match-case

In python to use switch case statement, in version 3.10 match—case statement is added The syntax is as follows

| match-case can be used with numbers | num=10  match num:  case 10:  print("num is 10")  case 20:  print("num is 20")  case others: # this is default case  print("num is other than 10 and 20") |
| --- | --- |
| Match case can be used with strings In a case statement, we can use multiple values to execute same statement for different values of variable , separate the values by | | color="yellow"  match color:  case "Red"|"yellow": #this is for multiple values  print("color is either red or yellow") case "green":  print("the color is green")  case \_: #this is default case  print("The color is blue") |

Functions

1. In python for writing functions we use def keyword

2. In function we may pass default values to the parameter

3. We may use keyword arguments also in the function, if we use keyword arguments then the position of the argument will not be considered

| Function without default parameter,  So the parameter is  mandatory parameter | #calculate factorial of a number  def factorial(num):  fact=1  for i in range(2,num+1):  fact=fact\*i  return fact |
| --- | --- |
| Function with default value,  In f2 function a is  mandatory parameter and b and c are optional parameter | def f2(a,b=20,c=30):  print(a,b,c)    f2(10) # a=10, b=20, c=30  f2(12,25) #a=12, b=25, c=30  f2(12,13,14) #a=12,b=13,c=14 |
| If in the function you start assigning default values then all parameters on the right side should have default parameters, | def f2(a,b=20,c=30):  print(a,b,c)    f2(10) # a=10, b=20, c=30  f2(12,25) #a=12, b=25, c=30  f2(12,13,14) #a=12,b=13,c=14 |

| Otherwise, it will give error | def f2(a=10,b,c=30): #this is error  print(a,b,c)    f2(10) # a=10, b=20, c=30  f2(12,25) #a=12, b=25, c=30  f2(12,13,14) #a=12,b=13,c=14 |
| --- | --- |
| While calling function you may use keyword  arguments, when you don’t want to follow the position | def f2(a,b,c=30):  print(a,b,c)    f2(b=20,a=10) # a=10, b=20, c=30  f2(c=12,a=13,b=14) #a=13,b=14,c=12 |
| When a function calls itself, then it is called as recursive function,  In recursive functions find the terminating condition, and also find the  recursive definition | def factorial(num):  if num==1: #terminating condition  return 1  return num\* factorial(num-1) #recurssive definition |

Builtin functions

Numbers

| To find the binary representation of the number | bin(num) |
| --- | --- |
| To find decimal number for given binary representation or to convert string into integer | int(0b101010)  int(‘20’)  int(input(“enter number”)) |
| To convert string into float | float(’30.5’)  float(input(“enter price”)) |
| To round the number | round(3.5678,2)=3.57  round(3.56789)=4  round(3.45678,2)=3.46  round(3.42156,2)=3.42 |
| To conver -ve number into + ve | abs(-23)=23 |
| Most of the functions used with numbers are available in math library, to use those function, use import math  To see the list of all the functions in math module  dir(math)  to get help of one of the function in math library  help(math.sqrt) | import math  dir(math)  help(math.sqrt) |
|  | import math  math.ceil(3.12)=4  math.ceil(3.56)=4  math.floor(3.12)=3  math.floor(3.56)=3  math.sqrt(4)=2 |

|  | math.trunc(-2.56)=-2  math.floor(-2.56)=-3 |
| --- | --- |

String functions

When we use string in python , it is treated as list of characters

| T | H | I | S |  | I | S |  | S | T | R | I | N | G |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 |  | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| -14 | -13 | -12 | -11 | -10 | -9 | -8 | -7 | -6 | -5 | -4 | -3 | -2 | -1 |

| Str=TTHIS IS  STRING” |  |  |
| --- | --- | --- |
| Str[-1] |  | G |
| Str[5] or str[-9] |  | I |
| Str[3:9] | To find characters from 3rd position to 8 th position, 9 is excluded | S IS S |
| Str[2:] | Start from 2 to end | IS IS STRING |
| Str[:7] | Start from 0 upto 6 th position | THIS IS |
| Str[2:11:2] | Start with 2 nd index position , every time increase the position by 2, upto 10 th index position | I SSR |
| Str[-5:2:-1] | Since we are reading the string from right to left, step has tobe -ve | TS SI S |
| Str[-11:10] |  | S IS ST |
| Str[::-1] | Reverse the string | GNIRTS SI SIHT |
| Str[0::-1] |  |  |

functions of string

| s.upper() | convert string into upprecase |
| --- | --- |
| s.lower() | convert the string into lowercase |
| s.startswith(“xxxx”) | return true if the string s starts with xxxx otherwise false |
| s.endswith(“xxxx”) | return true if the string s ends with xxxx otherwise false |
| s.find(substr[,start,end]) | the index position of the first occurrence of the given substring if found, otherwise it returns -1 |
| s.rfind(substr[,start,end]) | the index position of the last occurrence of the given substring if found, otherwise it returns -1 |
| s.index(substr[,start,end]) | the index position of the first occurrence of the given substring if found, otherwise it returns exception |
| s.rindex(substr[,start,end]) | the index position of the last occurrence of the given substring if found, otherwise it returns exception |
| s.strip(set of characters) | It will remove all occurrences of characters from both sides of the string |
| s.lstrip(set of characters) | It will remove all occurrences of characters from left sides of the string |
| s.rstrip(set of characters) | It will remove all occurrences of characters from right sides of the string |
| len(s) | to find the length of the string |

| s.isdigit() | It will check whether all the characters in the string are digits |
| --- | --- |
| s.isdecimal() | It will check whether all the characters in the string are decimal number |
| s.isnumberic() | It will check whether all the characters in the string are digits |
| s.isalpha() | It will check whether all the characters in the string are alphabets |
| s.isalnum() | It will check whether all the characters in the string are either digits or alphabets |
| s.count(“cat”) | it will return count of the number of occurrences of string cat |
|  |  |

**String**

**Type Example Python Python Python**

Base 10

| **()** |
| --- |

| **.isdecimal** |
| --- |

| **digit()** |
| --- |

| **.is** |
| --- |

| **.isnumeric()** |
| --- |

Number s

Fraction s and

Supersc ripts

Roman Numeral s

| '0123' |
| --- |

| '⅔', '2²' |
| --- |

| 'ↁ' |
| --- |

True True True False True True False False True

In python the string , numbers and Boolean are immutable objects

a=10 b=10 c=a

all will point to same memory in the heap are

s1=”Rajan” s2=”Rajan s3=s1

all will point to same object

to check whether they point to same object we may use either id function or is operator.

| a=10  b=10  c=a  print(id(a),id(b),id(c))  print(a is b)  a=15  print(id(a),id(b),id(c)) | s1="Rajan"  s2=s1  s3="Rajan"  print(id(s1),id(s2),id(s3))  print("Rajan" is s2)  s1="Revati"  print(id(s1),id(s2),id(s3)) |
| --- | --- |

| d=int(input("enetr number"))  print(d,id(d)) | s4=input("enter name") #even if we enter #Rajan the location will be different when we #accept string using input function  print(s4,id(s4)) |
| --- | --- |