INTERNSHIP TASKS

Name : S. Deva Manikanta

Clg Id : 12119003

Course : Python

Org : IGIAT – VSKP

Date : 31-03-2024

Day 5: EXERCISE – 5

Exercise Level 1

#Task 1:

print("Task 1 : Declare an empty list")*;*

my\_list = []*;*

#Task 2:

print("\n\nTask 2 : Declare a list with more than 5 items")*;*

my\_list = [1, 2, 3, 4, 5, "Deva Manikanta"]*;*

#Task 3:

print("\n\nTask 3 : Find the length of your list")*;*

print("The length of my list : ", len(my\_list))*;*

#Task 4:

print("\n\nTask 4 : Get the first item, the middle item and then the last item of the list")*;*

print("The First Item : ", my\_list[0])*;*

print("The Middle Item : ", my\_list[int(len(my\_list) / 2)])*;*

print("The Last item : ", my\_list[-1])*;*

#Task 5:

print("\n\nTask 5 : Declare a list mixed\_data\_types, put your --> name, age, height, maritial status, address")*;*

mixed\_data\_types = ["Deva Manikanta", 20, 150.5, "unmarried", "Palacole-53460"]*;*

#Task 6:

print("\n\nTask 6 : Declare a list variable named it\_companies and assign initial values Facebook, Google, Microsoft, Apple, IBM, Oracle, and Amazon")*;*

it\_companies = ["Facebook", "Google", "Microsoft", "Apple", "IBM", "Oracle", "Amazon"]*;*

#Task 7:

print("\n\nTask 7 : Print the list using print()")*;*

print("List = it\_companies : ", it\_companies)*;*

#Task 8:

print("\n\nTask 8 : Print the number of companies in the list")*;*

print("No.of companies : ", len(it\_companies))*;*

#Task 9:

print("\n\nTask 9 : Print the first, middle, and last company")*;*

print("The First Company : ", it\_companies[0])*;*

print("The Middle Company : ", it\_companies[int(len(it\_companies) / 2)])*;*

print("The Last Company : ", it\_companies[-1])*;*

#Task 10:

print("\n\nTask 10 : Print the list after modifying one of the companies")*;*

it\_companies[0] = "Meta"*;*

print("The Modified list : ", it\_companies)*;*

#Task 11:

print("\n\nTask 11 : Add an IT company to it\_companies")*;*

it\_companies.append("Infosys")*;*

print(it\_companies)*;*

#Task 12:

print("\n\nTask 12 : Insert an IT company in the middle of the companies list")*;*

it\_companies.insert(int(len(it\_companies)/2) , "TCS")*;*

print(it\_companies)*;*

#Task 13:

print("\n\nTask 13 : Change one of the it\_companies names to uppercase")*;*

import random as r;

random = r.randrange(0, len(it\_companies))*;*

it\_companies[random] = it\_companies[random].upper()*;*

print("The Upper case converted for : ", it\_companies[random])*;*

#Task 14:

print("\n\nTask 14: Join the it\_companies with a string \'#;\'")*;*

string = '#; '.join(it\_companies)*;*

print(string)*;*

#Task 15:

print("\n\nTask 15: Check if a certain company exists in the it\_companies list")*;*

is\_meta\_exists = ("Meta" or "META") in it\_companies*;*

print("Does \'Meta\' or \'META\' exists in the list: ", is\_meta\_exists)*;*

#Task 16:

print("\n\nTask 16: Sort the list using sort() method")*;*

it\_companies.sort()*;*

print("Sorted : ", it\_companies)*;*

#Task 17:

print("\n\nTask 17: Reverse the list descending order using reverse() method")*;*

it\_companies.reverse()*;*

print("Desending order : ", it\_companies)*;*

#Task 18:

print("\n\nTask 18: Slice out the first 3 companies from the list")*;*

it\_companies.sort()*;*

print("The First 3 companies : ", it\_companies[0:3])*;*

#Task 19

print("\n\nTask 19: Slice out the last 3 companies from the list:")*;*

print("The Last 3 companies : ", it\_companies[-3:])*;*

#Task 20

import math as m;

print("\n\nTask 20: Slice out the middle IT company or companies from the list")*;*

print("The Middle companies : ", it\_companies[int(m.floor(len(it\_companies)/2)) : int(m.ceil((len(it\_companies)/2)))])*;*

