

1. What statement is used to implement a decision?

1. while
2. if
3. for
4. import

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?

1. A condition and a body
2. A check and an increment
3. An increment and a body
4. An increment and a return value

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?

1. The `if` statement can have only one condition that evaluates to an integer value.
2. The `if` block is optional.
3. The `else` block is optional.
4. The `if` and `else` blocks can be aligned to any indentation level.

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?

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

```
    size = "small"
else (x < 20)
    size = "medium"
```

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

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

Title

What is the correct syntax for an if statement?

type

mc

Section

3.1 The if statement

id

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)
```

1. Nothing, the program runs as intended
2. The statement after the `if` statement must be indented
3. The statement after the `if` statement and the statement after the `else` statement must be indented
4. No colon is needed after the `else` statement

Title

What is the correct syntax for an if statement?

type

mc

Section

3.1 The if statement

id

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)
```

1. x: 0
2. x: 303
3. x: 300
4. There is no output due to a syntax error.

Title

What is the output of an if/else statement given sample input value  
type

mc

Section

3.1 The if statement

id

testbank-py-1-ch03-06

7. 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)
```

1. Syntax error: missing colon after if statement
2. Logical error: use of an uninitialized variable
3. Syntax error: missing an else statement
4. Syntax error: incorrect indentation

Title

Find the error in a code snippet containing an if statement

type

mc

Section

3.1 The if statement

id

testbank-py-2-ch03-07

from

testbank-py-1-ch03-07

8. 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)
```

1. x: 0
2. x: 95
3. x: 300
4. There is no output due to a syntax error

Title

What is the output of an if/else statement for a given input value?

type

mc

Section

3.1 The if statement

id

testbank-py-2-ch03-08

from

testbank-py-1-ch03-08

9. 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)
```

1. Nothing, the program will run but not print any results
2. Total discounted price is: 64.0
3. Total discounted price is: 0.0
4. Total discounted price is: 16.0

Title

What is the output of a compound statement?

type

mc

Section

3.1 The if statement

id

testbank-py-2-ch03-09

from

testbank-py-1-ch03-09

10. 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)
```

1. Nothing, the code contains a syntax error
2. Your cost is: 0
3. Your cost is: 80
4. Your cost is: 100

Title

What is the output of a conditional expression?

type

mc

Section

3.1 The ifstatement

id

testbank-py-1-ch03-10

11. 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:

1. -1
2. 0
3. 4
4. 5

Title

Trace an if statement

type

mc

Section

3.1 The if Statement

id

testbank-py-1-ch03-11

12. 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`?

1. 1
2. 2
3. 4

4. 5

Title

Trace an if statement

type

mc

Section

3.1 The if Statement

id

testbank-py-1-ch03-12

13. 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?

1. 1
2. 2
3. 3
4. 4

Title

Trace an if/else statement

type

mc

Section

3.1 The if Statement

id

testbank-py-1-ch03-13

14. What is the error in this statement?

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

1. Equality is evaluated using two equal signs (==), not one.
2. The print function should not be indented
3. There must be an else statement
4. Nothing, if count equals max, it would print "You win"

Title

What is the error in this statement?

type

mc  
Section  
3.2 Relational Operators  
id  
testbank-py-1-ch03-14

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

1. `count >= 10`
2. `count < 9`
3. `count <= 10`
4. `count <= 9`

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

16. 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")
```

1. **Count is an even number**
2. **Count is an odd number**
3. **Nothing, there is a syntax error**
4. Nothing, the program runs but does not print any output

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

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")
```

1. Count is an even number
2. Count is an odd number
3. Nothing, there is a syntax error
4. Nothing, the program runs but does not print any output

Title

What is the output of a code snippet with relational operators?

type

mc

Section

3.2 Relational Operators

id

testbank-py-1-ch03-17

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

1. Lexicographic
2. Arithmetic
3. Inequality
4. Relational

Title

What type of operator is `!<` operator?

type

mc

Section

3.2 Relational Operators

id

testbank-py-1-ch03-18

19. The operator `>=` stands for

1. greater than
2. greater than or equal to
3. not equal to
4. this is not a valid Python operator

Title

Which relational operator is this?

