fingerTips

ASSIGNMENT QUESTIONS & ANSWERS

1. One-word answers(1*5=5)
2. Conditionals(2*10=20)
3. Lists(2*10=20)
4. Tuple(2*10=20)
5. Sets(2*10=20)
6. Dictionary(2*10=20)
7. Operators(1*5=5)
8. for-loop(3*10=30)
9. while-loop(3*10=30)
10. Functions(3*10=30)

Total=200 Marks

One-word answers, 1-Mark(1*5=5)

Q1. Which of the following words cannot be used as a variable name in Python?

1. for
2. while
3. class
4. All of the above

Solution:

1. All of the above

Q2. What will be the output of the following code:

In [1]:
```
x = range(6)
for n in x:
    pass
print(n)
```

```
5
```

Solution:

5

Q3. What is the output of the following code:

In [2]:
```
x = [5,6,7,8,9]
print(4 in x)
```

```
False
```

Solution:

False

Q4. What is the output of the following code:

In [3]:
```
x = {"name" : "John", "age" : 36};
print(x["age"])
```

```
36
```

Solution:

36

Q5. Which of these are correct identifier names:

1. _python
2. python12_
3. 12_python
4. python@12
5. python_12

Solution:

1. _python
2. python12_
3. python_12

Conditionals, 2-Marks(2*10=20)

Q1. What will be the output of following code?

In [4]:
```
x = 15
if x % 2 == 0:
    print("x is even")
else:
    print("x is odd")
```

```
x is odd
```

Solution:

x is odd

Q2. What will be the output of following code?

In [5]:
```
x = 5
if x > 0:
    print("Positive")
```

```
Positive
```

Solution:

"Positive"

Q3. What will be the output of following code?

In [6]:
```
x = 0
if x > 0:
    print("Positive")
elif x < 0:
    print("Negative")
else:
    print("Zero")
```

```
Zero
```

Solution:

Zero

Q4. What will be the output of following code?

In [7]:
```
x = [1, 2, 3]
if 2 in x:
    print("m")
else:
    print("2 is not in the list")
```

```
m
```

Solution:

2 is in the list

Q5. What will be the output of following code?

In [8]:
```
x = 5
if x > 0 or x < -5:
    print("x is either greater than 0 or less than -5")
else:
    print("x is not greater than 0 or less than -5")
```

```
x is either greater than 0 or less than -5
```

Solution:

x is either greater than 0 or less than -5

Q6. What will be the output of following code?

In [9]:
```
x = True
if not x:
    print("x is False")
else:
    print("x is True")
```

```
x is True
```

Solution:

x is True

Q7. What will be the output of following code?

In [10]:
```
x = 5
y = 10
z = 15
if x < y < z:
    print("x is less than y and y is less than z")
else:
    print("x is greater than or equal to y or y is greater than or equal to z")
```

```
x is less than y and y is less than z
```

Solution:

x is less than y and y is less than z

Q8. What will be the output of following code?

In [11]:
```
x = "python"
if x.startswith("p") and x.endswith("n"):
    print("x starts with p and ends with n")
else:
    print("x does not start with p or does not end with n")
```

```
x starts with p and ends with n
```

Solution:

x starts with p and ends with n

Q9. What will be the output of following code?

In [12]:
```
x = ["apple", "banana", "cherry"]
if x[0] == "apple" and x[-1] == "cherry":
    print("The first or last item in the list is apple or cherry")
else:
    print("The first or last item in the list is not apple or cherry")
```

```
The first or last item in the list is apple or cherry
```

Solution:

The first or last item in the list is apple or cherry

Q10. What will be the output of following code?