#Task 21

print("\n\nTask 21: Remove the first IT company from the list")*;*

first\_company = it\_companies[0]*;*

it\_companies.remove(first\_company)*;*

print(f"After removing : {first\_company} : {it\_companies}")*;*

#Task 22

print("\n\nTask 22: Remove the middle IT company or companies from the list")*;*

middle\_company = it\_companies[int(len(it\_companies) / 2)]*;*

it\_companies.remove(middle\_company)*;*

print(f"After removing : {middle\_company} : {it\_companies}")*;*

#Task 23

print("\n\nTask 23: Remove the last IT Company from the list")*;*

last\_company = it\_companies[-1]*;*

it\_companies.remove(last\_company)*;*

print(f"After removing : {last\_company} : {it\_companies}")*;*

#Task 24

print("\n\nTask 24: Remove all IT companies from the list")*;*

it\_companies.clear()*;*

print("The list : ", it\_companies)*;*

#Task 25:

print("\n\nTask 25: Destroy the IT Companies list")*;*

try:

    del it\_companies; #destroying aka deleting the list.

    print(it\_companies); #throws an exception that list is not accessible.

except Exception as e:

    print(f"The list is not accessible it is removed or deleted!\nError : {e}")*;*

#Task 26:

print("\n\nTask 26: Join the following lists:")*;*

front\_end = ['HTML', 'CSS', 'JS', 'React', 'Redux']*;*

back\_end = ['Node', 'Express', 'MongoDB']*;*

print(front\_end)*;*

print(back\_end)*;*

joined = front\_end + back\_end*;*

print("Joined : ", joined)*;*

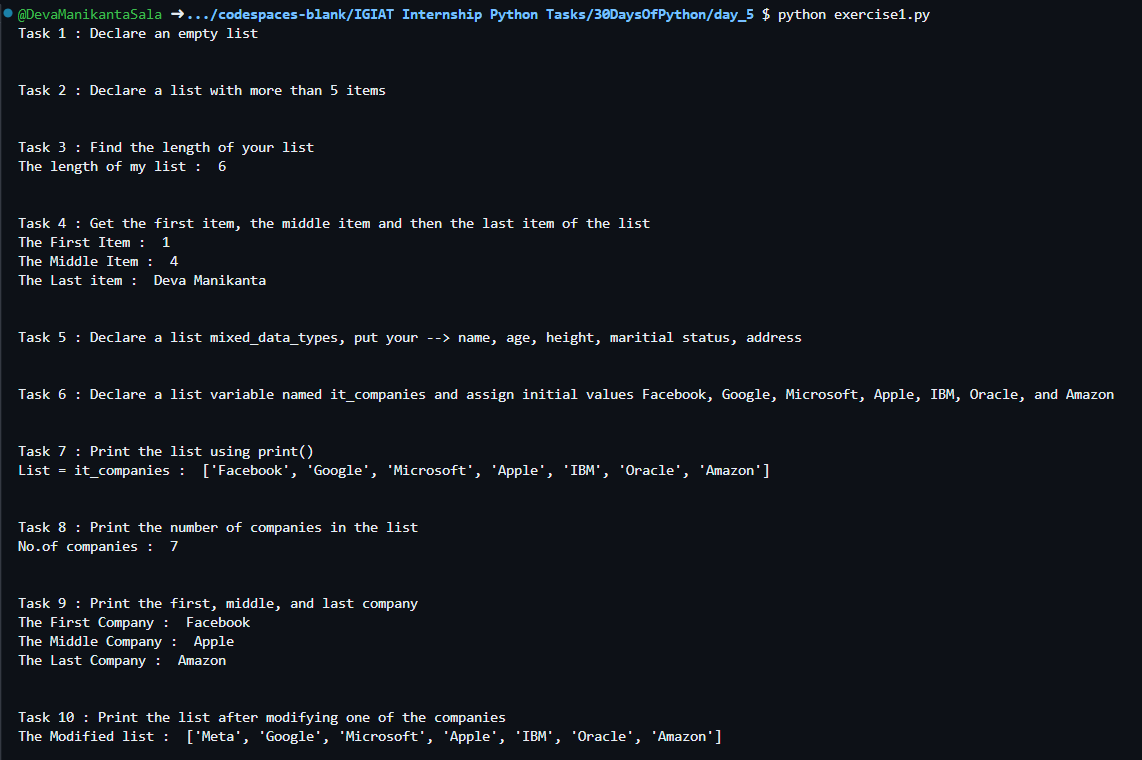
#Task 27:

