Name:_____ **Partners:** _____

Python Activity 6: Boolean Expressions

Writing programs that can make decisions!

Learning Objectives

Students will be able to:

Content:

- Explain the three types of programming structures
- Explain how conditional operators and logical operators are used in programming
- Use conditional operators with strings and numeric values

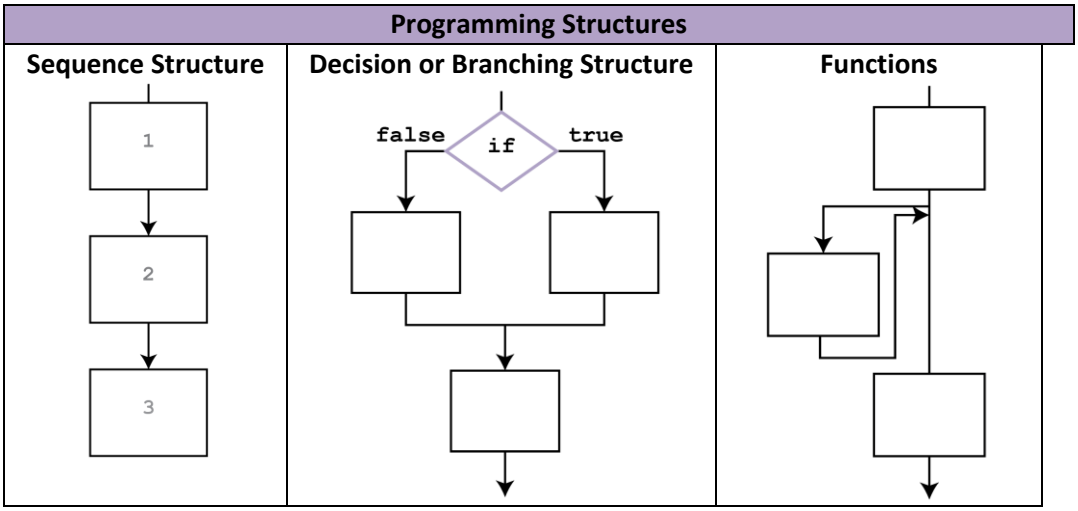
Process:

- Write correct Boolean expressions and compound expressions

Prior Knowledge

- Variables, arithmetic expressions

Concept Model



CM1. Which structure(s) best describe the types of Python programs you have seen so far?

_____

CM2. Which structure allows the code to decide what code is executed when the program is run?

_____

FYI: Conditional operators, also known as relational operators, are used to compare the relationship between two operands. Expressions whose result can only be `True` or `False` are known as **Boolean expressions.**

Critical Thinking Questions

1. State the meaning of each of the following **conditional operators.** If you are not sure of the meaning of any symbol, make a guess, and we'll put them into *Interactive* Python as a class (See Figure to the right).

```
>>> 34 < 56
True
```

a. < _____ b. > _____

c. <= _____ d. >= _____

e. != _____ f. == _____

2. What will be the result of each of the following expressions?
   Assume: `x = 4, y = 5, and z = 4`

a. `x > y` _____

b. `x < y` _____

c. `x == y` _____

d. `x != y` _____

e. `x >= z` _____

f. `x <= z` _____

g. `x + y > 2 * x` _____

h. `y * x - z != 4 % 4 + 16` _____

i. `pow(x,2) == abs(-16) #guess!` _____

3. What will be the result of the following expressions?
   Assume: `word1 = "hello"` and `word2 = "good-bye"`

a. `word1 = = word2` _____

b. `word1 != word2` _____

c. `word1 < word2` _____

d. `word1 >= word2` _____

4. How do the conditional operators work when the operands are strings? _____

_____

5. What are the two possible answers for each expression in questions 4 and 5? _____

FYI: We can use **logical operators** to determine logic between conditions (relational expressions).

6. Sometimes you want to test more than one condition to determine which code segment should be executed. You can use the following **logical operators** to create **compound conditions**. Examine each operator and a sample of its use. Provide an explanation of how each operator works.

Operator	Example	Explanation
and	(age >= 17) **and** (has_license = = True)	
or	(cost < 20.00) **or** (shipping = = 0.00)	
not	**not** (credits> 120)	

7. Assume the value of the variable **num_books** is 40. State the values of each of the Boolean expression.

Expression	Value
(num_books > 5) and (num_books < 100)	
(num_books < 5) or (num_books > 100)	
not(num_books * 10 == 100)	

Application Questions: Use the Python Interpreter to check your work

1. Assign a value to **num1** and **num2.** Write a Boolean expression that tests if the value stored in the variable **num1** is equal to the value stored in the variable **num2**.

   _____

   _____

   _____

2. Assign a value to the variables listed in this problem (time, max_time, cost, and max_cost). Write a Boolean expression that tests if the value stored in the variable **time** is less than the value stored in the variable **max_time** or if the value stored in the variable **cost** is less than the value stored in the variable **max_cost**

   _____

   _____

   _____

   _____

   _____

3. Assign a value to **weight** and **cost.** Write a Boolean expression that tests if the value stored in weight is < 10 and the value store in cost is not greater than 20.00

   _____

   _____

   _____