In [13]:
```
x = "Hello"
if x.islower() == True:
    print("x is not all lowercase")
else:
    print("x is all lowercase")
```

```
x is all lowercase
```

Solution:

x is all lowercase

Lists, 2-Marks(2*10=20)

Q1. Write a code to append 'Deep Learning' in list courses=['Python', 'Machine Learning', 'NLP']

In [14]:
```
#SOLUTION

courses=['Python', 'Machine Learning', 'NLP']
courses.append('Deep Learning')
print(courses)
```

```
['Python', 'Machine Learning', 'NLP', 'Deep Learning']
```

Q2. Write a code to insert 'Deep Learning' after 'Python' in list courses= ['Python', 'Machine Learning', 'NLP']

In [15]:
```
#SOLUTION

courses=['Python', 'Machine Learning', 'NLP']
courses.insert(1, 'Deep Learning')
print(courses)
```

```
['Python', 'Deep Learning', 'Machine Learning', 'NLP']
```

Q3. Write a code to remove 'Deep Learning' in list courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']

In [16]:
```
#SOLUTION

courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']
courses.remove('Deep Learning')
print(courses)
```

```
['Python', 'Machine Learning', 'NLP']
```

s the length of list, courses=['Python', 'Deep Learning', 'Machine

Learning', 'NLP']

In [17]:
```
#SOLUTION

courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']
print(len(courses))
```

```
4
```

Q5. Write a code to delete last item of the list, courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']

In [18]:
```
#SOLUTION

courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']
courses.pop()
print(courses)
```

```
['Python', 'Deep Learning', 'Machine Learning']
```

Q6. Write a code to sort the items of the list, courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']

In [19]:
```
#SOLUTION

courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']
courses.sort()
print(courses)
```

```
['Deep Learning', 'Machine Learning', 'NLP', 'Python']
```

Q7. Write a code to reverse the items of the list, courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']

In [20]:
```
#SOLUTION

courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']
courses.reverse()
print(courses)
```

```
['NLP', 'Machine Learning', 'Deep Learning', 'Python']
```

Q8. Write a code to find whether 'NLP' is in the list, courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']

In [21]:
```
#SOLUTION

courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']
print("NLP" in courses)
```

```
True
```

Q9. Write a code to count number of 'NLP' in the list, courses=['Python', 'NLP', 'Deep Learning', 'NLP', 'Machine Learning', 'NLP']

In [22]:
```
#SOLUTION

courses=['Python', 'NLP', 'Deep Learning', 'NLP', 'Machine Learning', 'NLP']
('NLP')
```

Out[22]:
```
3
```

Q10. Write a code to clear the list, courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']

In [23]:
```
#SOLUTION

courses=['Python', 'Deep Learning', 'Machine Learning', 'NLP']
courses.clear()
print(courses)
```

```
[]
```

Tuples, 2-Marks(2*10=20)

Q1. write a code to concatenate 2 tuples, t1 = (23, 'python', 'coders') & t2 = (45.78, 'ML', 'NLP')

In [24]:
```
t1 = (23, 'python', 'coders')
t2 = (45.78, 'ML', 'NLP')
t3 = t1 + t2
print(t3)
```

```
(23, 'python', 'coders', 45.78, 'ML', 'NLP')
```

Q2. Write a code to thrice the elements of tuple, t = (23, 'python', 'coders')

In [25]:
```
t = (23, 'python', 'coders')
print(t * 3)
```

```
(23, 'python', 'coders', 23, 'python', 'coders', 23, 'python', 'coders')
```

Q3. Write a code to print elemnts of tuple, t = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10) from iindex 3 to 9 with step size of 2.

In [26]:
```
t = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
print(t[3:9:2])
```

```
(4, 6, 8)
```

Q4. Write a code to check whether 'python' is in the tuple, t = (23, 'python', 'coders') or not.

In [27]:
```
t = (23, 'python', 'coders')
print('python' in t)
```

```
True
```

Q5. Write a code to find length of tuple, t = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)

In [28]:
```
t = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
print(len(t))
```

```
10
```

Q6. Write a code to print smallest & largest number from t = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)

In [29]:
```
t = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
print(min(t))
print(max(t))
```

```
1
10
```

Q7. Write a code to convert tuple, t = (23, 'python', 'coders') into list.