type

mc

Section

3.2 Relational Operators

id

testbank-py-1-ch03-19



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

1. `if userInput = "Y" :`
2. `if userInput = "Y" :`
3. `if userInput == "Y" :`
4. `if userInput == "y" :`

Title

Which statement correctly tests for a user input value?

type

mc

Section

3.2 Relational Operators

id

testbank-py-1-ch03-20

21. 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")
```

1. There is no output due to a syntax error
2. The number is big
3. The number is small
4. The program runs successfully but does not print any output

Title

What is the output of a given code snippet and given input value?

type

mc

Section

3.2 Relational Operators

id

testbank-py-1-ch03-21

22. 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)
```

1. 34
2. 33
3. 35
4. 36

Title

What is the output of a given code snippet and given input value?

type

mc

Section

3.2 Relational Operators

id

testbank-py-1-ch03-22

23. 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)
```

1. 105
2. 45
3. 43
4. 49

Title

What is the output of a code statement containing multiple if statements and a given input value?

type

mc

Section

3.2 Relational Operators

id

testbank-py-1-ch03-23

24. 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?

1. discount = 0  
if price >= 100 :  
    discount = 0.10 \* price
2. discount = 0.10 \* price  
if price <= 100 :  
    discount = 0
3. discount = 0  
if price >= 100 :  
    discount = price / 10

```
4. discount = 10
   if price >= 100 :
       discount = 0.1 * price
   else :
       discount = 0
```

Title

Which statement DOES NOT correctly compute the discount?

type

mc

Section

3.2 Relational Operators

id

testbank-py-2-ch03-24

from

testbank-py-1-ch03-24

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

1. `num1 + 1 < num2`
2. `num1 + 1 > num2`
3. `num1 + num2 != 7`
4. `num1 - num2 <= 0`

Title

Which of the following conditions evaluates to True given two variables?

type

mc

Section

3.2 Relational Operators

id

testbank-py-2-ch03-25

from

testbank-py-1-ch03-25

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

1. Semantic
2. Alphabetic
3. Syntactic
4. Lexicographic

Title

Which ordering is used to compare strings?

type

mc

Section

3.2 Relational Operators

id

testbank-py-1-ch03-26

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

1. `if count = 0 :`
2. `if count < 0 :`
3. `if count =< 0 :`
4. `if count == 0 :`

Title

Which condition will cause the statement block of the if statement to execute?

type

mc

Section

3.2 Relational Operators

id

testbank-py-2-ch03-27

from

testbank-py-1-ch03-27

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

1. `if 4 // 3 == 1 :`
2. `if sqrt(2) * sqrt(2) == 2.0 :`
3. `if "10" == 5 :`
4. `if 4 <= 4 :`

Title

If statements and the limited precision of floating-point numbers

type

mc

Section

3.2 Relational Operators

id

testbank-py-1-ch03-28

29. Consider the following code segment:

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

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

1. `if s1 == s2 :`
2. `if s1 = s2 :`
3. `if s1 < s2 :`
4. `if s1 >= s2 :`

Title

Comparing strings with relational operators

type

mc

Section  
3.2 Relational Operators  
id  
testbank-py-1-ch03-29

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

```
name1 = "Heather"  
name2 = "hanna"
```

1. `name1 == name2`
2. `name1 > name2`
3. `name1 < name2`
4. Relational operators cannot be used to compare strings

Title  
Which statement evaluates to true when comparing two strings?  
type  
mc  
Section  
3.2  
id  
testbank-py-2-ch03-30  
from  
testbank-py-1-ch03-30

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

1. amy, Ann, hanna, Heather, Jim, joe, john, Leo
2. Ann, Heather, Jim, Leo, amy, hanna, joe, john
3. amy, hanna, joe, john, Ann, Heather, Jim, Leo
4. Leo, john, joe, Jim, Heather, hanna, Ann, amy

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

32. What is the definition of a nested statement?

1. A decision statement that is contained inside the statement block of another decision statement
2. A compound statement that consists of a header and a statement block
3. A decision statement that immediately follows another decision statement at the same indentation level
4. A statement that is used to validate user input

Title

What is the definition of a nested statement?

type

mc

Section

3.3 Nested Branches

id

testbank-py-2-ch03-32

from

testbank-py-1-ch03-32

33. 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)
```

