**Python Interview Questions (basics)**

*Q.1. How to convert a list into a comma-separated string?*

*Answer:* When we want to convert a list into a string, we can use the join() method which joins all the elements into one and returns as a string.

*weekdays = [‘sun’,’mon’,’tue’,’wed’,’thu’,’fri’,’sat’]*

*listAsString = ‘,’.join(weekdays)*

*print(listAsString)*

*output: sun,mon,tue,wed,thu,fri,sat*

*Q.2. How to convert a list into a tuple?*

*Answer:* By using Python function we can convert a list into a tuple. But we can’t change the list after turning it into tuple, because it becomes immutable.

*weekdays = [‘sun’,’mon’,’tue’,’wed’,’thu’,’fri’,’sat’]*

*listAsTuple = tuple(weekdays)*

*print(listAsTuple)*

*output: (‘sun’, ‘mon’, ‘tue’, ‘wed’, ‘thu’, ‘fri’, ‘sat’)*

*Q.3. How can you create empty NumPy array in Python?*

*Answer:* We can create Empty NumPy Array in two ways in Python.

*1. import numpy numpy.array([])*

*2. numpy.empty(shape=(0,0))*

*Q.4. Write a program to find the sum of the digits of a number in Python?*

*Answer: Python Program to find the sum of the digits of a number*

*n=int(input(“Enter a number:”)) tot=0*

*while(n>0):*

*dig=n%10*

*tot=tot+dig*

*n=n//10*

*print(“The total sum of digits is:”,tot)*

*Output:*

*Enter a number:1928*

*The total sum of digits is: 20*

*Q.5. Write a Python program to check if a string is a palindrome or not?*

*Answer:* Python Program to Check if a String is a Palindrome or Not:

*string=raw\_input(“Enter string:”)*

*if(string==string[::-1]):*

*print(“The string is a palindrome”)*

*else:*

*print(“The string isn’t a palindrome”)*

*Output:*

*Enter string:MALAYALAM*

*The string is a palindrome*

*Q.6. How will you print the sum of numbers starting from 1 to 100 (inclusive of both)?*

*Answer:* Python program to print the sum of numbers starting from 1 to 100.

*print sum(range(1,101))*

*#range() returns a list to the sum function containing*

*#all the numbers from 1 to 100. Please see that*

*#the range function does not include the end given (101 here).*

*print(sum(xrange(1, 101)))*

*#xrange() returns an iterator rather than a list*

*#which is less heavy in the memory.*

*Q.7. How do I convert between tuples and lists?*

*Answer:* The function tuple(seq) converts any sequence (actually, any iterable) into a tuple with the same items in the same order.

*For example, tuple([1, 2, 3]) yields (1, 2, 3) and tuple(‘abc’) yields (‘a’, ‘b’, ‘c’).*

*If the argument is a tuple, it does not make a copy but returns the same object, so it is cheap to call tuple() when you aren’t sure that an object is already a tuple.*

*The function list(seq) converts any sequence or iterable into a list with the same items in the same order.*

*For example, list((1, 2, 3)) yields [1, 2, 3] and list(‘abc’) yields [‘a’, ‘b’, ‘c’].*

*If the argument is a list, it makes a copy just like seq[:] would.*

*Q.8. Write a one-liner that will count the number of capital letters in a file. Your code should work even if the file is too big to fit in memory.*

*Answer:* This can be done in several ways, but considering the fact that we need a one-liner logic I suggest something like:

*with open(SOME\_LARGE\_FILE) as fh:*

*count = sum(character.isupper() for line in fh for character in line)*

*Q.9. How would you create an empty NumPy array?*

*Answer:* To create an empty array with NumPy, we have two options:

*a. Option 1 >>> import numpy >>> numpy.array([])*

*b. Option 2 >>> numpy.empty(shape=(0,0))*

*Q.10. How to add a value to a python array?*

*Answer:* Elements can be added to an array using the append(), extend() and insert (i,x) functions.

