```
#1.CHEF DEVELOPER CODE

chef=input("Chose a Language 1 2:")

if chef=='':
    chef=0
if int(chef)==1:
    print(chef,"\nA1B1")
elif int(chef)==2:
    print(chef,"\nA2B2")
else:
    print("0","\nA1B1A2B2")


#2 DATE DIFFERENCE
class datecomp:
    def __init__(self, d1,m1,y1):
        self.d1 = d1
        self.m1=m1
        self.y1=y1

    def __lt__(self,U ):
        if(self.d1 < U.d1) & (self.m1 < U.m1 ) &  (self.y1 == U.y1 ):
            return "object_1 is less than object_2"
        elif(self.d1 < U.d1) & (self.m1 < U.m1 ) &  (self.y1< U.y1 ):
            return "object_1 is less than object_2"
        elif(self.d1 > U.d1) & (self.m1 < U.m1 ) &  (self.y1 < U.y1 ):
            return "object_1 is less than object_2"
        else:
            return "object_2 is less than object_1"

obj1=datecomp(3,1,2019)
obj2=datecomp(22,11,2020)
print(obj1<obj2)

#3 DISTANCE CALCULATOR
class calcdist:
    def __init__(self, km,mt):
        self.km = km
        self.mt=mt
    def __add__(self,u):
        okm=self.km+u.km
        omt=self.mt+u.mt
        if omt>=1000:
```

```
            print("Total",okm+(omt//1000),"KM",(omt%1000),"Meters")
        else:
            print ("Total",okm,"KM",omt,"Meters")
    def __sub__(self,u):

        okm=self.km-u.km
        omt=self.mt-u.mt
        print(omt)
        if omt>=1000:
          print("Difference",abs(okm-(omt//1000)),"KM",(omt%1000),"Meters")
        else:
          print("Difference",abs(okm),"KM",omt,"Meters")

    def __mul__(self,u):

        okm=self.km*u.km
        omt=self.mt*u.mt

        if omt>=1000:
          print("Multiply",abs(okm+(omt//1000)),"KM",(omt%1000),"Meters")
        else:
          print("Multiply",abs(okm),"KM",omt,"Meters")
    def __truediv__(self,u):

        okm=self.km/u.km
        omt=self.mt/u.mt

        if omt>=1000:
            print("Devide",abs(okm+(omt//1000)),"KM",(omt%1000),"Meters")
        else:
            print("Devide",abs(okm),"KM",omt,"Meters")
d1=calcdist(3, 20)
d2=calcdist(6,2000)
print(d1+d2)
print(d2-d1)
print(d2*d1)
print(d2/d1)

#4 INHERETANCE

class BOX:
    def __init__(self,l,b,d):
        self.l=l
        self.b=b
        self.d=d

class ColorBox(BOX):

    def __init__(self,l,b,d,color):
        super().__init__(l, b, d)
        self.color=color
    # def volume(self):
    #    print(self.l)

class ColorWeightBox(ColorBox):

    def __init__(self,l,b,d,color,weight):
```

```
        super().__init__(l, b,d,color)
        self.weight=weight
    def volume(self):
       print(self.color,"Color Box Volume is",self.l*self.b*self.d,"Weight is",self.weight)
obj1=ColorWeightBox(10,10,1,"red",20)
obj1.volume()
```