In [30]:
```
t = (23, 'python', 'coders')
t=list(t)
print(t)
```

```
[23, 'python', 'coders']
```

Q8. Write a code to find index of element 'coders' in tuple, t = (23, 'python', 'coders')

In [31]:
```
t = (23, 'python', 'coders')
t.index('coders')
```

Out[31]:
```
2
```

Q9. Write a code to sort the tuple, t = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10) in descending order

In [32]:
```
t = (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
print(sorted(t, reverse=True))
```

```
[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]
```

Q10. Write a code to count number of times 'ML' in tuple, t=('python', 'Ml', 'ML', 'DL', 'ML', 'NLP')

In [33]:
```
t=('python', 'Ml', 'ML', 'DL', 'ML', 'NLP')
t.count('ML')
```

Out[33]:
```
2
```

Sets, 2-Marks(2*10=20)

Q1. Write a code to add elements 'ML', 'DL' and 'NLP' one-by-one in a set, and print the set so formed.

In [34]:
```
# Create an empty set
my_set = set()

# Add elements to the set
my_set.add('ML')
my_set.add('DL')
my_set.add('NLP')
```

```
# Check the set
print(my_set)
```

```
{'DL', 'NLP', 'ML'}
```

Q2. Write a code to remove 'DL' from set my_set={'DL', 'NLP', 'ML'}

In [35]:
```
# Create a set
my_set={'DL', 'NLP', 'ML'}

# Remove an element from the set
my_set.remove('DL')

# Check the set
print(my_set)
```

```
{'NLP', 'ML'}
```

Q3. Write a code to check whether 'python' and 'NLP' are in set, my_set= {'DL', 'NLP', 'ML'} or not.

In [36]:
```
# Create a set
my_set={'DL', 'NLP', 'ML'}

# Check if an element is in the set
print('python' in my_set)
print('NLP' in my_set)
```

```
False
True
```

Q4. Write a code to find common elements from sets, set1 = {'python', 'ML', 'DL'} and set2 = {'python', 'NLP', 'DL', 'SQL'}

In [37]:
```
# Create two sets
set1 = {'python', 'ML', 'DL'}
set2 = {'python', 'NLP', 'DL', 'SQL'}

# Find the intersection of the two sets
intersection = set1.intersection(set2)

# Check the intersection
print(intersection)
```

```
{'DL', 'python'}
```

Q5. Write a code to find union of sets, set1 = {'python', 'ML', 'DL'} and set2 = {'python', 'NLP', 'DL', 'SQL'}

In [38]:
```
# Create two sets
set1 = {'python', 'ML', 'DL'}
set2 = {'python', 'NLP', 'DL', 'SQL'}

# Find the union of the two sets
union = set1.union(set2)

# Check the union
print(union)
```

```
{'python', 'ML', 'NLP', 'SQL', 'DL'}
```

Q6. Write a code to find those elements which are present in set1 = {'python', 'ML', 'DL'} but not in set2 = {'python', 'NLP', 'DL', 'SQL'}

In [39]:
```
# Create two sets
set1 = {'python', 'ML', 'DL'}
set2 = {'python', 'NLP', 'DL', 'SQL'}

# Find the difference between the two sets
diff = set1.difference(set2)

# Check the difference
print(diff)
```

```
{'ML'}
```

Q7. Write a code to find those elements which are present in set2 = {'python', 'NLP', 'DL', 'SQL'} but not in set1 = {'python', 'ML', 'DL'}

In [40]:
```
# Create two sets
set1 = {'python', 'ML', 'DL'}
set2 = {'python', 'NLP', 'DL', 'SQL'}

# Find the difference between the two sets
diff = set2.difference(set1)

# Check the difference
print(diff)
```

```
{'NLP', 'SQL'}
```

Q8. Write a code to find those elements which are present in set1 = {'python', 'ML', 'DL'} but not in set2 = {'python', 'NLP', 'DL', 'SQL'} & vice-versa.

