

Python Pre-Test

Firia Labs - Programming with CodeX

* Indicates required question

1. First and last name * 1 point

2. What grade are you in? * 1 point

Mark only one oval.

Grade 9

Grade 10

Grade 11

Grade 12

Other:

3. What state do you live in? * 1 point

Python questions

4. What does the "import" command do? *

1 point

```
from codex import *  
from time import sleep  
import random
```

Mark only one oval.

- Allows you to use the CodeX, time and random numbers
- Moves the code to a different programming environmnet
- Provides access to pre-built functions and methods in coding libraries
- Enables object-oriented programming

5. Which of these tips is NOT something that makes your code more readable? * 1 point

Mark only one oval.

- Variables names that are just one or two letters, like 'ab' or 'xy'
- Blank lines in the code
- Comments that explain the code
- Consistent 4-space indenting in code blocks

6. What do you expect the following code to do? *

1 point

```
display.show(pics.HEART)  
display.show(pics.HAPPY)
```

Mark only one oval.

- Display each image for about 1 second each
- Display the heart quickly and then the happy face
- Displays the heart only
- Displays the happy face only

7. What does the code do? *

1 point

```
delay = 1
```

Mark only one oval.

- Puts the CPU into sleep mode for 1 second
- Assigns the value 1 to a variable named 'delay'
- Delays program execution for 1 second
- Sets the parameter to 1

8. Which of the following is NOT a standard Python type? *

1 point

Mark only one oval.

- 'text'
- 'int'
- 'float'
- 'str'

9. What data type is num = 5

1 point

Mark only one oval.

- float
- Boolean
- integer
- string
- list

10. What data type is num = 4.3

1 point

Mark only one oval.

- float
- Boolean
- integer
- string
- list

11. What data type is choice = False

1 point

Mark only one oval.

- float
- Boolean
- integer
- string
- list

12. What data type is name = 'Angel'

1 point

Mark only one oval.

- float
- Boolean
- integer
- string
- list

13. What data type is `my_colors = ['Red', 'Blue', 'Green', 'White']`

1 point

Mark only one oval.

- float
- Boolean
- integer
- string
- list

14. What will happen when this code is run? *

1 point

```
x = False
if x:
    display.print("Yes")
else:
    display.print("No")
```

Mark only one oval.

- First 'Yes' will print, and the 'No' will print on the display
- Nothing -- there is an error in the code
- 'Yes' will print on the display
- 'No' will print on the display

15. What will happen when this code is run? *

1 point

```
choice = 2
if choice == 0:
    display.show(pics.HAPPY)
if choice == 1:
    display.show(pics.SAD)
if choice == 2:
    display.show(pics.TIARA)
if choice == 3:
    display.show(pics.TSHIRT)
```

Mark only one oval.

- All pictures will be displayed, one after the other.
- Only the Happy face will display
- Only the Tiara will display
- The Tiara and then the Tshirt will be displayed

16. What will happen when this code is run? *

1 point

```
value = 25
if value < 20:
    number = 1
if value < 30:
    number = 2
if value < 40:
    number = 3
```

Mark only one oval.

- number = 1
- number = 2
- number = 3
- numer = 2 and then number = 3

17. What will happen when this code is run? *

1 point

```
value = 25
if value < 20:
    number = 1
elif value < 30:
    number = 2
else:
    number = 3
```

Mark only one oval.

- number = 1
- number = 2
- number = 3
- number = 2 and then number = 3

18. What line of code initializes, or defines, a counter variable? *

1 point

Mark only one oval.

- count = 0
- count = 1
- count = count + 1
- if count == 1:
- def count = 0

19. What line of code increments a counter? *

1 point

Mark only one oval.

- count = 0
- count = 1
- count = count + 1
- if count == 1:
- def count = 1

20. What line of code compares a counter to 1? *

1 point

Mark only one oval.

- count = 0
- count = 1
- count = count + 1
- if count == 1:

21. What does the following line of code do? *

1 point

```
delay = delay + 0.02
```

Mark only one oval.

- decreases the delay variable by 0.02
- increases the delay variable by 0.02
- changes the value of delay to 0.02
- causes an error in the code

22. What are the possible values num, given for the following code? *

1 point

```
num = random.randrange(10)
```

Mark only one oval.

- 10
- 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
- an error will occur because you must give the code a beginning and ending value

23. What is the index of the first item in a list? *

1 point

Mark only one oval.

- 0
- 1
- A
- It depends on the list

24. What is the index of the last item in a list? *

1 point

Mark only one oval.

- len(my_list)
- len(my_list) - 1
- Z
- It depends on the list

25. What is the value of 'color' after the code is executed? *

1 point

```
my_colors = ['red', 'blue', 'green', 'yellow']  
color = my_colors[2]
```

Mark only one oval.

- red
- blue
- green
- yellow
- an error will happen

26. What condition stops the loop in this code? *

1 point

```
index = 0  
while index < 8:  
    index = index + 1  
    display.show('continue')
```

Mark only one oval.