*a=arr.array('d', [1.1 , 2.1 ,3.1] )*

*a.append(3.4)*

*print(a)*

*a.extend([4.5,6.3,6.8])*

*print(a)*

*a.insert(2,3.8)*

*print(a)*

*Output:*

*array(‘d’, [1.1, 2.1, 3.1, 3.4])*

*array(‘d’, [1.1, 2.1, 3.1, 3.4, 4.5, 6.3, 6.8]) array(‘d’, [1.1, 2.1, 3.8, 3.1, 3.4, 4.5, 6.3, 6.8])*

*Q.11. How to remove values to a python array?*

*Answer:* Array elements can be removed using pop() or remove() method. The difference between these two functions is that the former returns the deleted value whereas the latter does not.

*a=arr.array('d', [1.1, 2.2, 3.8, 3.1, 3.7, 1.2, 4.6])*

*print(a.pop())*

*print(a.pop(3))*

*a.remove(1.1)*

*print(a)*

*Output:*

*4.6*

*3.1*

*array(‘d’, [2.2, 3.8, 3.7, 1.2])*

*Q.12. How will you read a random line in a file?*

*Answer:* We can read a random line in a file using a module named ‘random’.

*import random*

*def read\_random(fname):*

*lines = open(fname).read().splitlines()*

*return random.choice(lines)*

*print(read\_random (‘hello.txt’))*

*Q.13. Given a string, write a Python program to split strings on Uppercase characters.*

*Answer:* Program to split string using re.split():

*import re*

*# Initialising string*

*ini\_str = 'Example'*

*# Printing Initial string*

*print ("Initial String", ini\_str)*

*# Splitting on UpperCase using re*

*res\_list = [s for s in re.split("([A-Z][^A-Z]\*)", ini\_str) if s]*

*# Printing result*

*print("Resultant prefix", str(res\_list))*

*Q.14. Given a list of strings, write a Python program to convert the given list of strings into a space-separated string.*

*Answer:* Program to convert list into string using join()

*function: def convert(lst):*

*return ' '.join(lst)*

*# Driver code*

*lst = [‘word’, ‘song’]*

*print(convert(lst))*

*Q.15. Given a string and a boolean value, write a Python program to concatenate the string with a boolean value.*

*Answer:* Program to concat boolean using type casting:

*# Initialising string and boolean value*

*ini\_string = "Facts are"*

*value = True*

*# Concatenate using str*

*res = ini\_string +" "+str(value)*

*# Printing resultant string*

*print ("Resultant String : ", res)*

*Q.16. Write a program to delete all occurrences of character from a string.*

*Answer:* Program to delete a character using string function:

*# initializing string*

*test\_str = "DataScienceIsFun"*

*# initializing removal character*

*rem\_char = "e"*

*# printing original string*

*print("The original string is : " + str(test\_str))*

*# Using replace()*

*# Deleting all occurrences of character*

*res = test\_str.replace(rem\_char, "")*

*# printing result*

*print("The string after character deletion : " + str(res))*

*Q.17. Given a list of numbers and a variable K, where K is also a number, write a Python program using Numpy module, to find the number in a list which is closest to the given number K.*

*Answer:* Python program to find Closest number in a list:

*import numpy as np*

*def closest(lst, K):*

*lst = np.asarray(lst)*

*idx = (np.abs(lst - K)).argmin()*

*return lst[idx]*

*# Driver code*

*lst = [3.64, 5.2, 9.42, 9.35, 8.5, 8]*

*K = 9.1*

*print(closest(lst, K))*

*Q.18. Sometimes, while working with strings, we may have situations in which we might have more than 1 space between intermediate words in strings that are mostly unwanted. How will you remove them?*

*Answer:* Program Using split() + join()

*# initializing string*

*test\_str = "This is a Sample"*

*# printing original string*

*print("The original string is : " + test\_str)*

*# using split() + join()*

*# remove additional space from string*

*res = " ".join(test\_str.split())*

*# printing result*

*print("The strings after extra space removal : " + str(res))*

*Q.19. Given a list, write a Python program to convert the given list to dictionary such that all the odd elements become the key, and even number elements become the value.*

*Answer:* Python program to Convert a list to dictionary:

