NAVID \_ ATALAR   
1= What possible values can a Boolean expression have?  
  
A Boolean expression can only have two possible values: true or false. These are also sometimes represented as 1 for true and 0 for false.  
  
2=Where does the term Boolean originate?  
The term Boolean originates from the name of George Boole, an English mathematician and philosopher who developed Boolean algebra in the mid-19th century. Boolean algebra is a branch of algebra that deals with logic operations and is fundamental to the design of digital electronic circuits and computer programming.  
  
3. What is an integer equivalent to True in Python?  
In Python, the integer equivalent to True is 1  
  
4. What is the integer equivalent to False in Python?   
In Python, the integer equivalent to False is 0  
  
5. Is the value -16 interpreted as True or False?   
In Python, any non-zero value is interpreted as True. Therefore, the value -16 is interpreted as True.  
  
6. Given the following definitions:  
Evaluating the following Boolean expressions with the given definitions:  
(a) x == 3 evaluates to True.  
(b) x < y evaluates to True.  
(c) x >= y evaluates to False.  
(d) x <= y evaluates to True.  
(e) x != y - 2 evaluates to False.  
(f) x < 10 evaluates to True.  
(g) x >= 0 and x < 10 evaluates to True.  
(h) x < 0 and x < 10 evaluates to False.  
(i) x >= 0 and x < 2 evaluates to False.  
(j) x < 0 or x < 10 evaluates to True.  
(k) x > 0 or x < 10 evaluates to True.  
(l) x < 0 or x > 10 evaluates to False  
  
  
7=Given the following definitions: x, y = 3, 5 b1, b2, b3, b4 = True, False, x == 3, y < 3 evaluate the following Boolean expressions:  
(a) b3 evaluates to True, since x is assigned to 3 and 3 == 3 is True.  
(b) b4 evaluates to False, since y is assigned to 5 and 5 < 3 is False.  
(c) not b1 evaluates to False, since b1 is assigned to True and not True is False.  
(d) not b2 evaluates to True, since b2 is assigned to False and not False is True.  
(e) not b3 evaluates to False, since b3 is assigned to True and not True is False.  
(f) not b4 evaluates to True, since b4 is assigned to False and not False is True.  
(g) b1 and b2 evaluates to False, since b2 is assigned to False and False and True is False.  
(h) b1 or b2 evaluates to True, since b1 is assigned to True and True or False is True.  
(i) b1 and b3 evaluates to True, since both b1 and b3 are assigned to True and True and True is True.  
(j) b1 or b3 evaluates to True, since b1 is assigned to True and True or True is True.  
(k) b1 and b4 evaluates to False, since b4 is assigned to False and True and False is False.  
(l) b1 or b4 evaluates to True, since b1 is assigned to True and True or False is True.  
(m) b2 and b3 evaluates to False, since b2 is assigned to False and False and True is False.  
(n) b2 or b3 evaluates to True, since b3 is assigned to True and False or True is True.  
(o) b1 and b2 or b3 evaluates to True, since b1 is assigned to True and True and False or True is True.  
(p) b1 or b2 and b3 evaluates to True, since b1 is assigned to True and False and True or True is True.  
(q) b1 and b2 and b3 evaluates to False, since b2 is assigned to False and True and False and True is False.  
(r) b1 or b2 or b3 evaluates to True, since b1 is assigned to True and True or False or True is True.  
(s) not b1 and b2 and b3 evaluates to False, since b1 is assigned to True and not True and False and True is False.  
(t) not b1 or b2 or b3 evaluates to True, since b1 is assigned to True and `not True   
(u) not (b1 and b2 and b3): True.  
(w) not b1 and not b2 and not b3: False.  
(x) not b1 or not b2 or not b3: True.  
(y) not (not b1 and not b2 and not b3): True.  
(z) not (not b1 or not b2 or not b3): False.  
  
