IHF: CODE - PYTHON WORKSHEET 2

Comparators

< less than
<= less than or equal to
> greater than
>= greater than or equal to
== equal to
!= not equal to

Conditional statements

If statements:

An if statement lets us decide what to do: if True, then do this, if False, then don't do this

Else statements:

If the first statement is False, you can use an else statement as a catch all statement. The else statement will be executed if the if statement is not satisfied.

Elif statements:

If you want to check more than two alternatives, you can use the elif statement.

When using if, elif, else statements; the if statement is used first, then as many elif statements as required and then ends on an else statement.

And/or/not statements:

You can combine conditionals with **and** or **or**.

When using **and**, if one of the conditions are false, the result will be **false**.

When using **or**, if one of the conditions are true, the result will be **true**.

When using **not**, if the condition satisfies the not statement, the result will be **true**. This can be useful for filtering.

Lists

Python uses lists to group data together.

Lists are written in square brackets [] and are separated by commas. Lists are ordered, can have duplicates and can be amended.

A list can contain any data type within Python, e.g. strings, booleans, dictionaries.
Example: animals = ["cat", "dog", "fish", "bird"]

- To create a list: list_name = []
- To add to the list: list_name.append("new_item")
- To amend value of list: list[2] = "new_value"
- To delete from the list: del list_name[3]
- To find the length of a set: print(len(list_name))
- To sort a list alphabetically: list_name.sort()
- To reverse a list order: list_name.sort(.sort(reverse=True))
- To check if a specified item exists in a list:
  if "x" in list_name
      print("x is in the list")

For loops

For loops iterate through a collection of items. Every time the loop runs, the selected item is available to be used.
Example: a for loop on a list would print each item of the loop in a row
Apples
Pears
Bananas

Ranges

You can use ranges with loops to determine how many times you want the loop to run.
range (10) - will allow a loop to iterate through the numbers 0-9
range(2001,2009, 3) - this range has a start, stop and step parameter. This step parameter will add 3 to each year.

CODE EXAMPLES

SECTION A & B

```
1. x = 5
   y = 6
   if x < y:
       print("X is smaller than Y")
   # X is smaller than Y

2. x = 9
   y = 7
   if x < y:
       print("X is smaller than Y")
   else:
       print("Y is smaller than X")
   # X is smaller than Y
```

```
3. x = 8
   y = 8
if x < y:
    print("X is smaller than Y")
elif x > y:
    print("Y is smaller than X")
else:
    print("X is equal to Y")
# X is equal than Y
```

```
4. x = 5
   y = 7
   z = 7
if x < y and y == z:
    print("X is smaller than Y but equal to Z")
# X is smaller than Y but equal to Z

if x == y or y == z:
    print("One condition is satisfied")
# One condition is satisfied
```

SECTION C

```
1. names = ["Alice", "Bob", "Charlie"]
   print(names[0]) # Alice
   print(names[1]) # Bob
   print(names[2]) # Charlie

2. names = ["Alice", "Bob", "Charlie"]
   names.append("Dave")
   print(names)
   # ["Alice", "Bob", "Charlie", "Dave"]

3. names = ["Alice", "Bob", "Charlie"]
   names[2] = "Chris"
   print(names)
   # ["Alice", "Bob", "Chris"]

4. names = ["Alice", "Bob", "Charlie"]
   del(names[1])
   print(names)
   # ["Alice", "Charlie"]

5. names = ["Alice", "Bob", "Charlie"]
   if "Eve" in names:
       print("Eve is here")
   else:
       print("Eve isn't here")

6. names = ["Alice", "Bob", "Charlie"]
   print(len(names)) #3
```

```
7. names = ["Charlie", "Alice", "Bob"]
   names.sort()
   print(names)
   # ["Alice", "Bob", "Charlie"]
```

SECTION D

```
1. names = ["Alice", "Bob", "Charlie"]
   for person in names:
       print(person)
   # Alice
   # Bob
   # Charlie

2. for my_number in range(5):
   print(my_number)
# 0
# 1
# 2
# 3
# 4

3. for years in range(1991, 2001, 3):
           print(years)
# 1991
# 1994
# 1997
# 2000
```

QUESTIONS

SECTION A

1. Ask for the user's name, if they are called "Bob", print "Welcome Bob!".
2. Ask for the user's name, if they are not called "Alice", print "You're not Alice!".
3. Ask the user for a password, if they enter the password "qwerty123", print "You have successfully logged in". If they get it wrong, print "Password failure".
4. Ask the user to enter a number, if the number is even, print "Even", otherwise print "Odd".
5. Ask the user for 2 different numbers, if the total of the two numbers is over 21, print "Bust" otherwise print "Safe".

SECTION B

1. Ask for the user's name, if they are called "Alice" print "Hello, Alice", if they are called "Bob", print "What's up Bob!", else print "You must be Charlie"
2. Ask the user to enter their age:
   i. If they are younger than 11, print "You're too young to go to this school".
   ii. If they are between 11 and 16, print "You can can come to this school".
   iii. If they are over 16, print 'You're too old for school".
   iv. If they are 0, print "You're not born yet!".
3. Ask the user to enter the name of a month. If the user enters March/April/May" print "<month> is in Spring", otherwise print "I don't know".
   i. Expand for the rest of the year, given that Summer is June/July/August. Autumn is September/October/November. Winter is December/January/February.
   ii. Ensure that when an unknown month is given it prints "I don't know".
4. Ask the user for two different numbers, if both numbers are even, print "Even", if both numbers are odd, print "Odd", else print the product of the two numbers.

QUESTIONS

SECTION C

1. Create the following list of items: apples, cherries, pears, pineapple, peaches, mango.
2. Add "grapes" to the list.
3. Change "pears" to "strawberries".
4. Remove "apples" from the list.
5. Print out the current length of the list.
6. Print out your list.
7. Order the list alphabetically.
8. Print out the list again.

SECTION D

1. Loop through the list you created in section C and print each item out.
2. Print all the numbers 1 to 100 (including the number 100).
3. Print all odd numbers from 1 to 100.
4. The modern olympics started in 1896, print all the years they have been held.
5. FizzBuzz - Write a program that prints the numbers from 1 to 100. For multiples of three, print “Fizz” instead of the number and for multiples of five, print “Buzz”. For numbers which are multiples of both three and five, print “FizzBuzz”.

FIZZBUZZ:

```
1
2
fizz
4
buzz
fizz
7
8
fizz
buzz
11
fizz
13
14
fizzbuzz
```