1. 0
2. 10
3. 20
4. 30

Title

What is the output of the nested if code snippet given three input values?

type

mc

Section

3.3 Nested Branches

id

testbank-py-1-ch03-33

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

1. `x = 10, y = 10, z = 15`
2. `x = 10, y = 20, z = 15`
3. `x = 10, y = 2, z = 5`
4. `x = 10, y = 20, z = 10`

Title

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

type

mc  
Section  
3.7 Boolean Variables and Operators  
id  
testbank-py-2-ch03-34  
from  
testbank-py-1-ch03-34

35. 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)
```

1. 95
2. 100
3. 105
4. 110

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

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

1. Compiling
2. Prototyping
3. Debugging
4. Tracing

Title  
Which refers to the technique of simulating program execution on a sheet of paper?  
type  
mc  
Section  
3.3 Nested Branches  
id

37. 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"
```

1.

price	status
0	"inexpensive"
25	"cheap"

2.

price	status
0	"inexpensive"
25	"reasonable"

3.

price	status
0	"inexpensive"
25	"reasonable"
	"costly"

4.

price	status
0	"inexpensive"
25	"costly"

Title

Which hand-trace table is valid for this snippet?

type

mc

Section

3.3 Nested Branches

id

testbank-py-2-ch03-37



from  
testbank-py-1-ch03-37

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

1. `if a == b :`  
    `print(a)`

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

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

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

Title

Which of the following is an example of a nested `if` statement?

type

mc

Section

3.3 Nested Branches

id

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?

1. `X`
2. `Y`
3. `X` followed by `Y` on the next line
4. Nothing

Title

Trace a nested `if` statement

type

mc

Section

3.3 Nested Branches  
id  
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?

1. x
2. y
3. x followed by y on the next line
4. Nothing

Title  
Trace a nested if statement  
type  
mc  
Section  
3.3 Nested Branches  
id  
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?

1. x
2. y
3. x followed by y on the next line
4. Nothing

Title  
Trace a nested if statement  
type  
mc  
Section  
3.3 Nested Branches  
id  
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:

1. W
2. W followed by Y on the next line
3. X followed by Y on the next line
4. W followed by X on the next line, followed by Y on the next line

Title

Trace nested if/else statements

type

mc

Section

3.3 Nested Branches

id

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:

1. W
2. X
3. X followed by Y on the next line
4. X followed by Z on the next line

Title

Trace nested if/else statements

type

mc

Section

3.3 Nested Branches

id

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:

1. W
2. X
3. X followed by Y on the next line
4. X followed by Z on the next line

Title

Trace nested if/else statements

type

mc

Section

3.3 Nested Branches

id

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")
```

1. Python will display an error indicating that == should be replaced with =
2. Python will display an error indicating that an if statement cannot reside inside the body of an else
3. Python will display an error indicating that there is a problem with the indentation
4. No error will be displayed

Title

Find the error in an if statement

type

mc

Section

3.3 Nested Branches

id  
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")
```

1. Python will display an error indicating that = is invalid syntax
2. Python will display an error indicating that an if statement cannot reside inside the body of an else
3. Python will display an error indicating that there is a problem with the indentation
4. No error will be displayed

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

47. 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)
```

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

Title  
What is the output of an if statement with multiple alternatives?

type  
mc  
Section  
3.4 Multiple Alternatives  
id  
testbank-py-1-ch03-47

48. 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)
```

1. Everyone will get an "E"
2. Anyone with a grade higher than 60 will receive a "D"
3. Nothing is wrong, students will get the correct grade
4. The code block will not compile

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

49. 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)
```

1. Change the `if` statements to `elif` statements (except the first one)
2. Change the `if` statements to `else` statements (except the first one)

3. Reverse the order of the `if` statements
4. Change the last statement to `if` instead of `else`

Title

How can you correct a code snippet that has multiple alternatives?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-2-ch03-49

from

testbank-py-1-ch03-49

50. 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="")
```

1. 1
2. 2
3. 3
4. 123

Title

What is the output of the following code snippet?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-1-ch03-50

51. 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?

1. `if number > 30 : ...`

2. `else if number > 20 : ...`
3. `else if number > 10 : ...`
4. `else : ...`

Title

Which statement is executed when the user enters 40 for the input value?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-1-ch03-51

52. 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?