- The loop stops when 'index' reaches 0
- The loop stops when 'index' reaches 8
- It is an infinite loop and never stops
- The statement 'index = index + 1' ends the loop

27. What does the 'break' command do? *

1 point

```
while True:
    if buttson.was_pressed(BTN_A):
        break
```

Mark only one oval.

- Breaks out of the loop
- Causes the code to stop
- Breaks out of the if statement
- Crashes the program

28. The following code is an example of: *

1 point

```
delay = 0.04
num = random.randrange(8)
color = my_colors[num]
```

Mark only one oval.

- iteration
- selection
- sequencing
- randomization

29. The following code is an example of: *

1 point

```
if state == 1:  
    delay = 0.04  
    num = random.randrange(8)  
    color = my_colors[num]
```

Mark only one oval.

- iteration
- selection
- sequencing
- randomization

30. The following code is an example of: *

1 point

```
while count > 0:  
    display.show(my_picc[count])  
    sleep(delay)  
    delay = delay + 0.005  
    index = index + 1
```

Mark only one oval.

- iteration
- selection
- sequencing
- randomization

31. What is a parameter? *

1 point

Mark only one oval.

- A counter
- A type of loop
- A value supplied to a function when it is called
- A value passed to a function when it is called

32. What is an argument? *

1 point

Mark only one oval.

- A counter
- A type of loop
- A value supplied to a function when it is called
- A value passed to a function when it is called

33. What is a global variable? *

1 point

Mark only one oval.

- A variable created outside of a function that can be seen and used throughout the program
- A variable that is created and used in a condition or loop
- A variable that is created inside a function and only exists while the function is running
- A variable that is created specifically for a list

34. What is a local variable? *

1 point

Mark only one oval.

- A variable created outside of a function that can be seen and used throughout the program
- A variable that is created and used in a condition or loop
- A variable that is created inside a function and only exists while the function is running
- A variable that is created specifically for a list

35. When do you need to use the 'global' command, like shown below? *

1 point

```
def show_random_die(delay):  
    global num
```

Mark only one oval.

- Every time you declare a global variable
- When you use a global variable in a condition or loop
- When you change the value of a global variable outside a function
- When you change the value of global variable inside a function

36. Which statement is NOT true about functions? *

1 point

Mark only one oval.

- Functions can only use local variables.
- You can reuse code by calling functions multiple times.
- Functions help keep code organized and readable.
- It is easier to make a change to code in one function than in repeated code.

37. What are the final colors of the pixels after the code is run? *

1 point

```
pixels.set([BLUE, BLUE, BLUE, BLUE])
pixels.set(2, RED)
```

Mark only one oval.

- BLUE, RED, BLUE, BLUE
- OFF, RED, OFF, OFF
- BLUE, BLUE, RED, BLUE
- OFF, OFF, RED, OFF
- RED, RED, RED, RED

38. What is the correct function CALL for the function below? *

1 point

```
def show_random_arrow(index):
    arrow = random.randrange(8)
    display.show(MY_ARROW_LIST[arrow])
```

Mark only one oval.

- index = show_random_arrow(index)
- def show_random_arrow(index):
- show_random_arrow(index)
- show_random_arrow()

39. What is the correct function CALL for the function below? *

1 point

```
def assign_cost(price, shirts):  
    cost = price * shirts  
    return cost
```

Mark only one oval.

- cost = assign_cost(price, shirts)
- def assign_cost(price, shirts):
- assign_cost(price, shirts)
- assign_cost(cost)

40. What is the error in the code below? *

1 point

```
index = 0  
while Index < 8:  
    index = index + 1  
    display.show('continue')
```

Mark only one oval.

- The index variable is misspelled.
- The indenting is not correct.
- The loop will never start.
- The while loop block does not need a colon (:)

41. What is the error in the code below? *

1 point

```
while True:
    choice = 2
    if choice == 0:
        display.show(pics.HAPPY)
    if choice == 1:
        display.show(pics.SAD)
    if choice == 2:
        display.show(pics.TIARA)
```

Mark only one oval.

- The name choice can't be used as a variable.
- The indenting is not correct.
- The loop will never end.
- The while loop block does not need a colon (:)

42. What is the error in the code below? *

1 point

```
while True:
choice = 2
if choice == 0:
    display.show(pics.HAPPY)
if choice == 1:
    display.show(pics.SAD)
if choice == 2:
    display.show(pics.TIARA)
```

Mark only one oval.

- The assignment should be choice == 2
- The indenting is not correct.
- A = should be used instead of ==.
- The if statements do not need a colon (:)

43. What is the error in the code below? *

1 point

```
index = 0
while index < 8
    index = index + 1
    display.show('continue')
```

Mark only one oval.

- The index variable is misspelled.
- The indenting is not correct.
- The loop will never end.
- The while loop block needs a colon (:)

This content is neither created nor endorsed by Google.

Google Forms