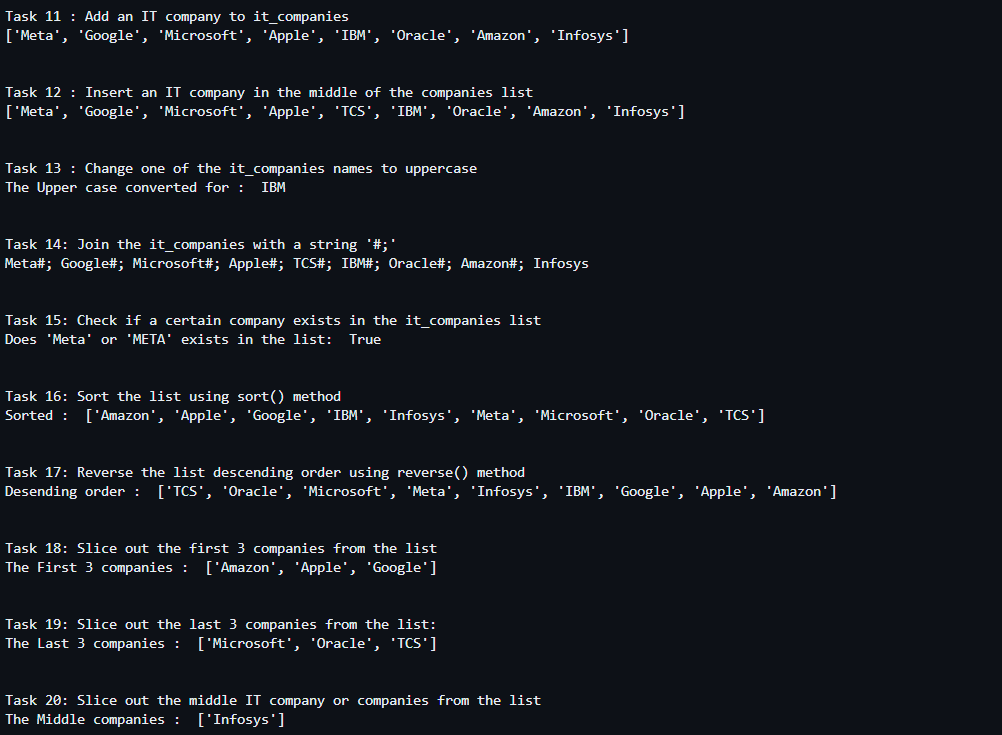
print("\n\nTask 27: After joining the lists in task 26. Copy the joined list and assign it to a variable full\_stack.\nThen insert Python and SQL after Redux")*;*