1. Too high
2. High
3. Intermediate
4. Too small

Title

What is output of the if/elif/else snippet with this input?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-1-ch03-52



53. 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?

1. num1 = 20 num2 = 12 num3 = 20 num4 = 0 num5 = 20
2. num1 = 20 num2 = 12 num3 = 12 num4 = 20 num5 = 0
3. num1 = 20 num2 = 12 num3 = 12 num4 = 0 num5 = 20
4. num1 = 20 num2 = 12 num3 = 20 num4 = 20 num5 = 0

Title

What is output of the if/else and if/elif snippet with two input values?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-1-ch03-53

54. 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
```

1. 42
2. 52
3. 84
4. 64

Title

What is value of a variable after (if/elif/else) snippet is executed?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-1-ch03-54

55. 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?

1. Child  
Young adult  
Old

2. Young adult  
Old

3. Impressively old

4. Child  
Young adult  
Old  
Impressive old

Title

What is output of the code snippet with this input?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-1-ch03-55

56. 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?

1. ChildYoung adultOldImpressively old
2. Young adultOldImpressively old
3. OldImpressively old
4. Impressively old

Title

What is output of the code snippet with this input?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-1-ch03-56

57. 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?

1. Child
2. ChildYoung Adult
3. ChildYoung AdultOld
4. ChildYoung adultOldImpressively old

Title

What is output of the code snippet with this input?

type

mc

Section

3.4 Multiple Alternatives

id  
testbank-py-2-ch03-57  
from  
testbank-py-1-ch03-57

58. 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"
```

1. The greater than or equal signs need to be replaced with equal signs
2. All instances of if, except the first, need to be replaced with elif
3. All instances of if, except the first, need to be replaced with else
4. There is nothing wrong with the code segment (it works as intended)

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

59. 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")
```

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

Title

What is wrong with this multiple alternatives statement?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-1-ch03-59

60. 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
```

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

Title

What is wrong with this multiple alternatives statement?

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-1-ch03-60

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

1. elements for tasks
2. elements for input/output
3. elements for pseudocode
4. elements for decisions

Title

What are the parts of a flowchart?

type

mc

Section

3.5 Problem Solving: Flowcharts

id

testbank-py-1-ch03-61

62. The flowchart shows the order in which steps should be executed, and the diamond-shaped boxes indicate:
1. input
  2. algorithms
  3. tasks
  4. decision statements

Title

What are the parts of a flowchart?

type

mc

Section

3.5 Problem Solving: Flowcharts

id

testbank-py-1-ch03-62

63. When testing code for correctness, it always makes sense to
1. Aim for complete coverage of all decision points
  2. Identify boundary cases and test them
  3. Check all cases using hand-tracing
  4. Assume invalid input will never occur

Title

What is reasonable for checking/testing code?

type

mc

Section

3.6 Problem Solving: Test Cases

id

testbank-py-1-ch03-63

64. 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?

1. 2
2. 3
3. 4
4. 5

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

65. What two values does the Boolean (`bool`) data type have in Python?
1. Yes / No
  2. True / False
  3. 0 / 1
  4. -1 / 1

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

66. Which operators listed below are considered boolean operators:
1. `< / >`
  2. `and / or`
  3. `== / !=`
  4. `<= / >=`

Title  
What are boolean operators?  
type  
mc  
Section  
3.7 Boolean Variables and Operators  
id  
testbank-py-1-ch03-66

