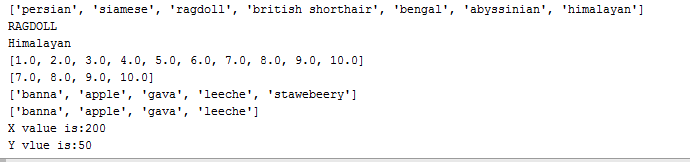
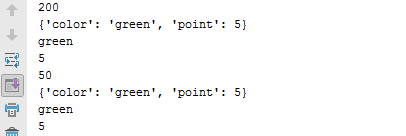
cats = [**'persian'**, **'siamese'**, **'ragdoll'**, **'british shorthair'**, **'bengal'**, **'abyssinian'**, **'himalayan'**]  
print(cats)  
**for** cat **in** cats:  
 **if** cat ==**'ragdoll'**:  
 print(cat.upper())  
**else**:  
 print(cat.title())  
  
squares = []  
**for** value **in** range(1, 11):  
 square = (value \*\* 2)\*\*0.5  
 squares.append(square)  
print(squares)  
print(squares[-4:])  
  
fruit=[**'banna'**,**'apple'**,**'gava'**,**'leeche'**]  
frien\_fr=fruit[:]  
fruit.append(**'stawebeery'**)  
print(fruit)  
print(frien\_fr)  
  
dimension=(200,50)  
print(**"X value is:"**+str(dimension[0]))  
print(**"Y vlue is:"**+str(dimension[1]))  
**for** dimension **in** dimension:  
 print(dimension)  
  
 alien\_o={**'color'**:**'green'**,**'point'**: 5}  
 print(alien\_o)  
 print(alien\_o[**'color'**])  
 print(alien\_o[**'point'**])





FUNCTION IN PHYTON

def sumProblem(x, y):  
 sum = x + y  
 sentence = 'The sum of {} and {} is {}.'.format(x, y, sum)  
 print(sentence)  
  
def main():  
 sumProblem(2, 3)  
 sumProblem(1234567890123, 535790269358)  
 a = int(input("Enter an integer: "))  
 b = int(input("Enter another integer: "))  
 sumProblem(a, b)  
  
main()

def fact(n):  
 if n==0:  
 return 1  
 return n\*fact(n-1)  
def main():  
 a = int(input("Enter the number of factorial:"))  
 print("factorial is:",fact(a))  
main()

def fob(i):  
 if i==0 or i==1 or i==2:  
 return i  
 else :  
 return fob(i-1)+fob(i-2)  
def main():  
 a = int(input("Enter the number:"))  
 print("Fobonicii series is:", fob(a))  
main()

def area(x,y):  
 return x\*x\*y  
def main():  
 c = int(input("Enter the radius of circle:"))  
 print("Area of the circle is:",area(3.142 ,c))  
main()

def cal(x ,y):  
 total = x + y  
 total = x - y  
 total = x \* y  
 total = x / y  
 return total  
def main():  
 a = int(input("Enter the 1st value:"))  
 b = int(input("Enter the 2nd value:"))  
 print("the total calculation is:", cal(a ,b))  
main()

def fact(n,k):  
 if n==0 or k==0:  
 return 1  
 return (n\*fact(n-1, 1))/(k\*fact(1, k-1))  
def main():  
 a = int(input("Enter the total number of combination:"))  
 b = int(input("Enter the number of selection:"))  
 print("combination is:",fact(a ,b))  
main()

import string  
class khan():  
 def setdata(self,m,n):  
 a = string(input("Enter the name of car"))  
 b = int(input("Enter the year of car"))  
 print("Name of car is:", a)  
 print("year of the car:", b)  
 def main(self):  
 k1 = khan()  
 print(k1.setdata())

class Car():  
 *"""A simple attempt to represent a car."""* def \_\_init\_\_(self, make, model, year):  
 *"""Initialize attributes to describe a car."""* self.make = make  
 self.model = model  
 self.year = year  
  
 def get\_descriptive\_name(self):  
 *"""Return a neatly formatted descriptive name."""* long\_name = str(self.year) + ' ' + self.make + ' ' + self.model  
 return long\_name.title()  
  
  
my\_new\_car = Car('audi', 'a4', 2016)  
print(my\_new\_car.get\_descriptive\_name())

from chatterbot import ChatBot  
from chatterbot.trainers import ListTrainer  
import os  
  
bot = ChatBot('Bot')  
bot.set\_trainer(ListTrainer)  
  
for files in os.listdir('C:/Users\Me\Documents\chatterbot-corpus-master\chatterbot\_corpus\data\english')  
  
 data =open('C:/Users\Me\Documents\chatterbot-corpus-master\chatterbot\_corpus\data\english'+ files , ).readline()  
  
bot.train(data)  
  
while True:  
 message = input("you:")  
 if message.strip()!='Bye':  
 reply = bot.get\_response(message)  
 print('ChatBot :', reply)  
 if message.strip() == 'Bye':  
 print ('ChatBot: Bye')  
 break