In [41]:
```
# Create two sets
set1 = {'python', 'ML', 'DL'}
set2 = {'python', 'NLP', 'DL', 'SQL'}

# Find the symmetric difference between the two sets
sym_diff = set1.symmetric_difference(set2)

# Check the symmetric difference
print(sym_diff)
```

```
{'NLP', 'ML', 'SQL'}
```

Q9. Write a code to find the length of set, s = {'python', 'NLP', 'DL', 'SQL'}

In [42]:
```
# Create a set
s = {'python', 'NLP', 'DL', 'SQL'}

# Find the length of the set
length = len(s)
```

Q10. Write a code to clear all the elements of s = {'python', 'NLP', 'DL', 'SQL'}

In [43]:
```
# Create a set
, 'NLP', 'DL', 'SQL'}
```

```
# Clear all elements from the set
s.clear()

# Check the set
print(s)
```

```
set()
```

Dictionary, 2-Marks(2*10=20)

Q1. Write a code to check whether the key, 'sub6' is available in dictionary, my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'} or not

In [44]:
```
my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}

if "sub6" in my_dict:
    print("key exists")
else:
    print("key does not exist")
```

```
key does not exist
```

Q2. Write a code to remove 'sub3' from my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}

In [45]:
```
my_dict.pop("sub3")
print(my_dict)
```

```
{'sub1': 'python', 'sub2': 'ML', 'sub4': 'DL', 'sub5': 'NLP'}
```

Q3. Write a code to iterate over the keys of a dictionary, my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}

In [46]:
```
my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}
for key in my_dict.keys():
    print(key)
```

```
sub1
sub2
sub3
sub4
sub5
```

Q4. Write a code to obtain a list of all the values in dictionary, my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}

In [47]:
```
my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}
values_list = list(my_dict.values())
print(values_list)
```

```
['python', 'ML', 'SQL', 'DL', 'NLP']
```

Q5. Write a code to merge two dictionaries dict1 = {"a": 1, "b": 2} and dict2 = {"c": 3, "d": 4}


```
1, "b": 2}
```

```
dict2 = {"c": 3, "d": 4}
dict1.update(dict2)
print(dict1)
```

```
{'a': 1, 'b': 2, 'c': 3, 'd': 4}
```

Q6. Write a code to make copy of dictionary, my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}

In [49]:
```
my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}
my_dict_copy = my_dict.copy()
my_dict_copy
```

Out[49]:
```
{'sub1': 'python', 'sub2': 'ML', 'sub3': 'SQL', 'sub4': 'DL', 'sub5': 'NLP'}
```

Q7. Write a code to access 'dl' from my_dict = {1:'one' , 2:'two' , 'A': ['python', 'sql', 'ml'], 'B':{'course1':'dl', 'course2':'nlp'}}

In [50]:
```
my_dict = {1:'one' , 2:'two' , 'A':['python', 'sql', 'ml'], 'B':{'course1':'dl', 'course
my_dict['B']['course1']
```

Out[50]:
```
'dl'
```

Q8. Write a code to find cubes from numbers 11 to 20 using dictionary comprehension.

In [51]:
```
#cubes of numbers from 11 to 20
cubes = {i:i**3 for i in range(11, 21)}
cubes
```

Out[51]:
```
{11: 1331,
 12: 1728,
 13: 2197,
 14: 2744,
 15: 3375,
 16: 4096,
 17: 4913,
 18: 5832,
 19: 6859,
 20: 8000}
```

Q9. Write a code to clear all items of the dictionary, my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}

In [52]:
```
my_dict.clear()
my_dict
```

Out[52]:
```
{}
```

Q10. Write a code to delete dictionary, my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}

In [53]:
```
my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'}
del my_dict # Delete the dictionary object
my_dict
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_6984\1752588784.py in <module>
      1 my_dict = {'sub1':'python', 'sub2':'ML', 'sub3':'SQL', 'sub4':'DL', 'sub5':'NLP'
}
      2 del my_dict # Delete the dictionary object
----> 3 my_dict

NameError: name 'my_dict' is not defined
```

Operators, 1-Mark(1*5=5)

Q1. What will be the output of following code?