67. Consider the following code snippet:

```
temp = 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:

1. Nothing is printed
2. Liquid
3. Not Liquid

#### 4. LiquidNotLiquid

Title

Given a code snippet and an input value, what output is produced?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-67

68. Which of the following variables is used to store a condition that can be either

True or False?

1. Logical
2. Boolean
3. Algebraic
4. Conditional

Title

What kind of variable is used to store a true/false condition?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-68

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

1. `if x == 0 or y == 0 :`
2. `if x = 0 or y = 0 :`
3. `if x == 0 and y != 0 or y == 0 and x != 0 :`
4. `if x == 0 and y != 0 and y == 0 and x != 0 :`

Title

Given a code snippet, determine the correct boolean expression to test for a given condition?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-2-ch03-69

from

testbank-py-1-ch03-69

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

1. `if x == 0 or y == 0 :`
2. `if x = 0 or y = 0 :`



3. `if x == 0 and y != 0 or y == 0 and x != 0 :`
4. `if x == 0 and y != 0 and y == 0 and x != 0 :`

Title

Given a code snippet, determine the correct boolean expression to test for a given condition?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-2-ch03-70

from

testbank-py-1-ch03-70

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

`32 <= temp <= 100`

1. `if temp <= 32 and temp <= 100`
2. `if temp <= 32 or temp <= 100`
3. `if temp >= 32 and temp <= 100`
4. `if temp >= 32 or temp <= 100`

Title

Rewrite an algebraic expression into the equivalent Python expression?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-71

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

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

1. `temp = 0`
2. `temp = 32`
3. `temp = 100`
4. `temp = 75`

Title

What value causes a given expression to short-circuit?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-72

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

```
quantity > 0 and price/quantity < 10
```

1. When quantity is equal to 0
2. When quantity is equal to 5
3. When price/quantity is less than 10
4. When price/quantity is greater than 10

Title

What value causes a given expression to short-circuit?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-2-ch03-73

from

testbank-py-1-ch03-73

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

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

1. if state != "PA" and state != "OH"
2. if state != "PA" or state != "OH"
3. if state == "PA" and state == "OH"
4. if state == "PA" or state == "OH"

Title

Using De Morgan's law, identify an equivalent expression?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-74

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

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

1. if state != "PA" and state != "OH"
2. if state != "PA" or state != "OH"
3. if state == "PA" and state == "OH"
4. if state == "PA" or state == "OH"

Title

Using De Morgan's law, identify an equivalent expression?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-75

76. Consider the following code snippet:

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

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

1. `if attendance == "true" :`
2. `if attendance :`
3. `if failed :`
4. `if attendance == failed :`

Title

What if statements include a condition that evaluates to true?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-76

77. 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?

1. Teen
2. Young at heart
3. Child
4. Adult

Title

What is output of the code snippet with this input?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-77

78. 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)?

1. `if num >= 150 and num <= 0 :`
2. `if num >= 0 and num <= 150 :`
3. `if num >= 0 or num <= 150 :`
4. `if num >= 150 or num <= 0 :`

Title

Which expression checks whether a value falls between 0 and 150 inclusive?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-2-ch03-78

from

testbank-py-1-ch03-78

79. 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?

1. `if num < 100 and num > 200 :`
2. `if num < 100 and num > 200 :`
3. `if num < 100 or num > 200 :`
4. `if num <= 100 or num >= 200 :`

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

Section

3.7 Boolean Variables and Operators

id

testbank-py-2-ch03-79

from

testbank-py-1-ch03-79

80. 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?

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

Title

What is the output of a code snippet given a specific input value?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-80

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

1. or
2. and
3. not
4. equal

Title

Which operator is used to invert a conditional statement?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-81

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

1. `middle >= 0 and middle <= 10`
2. `0 < middle < 10`
3. `0 <= middle or middle <= 10`
4. `middle > 0 and middle < 10`

Title

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

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-2-ch03-82

from

testbank-py-1-ch03-82

83. 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?

1. `mayVote = age > 18 or not isFelon`
2. `mayVote = not ( age >= 18 and isFelon)`
3. `mayVote = age >= 18 and not isFelon`
4. `mayVote = not ( age >= 18 or isFelon)`

Title

Which of the following statements assigns the Boolean variable correctly?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-83

84. 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?

1. 45
2. 55

3. 65
4. 50

Title

What is the output of a Boolean not code snippet?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-84

85. 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?

1. The output is `buy!`
2. The code snippet runs, but there is no output
3. The code snippet has syntax errors
4. The code snippet causes a divide-by-zero exception

Title

Which statement is true on the basis of this code snippet?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-85

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

1. `if not city == "Atlanta" or not city == "Philadelphia"`
2. `if not (city == "Atlanta" or city == "Philadelphia")`
3. `if not (city == "Atlanta" and city == "Philadelphia")`
4. `if not city == "Atlanta" or city == "Philadelphia"`

Title

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

type

mc

Section

3.7 Boolean Variables and Operators

id

87. 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)
```

