#Car => id,model,vendor,Engine,hasSpoiler(bu mashina aid ozellikdi)

#Ship => id,model,vendor,Engine,hasSail(sail - yelkən)

#Airplane => id,model,vendor,Engine,engineCount,passengersCapacity

#Engine=>engine\_no,company,volume

#=> ishare klas oldugunu bildirir

#Bu klaslar arasinda Inheritance elaqesi yaradilsin,

#her bir klasda show methodu yazilsin

#VehicleDepo=[cars,ships,airplanes]

#def addCar(car):

# cars.append(car)

class Engine:

def \_\_init\_\_(self, engine\_no,company,volume):

self.set\_engine\_no(engine\_no)

self.set\_company(company)

self.set\_volume(volume)

def set\_engine\_no(self, engine\_no):

if (engine\_no != ""):

self.\_\_engine\_no = engine\_no

else:

raise Exception("Engine number must be filled in!")

def set\_company(self, company):

if (company != ""):

self.\_\_company = company

else:

raise Exception("Company must be filled in!")

def set\_volume(self, volume):

if (int(volume) > 0):

self.\_\_volume = volume

else:

raise Exception("Value must be greater than zero!")

if (volume != ""):

self.\_\_volume = volume

else:

raise Exception("Volume must be filled in!")

def get\_engine\_no(self):

return self.\_\_engine\_no

def get\_company(self):

return self.\_\_company

def get\_volume(self):

return self.\_\_volume

def show\_info\_of\_engine(self):

print("Engine number : ",self.get\_engine\_no())

print("Company of engine : ",self.get\_company())

print("Volume of engine : ",self.get\_volume())

class Car(Engine):

cars=[]

def \_\_init\_\_(self, id, model, vendor, has\_spoiler, engine\_no, company, volume):

super().\_\_init\_\_(engine\_no, company, volume)

self.set\_id(id)

self.set\_model(model)

self.set\_vendor(vendor)

self.set\_has\_spoiler(has\_spoiler)

def set\_id(self, id):

if (id != ""):

self.\_\_id = id

else:

raise Exception("Vehicle Identification Number (ID) must be filled in!")

def set\_model(self, model):

if (model != ""):

self.\_\_model = model

else:

raise Exception("Model must be filled in!")

def set\_vendor(self, vendor):

if (vendor != ""):

self.\_\_vendor = vendor

else:

raise Exception("Vendor must be filled in!")

def set\_has\_spoiler(self, has\_spoiler):

if (has\_spoiler != ""):

self.\_\_has\_spoiler = has\_spoiler

else:

raise Exception("\"Does car have a spoiler?\" must be filled in!")

def get\_id(self):

return self.\_\_id

def get\_model(self):

return self.\_\_model

def get\_vendor(self):

return self.\_\_vendor

def get\_has\_spoiler(self):

return self.\_\_has\_spoiler

def add\_car(self,car):

cars.append(car)

def show\_vehicle(self):

print("==================================================================")

print("Vehicle Identification Number (ID) of the car : ",self.get\_id())

print("Model of the car : ",self.get\_model())

print("Vendor of the car : ",self.get\_vendor())

print("Does car have a spoiler? : ",self.get\_has\_spoiler())

super().show\_info\_of\_engine()

print("==================================================================")

class Ship(Engine):

def \_\_init\_\_(self, id, model, vendor, has\_sail, engine\_no, company, volume):

super().\_\_init\_\_(engine\_no, company, volume)

self.set\_id(id)

self.set\_model(model)

self.set\_vendor(vendor)

self.set\_has\_sail(has\_sail)

def set\_id(self, id):

if (id != ""):

self.\_\_id = id

else:

raise Exception("Vehicle Identification Number (ID) must be filled in!")

def set\_model(self, model):

if (model != ""):

self.\_\_model = model

else:

raise Exception("Model must be filled in!")

def set\_vendor(self, vendor):

if (vendor != ""):

self.\_\_vendor = vendor

else:

raise Exception("Vendor must be filled in!")