In [54]:
```
x = [1, 2, 3]
y = [1, 2, 3]
print(x is y)
```

```
False
```

Solution:

False

Q2. What will be the output of following code?

In [55]:
```
x = [1, 2, 3]
y = [1, 2, 3]
print(x == y)
```

```
True
```

Solution:

True

Q3. What will be the output of following code?

In [56]:
```
x = True
y = False
print(x and y)
```

```
False
```

Solution:

False

Q4. What will be the output of following code?

In [57]:
```
x = 5
print(x is 5)
```

```
True
```

```
<>:2: SyntaxWarning: "is" with a literal. Did you mean "=="?
<>:2: SyntaxWarning: "is" with a literal. Did you mean "=="?
C:\Users\madhu\AppData\Local\Temp\ipykernel_6984\3785439977.py:2: SyntaxWarning: "is" wi
th a literal. Did you mean "=="?
  print(x is 5)
```

Solution:

True

Q5. What will be the output of following code?

In [58]:
```
x = [1, 2, 3]
print(4 not in x)
```

```
True
```

Solution:

True

For-loop, 3-Mark(3*10=30)

Q1. Write a code to print squares of all the numbers from 21 to 30 using for-loop.

In [59]:
```
for i in range(21, 31):
    print(i**2)
```

```
441
484
529
576
625
676
729
784
841
900
```

Q2. Write a code to print only odd numbers from 31 to 50 using for-loop.

In [60]:
```
for i in range(31, 50, 2):
    print(i)
```

```
31
33
35
37
39
41
43
45
47
49
```


. code to print all the elements of list, post = ['data analyst', 'data

scientist', 'data engineer'] using for-loop.

In [61]:
```
post = ['data analyst', 'data scientist', 'data engineer']
for x in post:
    print(x)
```

```
data analyst
data scientist
data engineer
```

Q4. Write a code to print all the elements except 'ML' from list, my_list = ['data analyst', 'data scientist', 'data engineer', 'ML', 'python', 'DL'] using for-loop.

Hint : use continue statement

In [62]:
```
my_list = ['data analyst', 'data scientist', 'data engineer', 'ML', 'python', 'DL']
for x in my_list:
    if x == "ML":
        continue
    print(x)
```

```
data analyst
data scientist
data engineer
python
DL
```

Q5. Write a code to print all the elements from list, my_list = ['data analyst', 'data scientist', 'data engineer', 'ML', 'python', 'DL'] using for-loop but breakes the loop when 'python' occurs.

Hint : use break statement

In [63]:
```
my_list = ['data analyst', 'data scientist', 'data engineer', 'ML', 'python', 'DL']
for x in my_list:
    if x == "python":
        break
    print(x)
```

```
data analyst
data scientist
data engineer
ML
```

Q6. Write a code to print only even numbers from list, numbers = [24, 34, 65, 34, 77, 98, 67] using for-loop.

In [64]:
```
numbers = [24, 34, 65, 34, 77, 98, 67]
for num in numbers:
    if num % 2 == 0:
        print(num)
```

```
24
34
34
98
```

Q7. Write a code to print table of 34 using for-loop.

In [65]:
```
for i in range(1, 11):
    print(f'34 * {i} = {34*i}')
```

```
34 * 1 = 34
34 * 2 = 68
34 * 3 = 102
34 * 4 = 136
34 * 5 = 170
34 * 6 = 204
34 * 7 = 238
34 * 8 = 272
34 * 9 = 306
34 * 10 = 340
```

Q8. Use a for loop to calculate the sum of list, numbers = [56, 45, 89].