1. 55
2. 10
3. 0
4. 30

Title

What is the output of the nested if code snippet given three input values?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-87

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 num1 > num2 and num1 > num3 :
    print(num1)
elif num2 > num1 and num2 > num3 :
    print(num2)
elif num3 > num1 and num3 > num2 :
    print(num3)
```

1. 55
2. 10
3. 0
4. 30

Title

What is the output of the nested if code snippet given three input values?

type

mc

Section

3.7 Boolean Variables and Operators



id  
testbank-py-1-ch03-88

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

1. `if a != b and a != c and b != c :`
2. `if a != b or a != c or b != c :`
3. `if not (a == b and b == c and a == c) :`
4. `if a != b != c :`

Title

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

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-89

90. 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?

1. `cost > 50`
2. `cost < 75`
3. `cost >= 50 and cost <= 75`
4. `cost >= 50 or cost <= 75`

Title

Complete an if statement with an appropriate condition

type

mc

Section

3.7 Boolean Variable and Operators

id

testbank-py-1-ch03-90

91. 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?

1. `temp < 0`
2. `temp > 100`
3. `temp < 0 and temp > 100`
4. `temp < 0 or temp > 100`

Title

Complete an if statement with an appropriate condition

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-91

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

1. `b or x == 1`
2. `b and x == 0`
3. `not b and x == 1`
4. `not b or x == 1`

Title

Which of the following expressions evaluates to `True`?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-92

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

1. `not b and x == 1`
2. `b or x == -1`
3. `not b or b`
4. `x == 1 or x == -1`

Title

Which of the following expressions evaluates to `True`?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-1-ch03-93

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

1. `if "," in name :`
2. `if name.contains(",") :`
3. `if "," not in name :`
4. `if name.startswith(",") :`

Title

Which statement tests if a string contains a substring?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-2-ch03-94

from

testbank-py-1-ch03-94

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

1. `if "Dear" in sentence :`
2. `if sentence.find("Dear") :`
3. `if "Dear" not in sentence :`
4. `if sentence.startswith("Dear") :`

Title

Which statement tests if a string contains a substring at the beginning?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-95

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

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

1. -1
2. 0
3. 8
4. 1

Title

What value is returned when using the find substring command?

type

mc

Section  
3.8 Analyzing Strings  
id  
testbank-py-1-ch03-96

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

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

1. -1
2. 0
3. 8
4. 1

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

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

1. `text.isalnum()`
2. `text.isalpha()`
3. `text.isdigit()`
4. `text.islower()`

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

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

1. `text.isalnum()`
2. `text.isalpha()`
3. `text.isdigit()`
4. `text.islower()`

Title

What String method checks for all lowercase characters?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-99

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

1. `text.isalnum()`

2. `text.isalpha()`

3. `text.isdigit()`

4. `text.islower()`

Title

What String method checks for all numbers in a variable?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-100

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

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

1. 0

2. 1

3. 2

4. -1

Title

What value is returned when using the String method `count`?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-101

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

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

1. 0
2. 1
3. 2
4. -1

Title

What value is returned when using the String method count?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-102

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

1. `sentence.count(" ")`
2. `" " in sentence`
3. `sentence.find(" ")`
4. `count(sentence)`

Title

What method is used to identify the number of spaces in a string variable?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-103

104. 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?

1. `if firstCh.isupper() :`
2. `if not (firstCh.isupper()) :`
3. `if firstCh.isspace() :`
4. `if not (firstCh.isspace()) :`

Title

Which of the following statements correctly determines if the first letter of the string is an uppercase letter?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-2-ch03-104

from

testbank-py-1-ch03-104

105. 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?

1. Betty joe is first
2. Betty Jean is first
3. Betty Jane is first
4. Betty joe is firstBetty Jean is firstBetty Jane is first

Title

Review the code snippet below; what output is produced?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-105

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

1. if filename.endswith(".png" or ".jpg" or ".gif") :
2. if filename.endswith(".png") or filename.endswith(".jpg") or filename.endswith(".gif") :
3. if ".png" in filename or ".jpg" in filename or ".gif" in filename :
4. if filename.contains(".jpg", ".gif", ".png") :

Title

How do you test if a filename (given as a string) has a valid extension?

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-106

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

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

1. -1

2. 0

3. 5

4. 6

Title

Search for a substring within a string

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-107

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

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

1. -1

2. 0

3. False

4. True

Title

Determine what is output by the starts with method

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-1-ch03-108

109. 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?



1. `if grade > 0 and grade < 100 :`
2. `if grade >= 0 and grade <= 100 :`
3. `if grade <= 0 and grade >= 100 :`
4. `if grade <=0 or grade >=100 :`

Title

Which statement validates the user input is between 0 and 100?

type

mc

Section

3.9 Application: Input Validation

id

testbank-py-1-ch03-109

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

1. `if maritalStatus == "s" or maritalStatus == "m" :`
2. `if maritalStatus == "S" or maritalStatus == "M" :`
3. `if (maritalStatus == "s" or maritalStatus == "m" or maritalStatus == "S" or maritalStatus == "M") :`
4. `if maritalStatus == "s" or "S" or "m" or "M" :`

Title

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

type

mc

Section

3.9 Application: Input Validation

id

testbank-py-2-ch03-110

from

testbank-py-1-ch03-110

111. 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?

1 - `if maritalStatus == "S" or maritalStatus == "M" :`

2- `if maritalStatus == "s" or maritalStatus == "m" :`

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

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

Title

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

type

mc

Section

3.9 Application: Input Validation

id

testbank-py-2-ch03-111

from

testbank-py-1-ch03-111

112. 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?

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

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

3. `if month > 1 or month < 12 :`

4. `if month > 1 and month < 12 :`

Title

Which statements can be used to validate the user entered a valid birth month?

type  
mc  
Section  
3.9 Application: Input Validation  
id  
testbank-py-2-ch03-112  
from  
testbank-py-1-ch03-112

113. 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?

1. msg = MIMEApplication()
2. msg = MIMEImage()
3. msg = MIMEMultipart()
4. msg = MIMEText()

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

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

1. The application
2. The attachment
3. The footer
4. The header

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

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

1. MIMEApplication
2. MIMEDocument
3. MIMEImage
4. MIMEPDF

Title

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

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-2-ch03-115

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

1. email
2. login
3. mimelib
4. smtpplib

Title

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

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-2-ch03-116

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

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

Title

Which statement about if statements is not correct?

type

mc

Section

3.1 The if Statement

id

testbank-py-2-ch03-117

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

1. =
2. <
3. <=
4. !=

Title

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

type

mc

Section

3.2 Relational Operators

id

testbank-py-2-ch03-118

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

