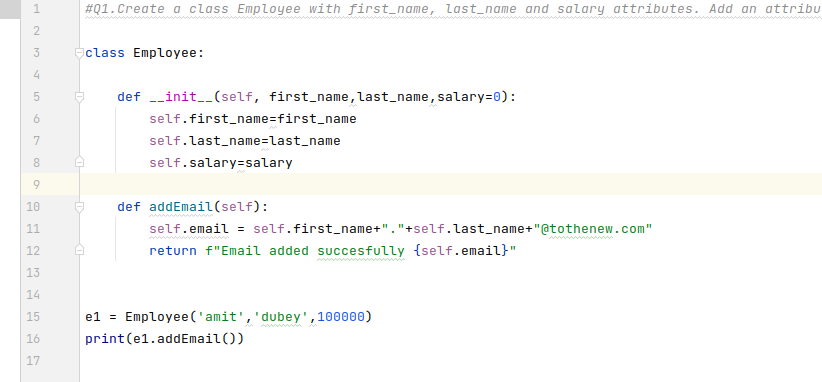
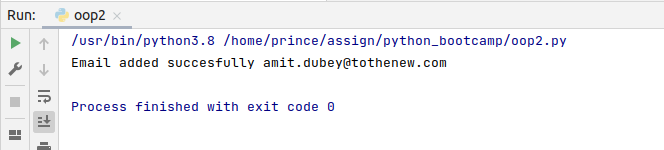
Q1.Create a class Employee with first\_name, last\_name and salary attributes. Add an attribute email\_Id which should always be [first\_name.last\_name@tothenew.com](mailto:first_name.last_name@tothenew.com).

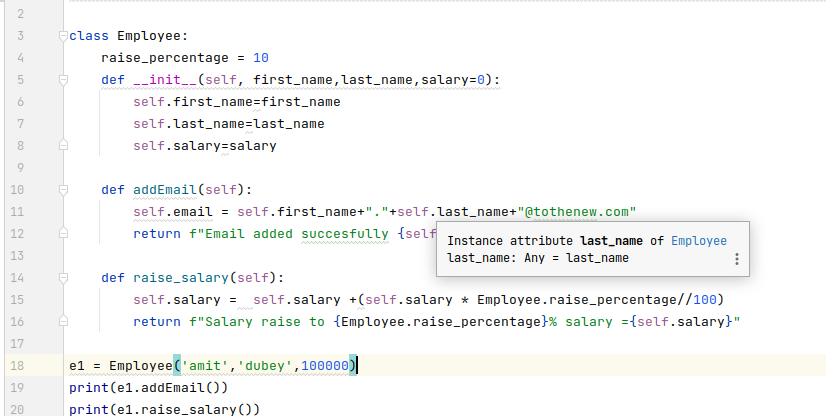
Soln

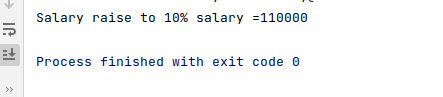




Q2.Define a class variable raise\_percentage = 10 and a function(raise\_salary) to raise an employee's salary. For example if raise % is 10 and salary of the empoyee is 10IK, after raise\_salary on the employee object should set salary of the employee to 11K (10K + 10% of 10K).

Soln.

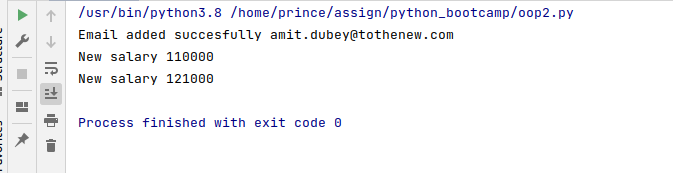




Q3.Create a method in the class raise\_percentage which increases the raise\_percentage by given number. Call riase\_percentage(5) and then call raise\_salary on an employee object and print the results.

Soln.

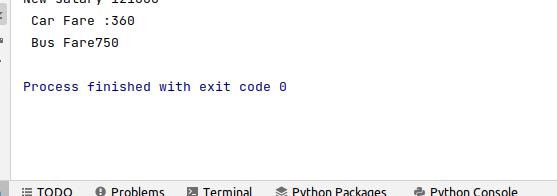




Q4.Create a class Vehicle with make, mileage and capacity attributes and a method to calculate fare(get\_fare). Ensure that get\_fare does not contain any logic and sub classes must override in order to call the get\_fare method on its object. Create two subclasses of Vehicle Car and Bus. Use base class init to initialise make, mileage and capacity in both the subclasses.

Soln.

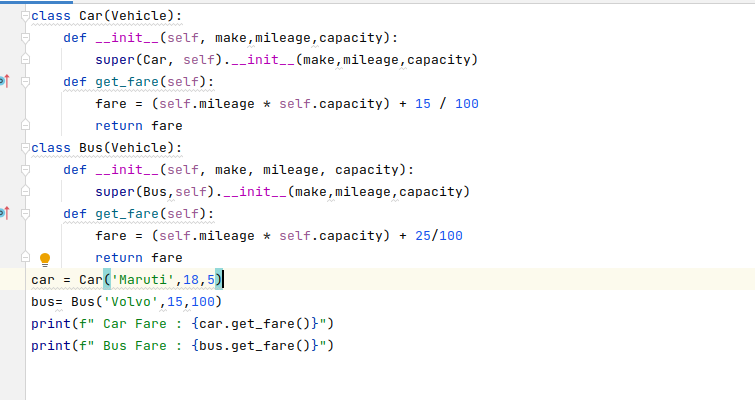


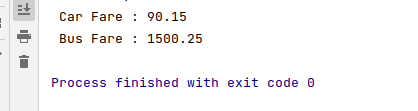


Q5.Override get\_fare methods in Car and Bus class to calculate fare as per the bello rule

fare = (mileage \* capacity) + 25% if Bus 15 % in case of Car

Soln.





Q6.Create a class Point with two instance variables x and y. Implement addition functionality on Point objects so that two Point objects can be added using + operator.

p1 = Point(3,4)

p2 = Point(2,6)

p3 = p1 + p2

print(p3) # should print (5, 10)

NOTE: adding two points returns another Point object

Soln.