In [66]:
```
numbers = [56, 45, 89]
sum = 0
for number in numbers:
    sum += number
print(sum)
```

```
190
```

Q9. Use a for loop to remove all the vowels from string, string = "Hello, World!".

In [67]:
```
string = "Hello, World!"

vowels = "aeiouAEIOU"
result = ""
for char in string:
    if char not in vowels:
        result += char
print(result)
```

```
Hll, Wrld!
```

Q10. Use a for loop to count the number of occurrences of a specific character('l') in string, string = "Hello, World!"

In [68]:
```
string = "Hello, World!"
char = "l"
count = 0
for c in string:
    if c == char:
        count += 1
print(count)
```

```
3
```

While-loop, 3-Mark(3*10=30)

Q1. Use a while loop to print numbers from 0 to 50 with step-size of 5.

In [69]:
```
i=0
while i<=50:
    print(i)
```

```
0
5
10
15
20
25
30
35
40
45
50
```

Q2. Use while loop to print all the multiples of 3 from 1 to 30.

In [70]:
```
i=1
while i<=30:
    if(i%3)==0:
        print(i)
    i+=1
```

```
3
6
9
12
15
18
21
24
27
30
```

Q3. Use a while loop to print the numbers 11 to 20 in Python?

In [71]:
```
counter = 11
while counter <= 20:
    print(counter)
    counter += 1
```

```
11
12
13
14
15
16
17
18
19
20
```

Q4. Use a while loop to print the numbers in reverse order from a given number(say 10)?.

In [72]:
```
num = 10
while num >= 0:
    print(num)
    num -= 1
```

```
10
9
8
7
6
5
4
3
2
1
0
```

Q5. Write a code to find largest number from the list, numbers = [56, 42, 67, 43, 99, 87] using while loop.

In [73]:
```
numbers = [56, 42, 67, 43, 99, 87]
largest = numbers[0]
index = 1
while index < len(numbers):
    if numbers[index] > largest:
        largest = numbers[index]
    index += 1
print(largest)
```

```
99
```

Q6. Write a code to find sum of all the elements of the list, numbers = [56, 42, 67, 43, 99, 87] using while loop.

In [74]:
```
numbers = [10, 20, 30, 40, 50]
total = 0
index = 0
while index < len(numbers):
    total += numbers[index]
    index += 1
print(total)
```

```
150
```

Q7. Use while loop to print table of 25.

In [75]:
```
i=1
while i<=10:
    print(f'25 * {i} = {25*i}')
    i=i+1
```

```
25 * 1 = 25
25 * 2 = 50
25 * 3 = 75
25 * 4 = 100
25 * 5 = 125
25 * 6 = 150
25 * 7 = 175
25 * 8 = 200
25 * 9 = 225
25 * 10 = 250
```

Q8. Use while loop to keep a program running until the user enters a specific command('bye') to quit.

In [1]:
```
while True:
    user_input = input("Enter a command: ")
    if user_input == "bye":
        break
```

```
Enter a command: hi
Enter a command: hello
Enter a command: bye
```

Q9. Separate even and odd numbers from a list,numbers = [56, 42, 67, 43, 99, 87, 88] using while-loop and append them in two different lists num_even and num_odd.

In [2]:
```
numbers = [56, 42, 67, 43, 99, 87, 88]
num_even=[]
num_odd=[]

i = 0
while i < len(numbers):
    if numbers[i]%2==0:
        num_even.append(numbers[i])
    else:
        num_odd.append(numbers[i])
    i+= 1
print(num_even)
print(num_odd)
```

```
[56, 42, 88]
[67, 43, 99, 87]
```

Q10. Use while-loop to find factorial of 8.

In [3]:
```
number = 8
result = 1
i = 1
while i <= number:
    result *= i
    i += 1
print("Factorial of ", number, " is: ",result)
```

```
Factorial of  8  is:  40320
```

Functions, 3-Mark(3*10=30)

Q1. Write a function to reverse a string.