```
floor - 1 < 13
```

1. 13 < floor - 1
2. 13 >= floor - 1
3. floor < 12
4. floor - 1 <= 12

Title

Which is expression is equivalent to the expression shown below?

type

mc

Section

3.2 Relational Operators

id

testbank-py-2-ch03-119

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

1. if
2. if-elif
3. if-else
4. if-elif-else

Title

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

type

mc

Section

3.4 Multiple Alternatives

id

testbank-py-2-ch03-120

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

1. knotted logic
2. spaghetti code
3. twisted conditions
4. zigzag functions

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

122. Which operator has the lowest precedence?

1. !=
2. \*
3. \*\*
4. and

Title

Which operator has the lowest precedence?

type

mc

Section

3.7 Boolean Variables and Operators

id

testbank-py-2-ch03-122

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

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

1. -1
2. 0
3. False
4. True

Title

Trace code involving the endswith method

type

mc

Section

3.8 Analyzing Strings

id

testbank-py-2-ch03-123

124. Which statement will successfully import the pyplot submodule?

1. `from math import pyplot`
2. `from matplotlib import pyplot`
3. `import pyplot`
4. `import * from pyplot`

Title

Which statement will successfully import the pyplot submodule?

type

mc

Section

3.9 Application: Input Validation

id

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
```

1. `bar(4, 44.5)`
2. `pyplot.bar(4, 44.5)`
3. `pyplot(4, 44.5)`
4. `bar.pyplot(4, 44.5)`

Title

Which statement adds a bar to a pyplot graph?

type

mc

Section

3.9 Application: Input Validation

id

testbank-py-2-ch03-125