*def Convert(lst):*

*res\_dct = {lst[i]: lst[i + 1]*

*for i in range(0, len(lst), 2)}*

*return res\_dct*

*# Driver code*

*lst = ['a', 1, 'b', 2, 'c', 3]*

*print(Convert(lst))*

*Output:* *{'a': 1, 'b': 2, 'c': 3}*

*Q.20. Given a list of integers and an integer variable K, write a Python program to find all pairs in the list with given sum K.*

*Answer:* Python program to find all pairs in a list of integers with given sum:

*def findPairs(lst, K):*

*res = [] while lst:*

*num = lst.pop()*

*diff = K - num*

*if diff in lst:*

*res.append((diff, num))*

*res.reverse()*

*return res*

*# Driver code*

*lst = [1, 5, 3, 7, 9]*

*K = 12*

*print(findPairs(lst, K))*

*Output:* *[(5, 7), (3, 9)]*

*Q.21. While programming, sometimes, we just require a certain type of data and need to discard other. How will you Extract digits from given string?*

*Answer:* Program to extract digits from a string using re package:

*import re*

*# initializing string*

*test\_string = ‘1word2song3’*

*# printing original strings*

*print("The original string : " + test\_string)*

*# using re*

*# Extract digit string*

*res = re.sub("\D", "", test\_string)*

*# print result*

*print("The digits string is : " + str(res))*

*Output :*

*The original string : 1word2song3*

*The digits string is : 123*

*Q.22. Write a program to Split a string on last occurrence of delimiter.*

*Answer:* Python3 code to demonstrate Split on last occurrence of delimiter using rsplit()

*# initializing string*

*test\_string = "data, is, good, better, and best"*

*# printing original string*

*print("The original string : " + str(test\_string))*

*# using rsplit()*

*# Split on last occurrence of delimiter*

*res = test\_string.rsplit(', ', 1)*

*# print result*

*print("The post-split list at the last comma : " + str(res))*

*Output :*

*The original string : data, is, good, better, and best*

*The post-split list at the last comma : ['data, is, good, better', 'and best']*

*Q.23. How will you Get the string after occurrence of a given substring.*

*Answer:* The partition function can be used to perform this task in which we just return the part of partition occurring after the partition word.

*# initializing string*

*test\_string = "Python is best for data"*

*# initializing split word*

*spl\_word = 'best'*

*# printing original string*

*print("The original string : " + str(test\_string))*

*# printing split string*

*print("The split string : " + str(spl\_word))*

*# using partition()*

*# Get String after substring occurrence*

*res = test\_string.partition(spl\_word)[2]*

*# print result*

*print("String after the substring occurrence : " + res)*

*Q.24. Given a string, write a Python program to check if the string is a valid email address or not.*

*Answer:* Python program to validate an Email :

*import re*

*# Make a regular expression*

*# for validating an Email*

*regex = '^\w+([\.-]?\w+)*[*\*@\w+([\.-]?\w+)\*(\.\w{2,3})+$*](mailto:*@\w+(%5b\.-%5d?\w+)*(\.\w%7b2,3%7d)+$)*'*

*# Define a function for # for validating an Email*

*def check(email):*

*# pass the regular expression*

*# and the string in search() method*

*if(re.search(regex,email)):*

*print("Valid Email")*

*else:*

*print("Invalid Email")*

*# Driver Code*

*if \_\_name\_\_ == '\_\_main\_\_' :*

*# Enter the email*

*email = "example@gmail.com"*

*# calling run function check(email)*

*email = "example.com" check(email)*

*Q.25. Sometimes, during the string manipulation, we are into a problem where we need to pad or add leading zeroes to the string as per the requirements. How will you achieve this?*

*Answer:* Program to add leading zeros:

*# initializing string*

*test\_string = ‘GAP’*

*# printing original string*

*print("The original string : " + str(test\_string))*

*# No. of zeros required*

*N = 4*

*# using rjust()*

*# adding leading zero*

*res = test\_string.rjust(N + len(test\_string), '0')*

*# print result*

*print("The string after adding leading zeros : " + str(res))*

*The original string : GAP*

*The string after adding leading zeros : 0000GAP*