In [4]:
```
def reverse_string(string):
    return string[::-1]

reverse_string('hello')
```

Out[4]:
```
'olleh'
```

Q2. Write a function to check if a given year is a leap year or not.

In [5]:
```
def is_leap_year(year):
    if year % 4 == 0 and (year % 100 != 0 or year % 400 == 0):
        return True
```

```
    else:
        return False

#is_leap_year(1996)
#is_leap_year(2000)
is_leap_year(3000)
```

Out[5]:
```
False
```

Q3. Write a function to remove duplicates from a list.

In [6]:
```
def remove_duplicates(lst):
    return list(set(lst)) #set gives unique values

remove_duplicates([56, 45, 32, 32, 45])
```

Out[6]:
```
[56, 32, 45]
```

Q4. Write a function to calculate power of a number raised to other. E.g.- a^b.

In [7]:
```
def power(a, b):
    return(a**b)

power(3, 4)
```

Out[7]:
```
81
```

Q5. Write a function to calculate area and circumference of a circle.

In [8]:
```
def circle(radius):
    area = 22/7*(radius*radius)
    circumference = 22/7*(2*radius)
    print(f'Area of circle = {area}')
    print(f'Circumference of circle = {circumference}')

circle(7)
```

```
Area of circle = 154.0
Circumference of circle = 44.0
```

Q6. Write a function to calculate area and perimeter of a rectangle.

In [9]:
```
def rectangle(length, breadth):
    area = length*breadth
    perimeter = 2*(length+breadth)
    print(f'Area of circle = {area}')
    print(f'Circumference of circle = {perimeter}')

rectangle(10, 20)
```

```
Area of circle = 200
Circumference of circle = 60
```

Q7. Write a function to find the sum of the digits of an integer using a while loop.

```
(number):
```

```
    sum = 0
    while(number!=0):
        digit = number%10
        sum = sum+digit
        number = number//10
        print("Sum of digits is: ", sum)

summation(234)
```

```
Sum of digits is:  4
Sum of digits is:  7
Sum of digits is:  9
```

Q8. Write a function to display all integers within the range 10-30 whose sum of digits is an even number.

In [11]:
```
for i in range(10, 30):
    num = i
    sum = 0
    while(num!=0):
        digit = num%10
        sum = sum + digit
        num = num//10
    if(sum%2==0):
        print(i)
```

```
11
13
15
17
19
20
22
24
26
28
```

Q9. Write a function to check whether the given integer is a multiple of both 5 and 7.

In [12]:
```
def mul5_7(number):
    if((number%5==0)and(number%7==0)):
        print(number, "is a multiple of both 5 and 7")
    else:
        print(number, "is not a multiple of both 5 and 7")

mul5_7(35)
```

```
35 is a multiple of both 5 and 7
```

Q10. Write 4 different functions to make simple calculator(+, -, *, /).

In [14]:
```
num1 = float(input("Enter number 1 : "))
num2 = float(input("Enter number 2 : "))

def to_add():
    tot = num1 + num2
    print(f"{num1} + {num2} = {tot}")

def to_sub():
    sub = num1 - num2
    print(f"{num1} - {num2} = {sub}")
```

```
def to_mul():
    mul = num1 * num2
    print(f"{num1} x {num2} = {mul}")

def to_div():
    div = num1 / num2
    print(f"{num1} / {num2} = {div}")

operation = input("Choose an operation(+,-,*, /):")
if operation == "+":
    to_add()
elif operation == "-":
    to_sub()
elif operation == "*":
    to_mul()
elif operation == "/":
    to_div()
else:
    print(f"No operation for {operation} for now! sorry!")
```

```
Enter number 1 : 45
Enter number 2 : 55
Choose an operation(+,-,*, /):+
45.0 + 55.0 = 100.0
```

END OF DOCUMENT.