8. Express the following Boolean expressions in simpler form; that is, use fewer operators or fewer symbols. x is an integer.  
symbols. x is an integer.  
(a) not (x == 2)  
(b) x < 2 or x == 2  
(c) not (x < y)  
(d) not (x <= y)  
(e) x < 10 and x > 20  
(f) x > 10 or x < 20  
(g) x != 0  
(h) x == 0  
  
9. Express the following Boolean expressions in an equivalent form without the not operator. x and y are integers  
x and yare integers.  
(a) not (x == y)  
(b) not (x > y)  
(c) not (x < y)  
(d) not (x >= y)  
(e) not (x <= y)  
(f) not (x != y)  
(g) not (x != y)  
(h) not (x == y and x < 2)  
(i) not (x == y or x < 2)  
(j) not (not (x == y))  
  
10. What is the simplest tautology?  
The simplest tautology is True (which is always true regardless of any input).  
 11. What is the simplest contradiction?   
The simplest contradiction is False (which is always false regardless of any input).  
  
12. Write a Python program that requests an integer value from the user. If the value is between 1 and 100 inclusive, print ”OK;” otherwise, do not print anything.   
  
value = int(input("Enter an integer between 1 and 100 inclusive: "))

if 1 <= value <= 100:

print("OK;")  
  
  
  
13. Write a Python program that requests an integer value from the user. If the value is between 1 and 100 inclusive, print ”OK;”   
value = int(input("Enter an integer between 1 and 100 inclusive: "))

if 1 <= value <= 100:

print("OK;")

else:

print("Out of range.")  
  
14. Write a Python program that allows a user to type in an English day of the week (Sunday, Monday, etc.). The program should print the Spanish equivalent, if possible.   
  
english\_day = input("Enter an English day of the week: ")

if english\_day.lower() == "sunday":

print("Domingo")

elif english\_day.lower() == "monday":

print("Lunes")

elif english\_day.lower() == "tuesday":

print("Martes")

elif english\_day.lower() == "wednesday":

print("Miércoles")

elif english\_day.lower() == "thursday":

print("Jueves")

elif english\_day.lower() == "friday":

print("Viernes")

elif english\_day.lower() == "saturday":

print("Sábado")

else:

print("Sorry, I don't know how to say that in Spanish.")  
15. Consider the following Python code fragment  
(a) The code will print "i = 5 j = 5 k = 7"  
(b) The code will print "i = 3 j = 7 k = 5"  
(c) The code will print "i = 5 j = 3 k = 7"  
(d) The code will print "i = 5 j = 7 k = 3"  
(e) The code will print "i = 7 j = 3 k = 5"  
(f) The code will print "i = 7 j = 5 k = 3"  
  
16=Consider the following Python program that prints one line of text:  
(a) The program will print "wow 3"  
(b) The program will print "whoa 21"  
(c) The program will print nothing  
(d) The program will print "27"  
(e) The program will print nothing (as the input is negative and therefore <10)  
  
17. Consider the following two Python programs that appear very similar  
The two programs behave differently because of the use of the elif statement in the second program. In the first program, each condition is checked independently, so all four if statements could potentially be true for a given input. In the second program, once a condition is true, the program skips the remaining elif statements and goes straight to the print() statement.  
(a) Both programs print four asterisks.  
(b) Both programs print three asterisks.  
(c) The first program prints two asterisks, while the second program prints one asterisk.  
(d) The first program prints one asterisk, while the second program prints none.  
(e) Both programs print no asterisks.  
(f) Both programs print no asterisks.  
  
18=Write a Python program that requests five integer values from the user. It then prints the maximum and minimum values entered. If the user enters the values 3, 2, 5, 0, and 1, the program would indicate that 5 is the maximum and 0 is the minimum. Your program should handle ties properly; for example, if the user enters 2, 4, 2, 3, and 3, the program should report 2 as the minimum and 4 as maximum.  
  
nums = []

for i in range(5):

num = int(input("Enter an integer: "))

nums.append(num)

max\_num = max(nums)

min\_num = min(nums)

print("Maximum value:", max\_num)

print("Minimum value:", min\_num)  
  
19. Write a Python program that requests five integer values from the user. It then prints one of two things: if any of the values entered are duplicates, it prints "DUPLICATES"; otherwise, it prints "ALL UNIQUE".  
  
nums = []

for i in range(5):

num = int(input("Enter an integer: "))

nums.append(num)

if len(nums) != len(set(nums)):

print("DUPLICATES")

else:

print("ALL UNIQUE")