</b>	3.8 Analyzing Strings
<b>Title</b>	Which statement tests if a string contains a substring?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-94
<b>section</b>	3.8 Analyzing Strings
<b>id</b>	testbank-py-2-ch03-94

120. Which of the following statements can be used to validate whether the value a user entered for a grade is in the range 0 to 100, including both 0 and 100?

- A. `if grade > 0 and grade < 100 :`
- B. `if grade >= 0 and grade <= 100 :`
- C. `if grade <= 0 and grade >= 100 :`
- D. `if grade <=0 or grade >=100 :`

<b>Section</b>	3.9 Application: Input Validation
<b>Title</b>	Which statement validates the user input is between 0 and 100?
<b>type</b>	mc

<b>section</b>	3.9 Application: Input Validation
<b>id</b>	testbank-py-1-ch03-109

121. Which of the following statements is the best choice to validate user input when entering a marital status as a single letter?

- A.  

```
if maritalStatus == "s" or maritalStatus == "m" :
```
- B.  

```
if maritalStatus == "S" or maritalStatus == "M" :
```
- C.  

```
if (maritalStatus == "s" or maritalStatus == "m" or  
    maritalStatus == "S" or maritalStatus == "M") :
```
- D.  

```
if maritalStatus == "s" or "S" or "m" or "M" :
```

<b>Section</b>	3.9 Application: Input Validation
<b>Title</b>	Which statements can be used to validate the user entered a valid marital status?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-110
<b>section</b>	3.9 Application: Input Validation
<b>id</b>	testbank-py-2-ch03-110

122. Review the code snippet below:

```
maritalStatus = input("Enter your marital status (s for single, m for married): ")  
maritalStatus = maritalStatus.upper()
```

Which of the following statements can be used to validate whether the user entered a valid marital status?

- A.  

```
if maritalStatus == "S" or maritalStatus == "M" :
```
- B.  

```
if maritalStatus == "s" or maritalStatus == "m" :
```
- C.  

```
if (maritalStatus == "s" or maritalStatus == "m") and  
    (maritalStatus == "S" or maritalStatus == "M") :
```
- D.  

```
if maritalStatus == "s" or "S" or "m" or "M" :
```

<b>Section</b>	3.9 Application: Input Validation
<b>Title</b>	Which statements can be used to validate the user entered a valid marital status?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-111
<b>section</b>	3.9 Application: Input Validation
<b>id</b>	testbank-py-2-ch03-111

123. Review the code snippet below:

```
month = int(input("Enter your two digit birth month: "))
```

Which of the following statements checks that the user entered a valid month?

- A. 

```
if month >= 1 or month <= 12 :
```
- B. 

```
if month >= 1 and month <= 12
```

- C. `if month > 1 or month < 12 :`
- D. `if month > 1 and month < 12 :`

<b>Section</b>	3.9 Application: Input Validation
<b>Title</b>	Which statements can be used to validate the user entered a valid birth month?
<b>type</b>	mc
<b>from</b>	testbank-py-1-ch03-112
<b>section</b>	3.9 Application: Input Validation
<b>id</b>	testbank-py-2-ch03-112

124. Which statement will successfully import the pyplot submodule?

- A. `from math import pyplot`
- B. `from matplotlib import pyplot`
- C. `import pyplot`
- D. `import * from pyplot`

<b>Section</b>	3.9 Application: Input Validation
<b>Title</b>	Which statement will successfully import the pyplot submodule?
<b>type</b>	mc
<b>section</b>	3.9 Application: Input Validation
<b>id</b>	testbank-py-2-ch03-124

125. Which statement adds a bar to a pyplot graph after pyplot has been imported by the following statement?

```
from matplotlib import pyplot
```

- A. `bar(4, 44.5)`
- B. `pyplot.bar(4, 44.5)`
- C. `pyplot(4, 44.5)`
- D. `bar.pyplot(4, 44.5)`

<b>Section</b>	3.9 Application: Input Validation
<b>Title</b>	Which statement adds a bar to a pyplot graph?
<b>type</b>	mc
<b>section</b>	3.9 Application: Input Validation
<b>id</b>	testbank-py-2-ch03-125