ASCII/UNICODE VALUES

According the te to Amarican standard coding information interchange each and every alphabet is asscociated with its unqiue valyue of integer and that unique value is known as Ascii Value.

In [ ]:
```
#UNQ
Each and every alphabet is associated with its integer unqiue value is known as ascii value
A=65 --> 65+25+90
B=66
C=67
E=69

Z=90

a=97
b=98
c=99
d=100
e=101
Computer understand which type of language --> Binary form 0-1
If we want to write any code --> English(for each and every alphabet we have just unique value for that
                                         alphabet)
```

In ord function you need to give only a single character

if you give more than one character then it will give you an error

print(ord('ABC')) #-- return the integr associate with the given alaphabet print(ord("c"))

In [58]:
```
ord('L')
```

Out[58]:
```
76
```

In [24]:
```
chr(1114111)#-->
```

Out[24]:
```
'\U0010ffff'
```

In [29]:
```
#Write a python program to print all Ascii values from 65 to 122

for i in range(65,123):
    print(chr(i),end=" ,")
```

```
A ,B ,C ,D ,E ,F ,G ,H ,I ,J ,K ,L ,M ,N ,O ,P ,Q ,R ,S ,T ,U ,V ,W ,X ,Y ,Z ,[ ,\ ,] ,^ ,_ ,` ,a ,b ,c ,d ,e ,f ,g ,h ,i ,j ,k ,l ,m ,n ,o ,p ,q ,
r ,s ,t ,u ,v ,w ,x ,y ,z ,
```

In [12]:
```
list=[]
for i in range(97,123):
list
```

Out[12]:
```
['a',
 'b',
 'c',
 'd',
 'e',
 'f',
 'g',
 'h',
 'i',
 'j',
 'k',
 'l',
 'm',
 'n',
 'o',
 'p',
 'q',
 'r',
 's',
 't',
 'u',
 'v',
 'w',
 'x',
 'y',
 'z']
```

In [ ]:
```
isalpha() -->return True if all character of the string are alphabet else return False
```

In [33]:
```
x="789dfgh"
x.isalpha()
```

Out[33]:
```
False
```

In [37]:
```
#sorted()--> return the sorted string based on its unicode or ascii values
x="abcdeABCDEFzdef"
sorted(x)#-->return result in form of list only
```

Out[37]:
```
['A', 'B', 'C', 'D', 'E', 'F', 'a', 'b', 'c', 'd', 'd', 'e', 'e', 'f', 'z']
```

In [36]:
```

```

Out[36]:
```
False
```

In [ ]:
```
#Palindrome String check
ABCDCBA --> ORGINAL STRING
ABCDCBA --> REVERSE STRING
ORGINAL == REVERSE STRING
PALINDROME STRING
MADAM #PEEP #MALYALAM #NOON
```

In [51]:
```
String_1=input("Enter Original String: ").lower()
Reverse_string_1 = String_1[::-1]
if String_1==Reverse_string_1:
    print("Palindrome String")
else:
    print("Not Palindrome")
```

```
Enter Original String: madaM
Palindrome String
```

In [57]:
```
#Anagram or Not   exmaple--> Listen --> siLent , RACEs-->CAREs
string_1=input().upper()
string_2=input().upper()
if sorted(string_1)==sorted(string_2):
    print("It is an anagram string")
else:
    print("It is not an anaagram string")

#Leints
#Leints
```

```
manager
manage
It is not an anaagram string
```

In [ ]:
```
#Sort on the basis of first alphabet followed by digits
Example:X = B4A1D3
OUTPUT: ABD134
```

#What is list? --> if we want to store a group of dissimilar elements as a single entity then we should go for list. --> Indexing is very important incase of list. Because we will differentiate elements based on index value. ---> Duplicates are allowed in list --->List is dynamic in nature that means we can add or delete elements from the list(Mutable)

In [63]:
```
#Create alist object
list=[]
print(type(list))
#Add any element at last -->append() --> at element at last
list.append(10)
print(list)
list.append(20)
print(list)
list[1]=200
print(list)
```

```
<class 'list'>
[10]
[10, 20]
[10, 200]
```

In [ ]:
```
#Accessing the element in list
1.indexing-->return the element asscicated with that index
2.Slicing --> substring(part) of the string
```

In [67]:
```
#Indexing => List_Name[Index_value]
#Slicing =>List_Name[Starting_inddex:ending_index_step]
x=[10,20,30,40,50]
#print(x[1]) #20
#print(x[5])  #error
#print(x[80]) #error
print(x[::1])  #full list
print(x[2:9])   #30,40,50
print(x[::-1])  #reverse list
print(x[::-2])   #50,30,10
print(x[10::9])
```

```
[10, 20, 30, 40, 50]
[30, 40, 50]
[50, 40, 30, 20, 10]
[50, 30, 10]
[]
```

In [73]:
```
#Deletion of Element-->pop()--> it will delete the value and return the deleted value
x=[10,20,30]
print(x.pop(100)) #-->30
print(x) #-->10,20
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
~\AppData\Local\Temp/ipykernel_14884/1566853879.py in <module>
      1 #Deletion of Element-->pop()--> it will delete the value and return the deleted value
      2 x=[10,20,30]
----> 3 print(x.pop(100)) #-->30
      4 print(x) #-->10,20

IndexError: pop index out of range
```

In [ ]:
```

```

In [5]:
```
#153-->  1**3+5**3+3**3=153
a=(1,2,3)
b=(1,2,3)
id(a)==id(b)
```

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

In [ ]:
```

```