Top of Form

Question 1:

Guess the output of:

1. val = 10
3. if val < 10:
4. print(1)
5. elif val < 20:
6. print(2)
7. else:
8. print(3)

1

2

3

Top of Form

Question 2:

Guess the output of:

1. val = 20
3. if val < 10:
4. print(1)
5. elif val < 20:
6. print(2)
7. else:
8. print(3)

1

2

3

Top of Form

Question 3:

Guess the output of:

1. val = 5
3. if val < 10:
4. print(1)
5. if val < 20:
6. print(2)
7. else:
8. print(3)

1

2

3

1

2

1

3

Top of Form

Guess the output of this program:

1. val = -5
3. if val:
4. print(1)

1

Empty Output

Bottom of Form

Bottom of Form

Top of Form

Guess the output of this program:

1. salary = 500
3. if salary < 1000:
4. salary += 5000
5. if 1000 < salary <= 20000:
6. salary += 6000
7. else:
8. salary -= 10000
10. print(salary)

11000

11500

-9500

Top of Form

Question 6:

Guess the output of this program:

1. x = 0
3. if True:
4. x += 1
5. if True:
6. x += 2
8. if False:
9. x += 3
10. else:
11. x += 4
12. if True:
13. x += 5
15. print(x)

0

12

15

3

Top of Form

Question 7:

Guess the output of this program:

1. if True:
2. print('Hi Customer')
3. print('''Too many people spend money they earned..to buy things
4. they don't want..to impress people that they don't like. --Will Rogers''')

Error

* 1. Hi Customer
  2. Too many people spend money they earned..to buy things
  3. they don't want..to impress people that they don't like. --Will Rogers
  4. Hi Customer
  5. Too many people spend money they earned..to buy things
  6. they don't want..to impress people that they don't like. --Will Rogers

Top of Form

Question 8:

This program is for even/odd printing. From a correctness perspective:

1. num = int(input())
3. if num % 2 == 1:
4. print('Odd number')
5. else:
6. print('Even number')

It is completely valid for integer values

It may not work well for negative values

Unlike C or C++, Python's modulo operator (%) always return a number having the same sign as the denominator (divisor). Note for future. There are some small differences between arithmetic on Decimal objects and arithmetic on integers and floats. When the remainder operator % is applied to Decimal objects, the sign of the result is the sign of the dividend rather than the sign of the divisor: from decimal import \* print(-5 % 2, 5 % -2, Decimal(-5) % 2)

Top of Form

Question 9:

background (good to know): *A leap year contains one additional day that is added to keep the calendar year synchronized with the astronomical year. A year that is divisible by 4 is known as a leap year. However, years divisible by 100 are not leap years while those divisible by 400 are.*

For input 1000, the output is

1. year = int(input())
2. if year % 4 == 0 and year % 100 != 0 or year % 400 == 0:
3. print("A leap year")
4. else:
5. print("Not a leap year")

A leap year

* 1. Not a leap year

Bottom of Form

Top of Form

Question 10:

which code is equivalent to this condition:

if pos <= 0 or pos > 1:

if pos == 1:

if pos == 2:

if pos != 1:

if pos != 2:

None

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form

Bottom of Form