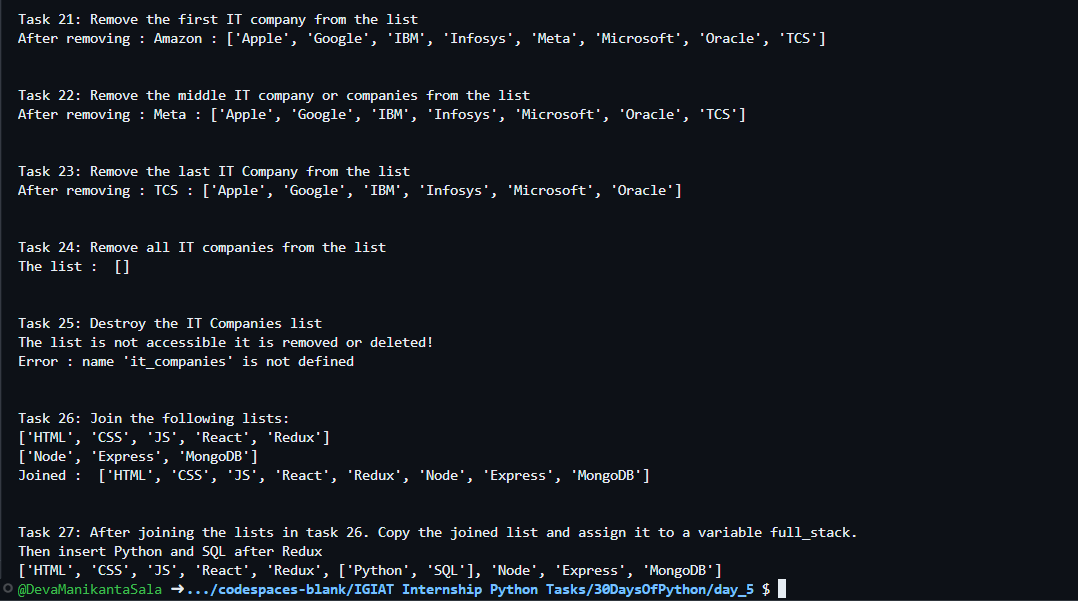
full\_stack = joined.copy()*;*

full\_stack.insert(full\_stack.index("Redux") + 1, ["Python", "SQL"])*;*

print(full\_stack)*;*

**Outputs:** 

****

****

Exercise Level 2

#Task 1

print("\n\nTask 1: The Following is a list of 10 students ages")*;*

ages = [19, 22, 19, 24, 20, 25, 26, 24, 25, 24]*;*

print(ages)*;*

#Task 2

print("\n\nTask 2: Sort the list and find the min and max age")*;*

ages.sort()*;*

print("Min Age : ", min(ages))*;*

print("Max Age : ", max(ages))*;*

#Task 3

print("\n\nTask 3: Add the min age and max age again to the list")*;*

ages.append(min(ages))*;*

ages.append(max(ages))*;*

print(ages)*;*

#Task 4

import statistics as st; #pip install statistics

print("\n\nTask 4: Find the median age (one middle item or two middle items divided by two)")*;*

ages.sort()*;*

median = int(st.median(ages))*;*

print("The median age : ", median)*;*

#Task 5

print("\n\nTask 5: Find the average age")*;*

average = sum(ages)/len(ages)*;*

print("The average age : ", average)*;*

#Task 6

print("\n\nFind the range of the ages")*;*

range\_ages = max(ages) - min(ages)*;*

print("The Range of ages : ", range\_ages)*;*

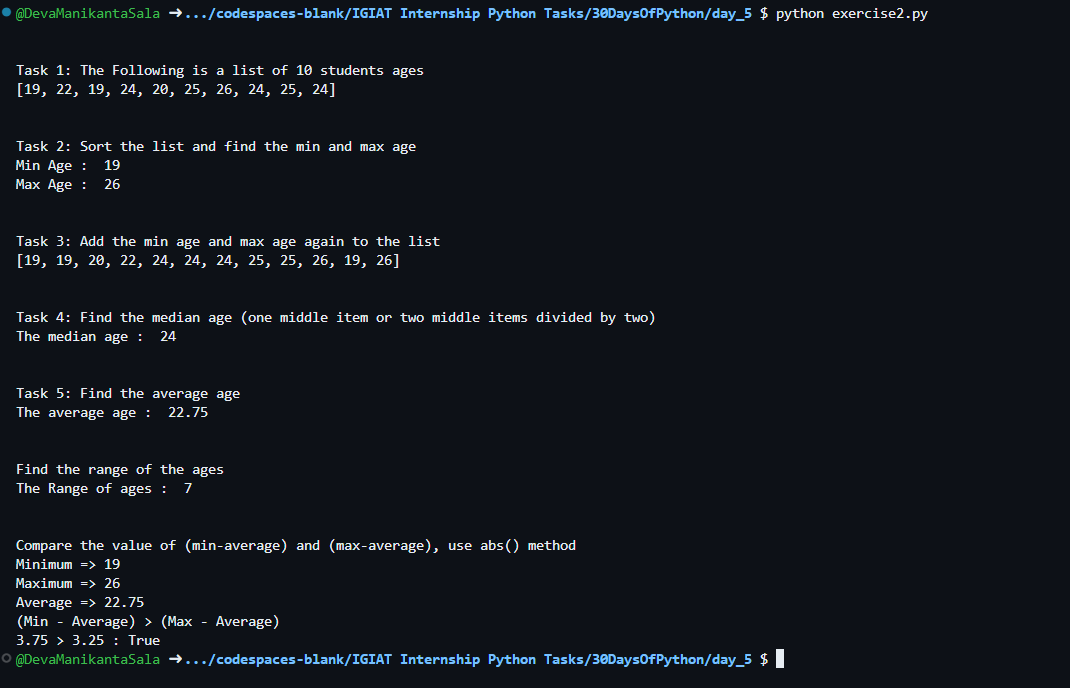
#Task 7

print("\n\nCompare the value of (min-average) and (max-average), use abs() method")*;*

print(f"Minimum => {min(ages)}\nMaximum => {max(ages)}\nAverage => {average}")*;*

print(f"(Min - Average) > (Max - Average)\n{abs(min(ages)-average)} > {abs(max(ages)-average)} : {abs(min(ages)-average) > abs(max(ages)-average)}")*;*

**Outputs:**

****