def set\_has\_sail(self, has\_sail):

if (has\_sail != ""):

self.\_\_has\_sail = has\_sail

else:

raise Exception("\"Does car have a sail?\" must be filled in!")

def get\_id(self):

return self.\_\_id

def get\_model(self):

return self.\_\_model

def get\_vendor(self):

return self.\_\_vendor

def get\_has\_sail(self):

return self.\_\_has\_sail

def show\_vehicle(self):

print("==================================================================")

print("Vehicle Identification Number (ID) of the ship : ",self.get\_id())

print("Model of the ship : ",self.get\_model())

print("Vendor of the ship : ",self.get\_vendor())

print("Does ship have a sail? : ",self.get\_has\_sail())

super().show\_info\_of\_engine()

print("==================================================================")

class Plane(Engine):

def \_\_init\_\_(self, id, model, vendor, engine\_count, passengers\_capacity, engine\_no, company, volume):

super().\_\_init\_\_(engine\_no, company, volume)

self.set\_id(id)

self.set\_model(model)

self.set\_vendor(vendor)

self.set\_engine\_count(engine\_count)

self.set\_passengers\_capacity(passengers\_capacity)

def set\_id(self, id):

if (id != " "):

self.\_\_id = id

else:

raise Exception("Vehicle Identification Number (ID) must be filled in!")

def set\_model(self, model):

if (model != ""):

self.\_\_model = model

else:

raise Exception("Model must be filled in!")

def set\_vendor(self, vendor):

if (vendor != ""):

self.\_\_vendor = vendor

else:

raise Exception("Vendor must be filled in!")

def set\_engine\_count(self, engine\_count):

if (engine\_count > 0):

self.\_\_engine\_count = engine\_count

else:

raise Exception("Engine count can't be less thn zero or zero!")

def set\_passengers\_capacity(self, passengers\_capacity):

if (passengers\_capacity > 0):

self.\_\_passengers\_capacity = passengers\_capacity

else:

raise Exception("Passenger capacity can't be less thn zero or zero!")

def get\_id(self):

return self.\_\_id

def get\_model(self):

return self.\_\_model

def get\_vendor(self):

return self.\_\_vendor

def get\_engine\_count(self):

return self.\_\_engine\_count

def get\_passengers\_capacity(self):

return self.\_\_passengers\_capacity

def show\_vehicle(self):

print("==================================================================")

print("Vehicle Identification Number (ID) of the plane : ",self.get\_id())

print("Model of the plane : ",self.get\_model())

print("Vendor of the plane : ",self.get\_vendor())

print("Engine count : ",self.get\_engine\_count())

print("Passengers capacity : ",self.get\_passengers\_capacity())

super().show\_info\_of\_engine()

print("==================================================================")

#Task 1

#try:

# car1=Car("PJ12345U123456P","Toyota","John Johnlu","Yes",2488152,"Blackstone",3.5)

# car1.show\_vehicle()

# ship1=Ship("WBAFR7C57CC811956","HMY ROYAL CAROLINE","Tural Turallı","Yes",2378237,"MTU Marine Diesel Engine",15)

# ship1.show\_vehicle()

# plane1=Plane("RE34126P894593E","Boeing","Məmməd Məmmədli",6,99,3674093,"General Electric",55)

# plane1.show\_vehicle()

#except Exception as ex:

# print("Exception : ",ex)

#Task 2

#car1=Car("PJ12345U123456P","Toyota","John Johnlu","Yes",2488152,"Blackstone",3.5)

#ship1=Ship("WBAFR7C57CC811956","HMY ROYAL CAROLINE","Tural Turallı","Yes",2378237,"MTU Marine Diesel Engine",15)

#plane1=Plane("RE34126P894593E","Boeing","Məmməd Məmmədli",6,99,3674093,"General Electric",55)

#VehicleDepo=[car1,ship1,plane1]

#for x in VehicleDepo:

# print(x.show\_vehicle())