|  |
| --- |
| # Method Overloading # There is no method overloading in python, because of datatype # method overloading will call which has last precedence  class Test:   def d1(self, a, b, c):  print(a, b, c)   def d1(self, a, b):  print(a, b)  t1 = Test() t1.d1(10,20)  10 20 |

|  |
| --- |
| # No Method Overloading in Python class Test:   # For all datatype a single method  def d1(self, a, b):  print(a, b)  t = Test() t.d1(5, 10) # 5 10 t.d1(5.0,10.0) # 5.0 10.0 t.d1('Sai Kiran', 'Sai Ram') # Sai Kiran Sai Ram t.d1(5j, 10j) # 5j 10j |

|  |
| --- |
| The multimethod are also called multiple dispatch: Multiple dispatch or multi methods is the feature of some object-oriented programming languages in which a function or method can be dynamically dispatched based on the run time (dynamic) type of more than one of its arguments  from multipledispatch import dispatch  @dispatch(int, int) def d1(a, b):  print(a, b) @dispatch(int, int, int) def d1(a, b, c):  print(a,b,c) @dispatch(float, float, float) def d1(a, b, c):  print(a, b, c)  d1(5,10) # 5 10 d1(5,10,15) # 5 10 15 d1(5.0, 10.0, 15.0) # 5.0 10.0 15.0 |