1. Create a Food Storage System which perform following Operation on it. using OOPS Concept (Class and Object )

Class Food has Following Properties (foodId, foodName, foodType(Veg/Non-Veg) ,foodPrice)

1. Add Food
2. Update Food
3. Delete Food
4. Show Food List
5. Show Food by Id
6. Show Food by Name
7. Show Food by Type
8. Sort Food List by Name
9. Sort Food List by Price

model.py

# It is a Food model which represent food details.

class Food:

    def \_\_init\_\_(self,foodid=0,foodName='',foodType='',foodPrice=0.0):

        self.foodid=foodid

        self.foodName=foodName

        self.foodType=foodType

        self.foodPrice=foodPrice

    def \_\_repr\_\_(self):

        return f'Food[{self.foodid},{self.foodName},{self.foodType},{self.foodPrice}]'

view.py

from model import Food

foodlist=[]

foodlist.append(Food(1001,'Chicken Handi','Nonveg',500))

foodlist.append(Food(1002,'Veg Handi','Veg',400))

# Create

def save(fd):

    foodlist.append(fd)

    print('Food is Added Successfully.')

# Update

def update(foodid,newfood):

    food=getFood(foodid)  #find which we want to update

    index=foodlist.index(food)    # index its index

    foodlist[index].foodName=newfood.foodName   # replace with new value

    foodlist[index].foodType=newfood.foodType

    foodlist[index].foodPrice=newfood.foodPrice

# Delete

def delete(foodid):

    food=getFood(foodid)  #find which we want to update

    index=foodlist.index(food)

    foodlist.pop(index)

# Read All object

def getAll():

    return foodlist

# Read Single object

def getFood(foodId):

    l1=list(filter(lambda fd:fd.foodid==foodId,foodlist))

    if len(l1)==1:

        return l1[0]

    else:

        print('\tFood Not Found')

# Show Food by Name

def getfoodbyname(foodName):

    l1=list(filter(lambda fd:fd.foodName==foodName,foodlist))

    if len(l1)==1:

        return l1[0]

    else:

        print('\tFood Not Found')

# Show Food by Name

def getfoodbytype(foodType):

    l1=list(filter(lambda fd:fd.foodType==foodType,foodlist))

    if len(l1)>=1:

        return l1

    else:

        print('\tFood Not Found')

# Sort Food List by Name

def sortbyname():

    foodlist.sort(key=lambda fd:fd.foodName)

    return foodlist

# Sort Food List by Price

def sortbyprice():

    foodlist.sort(key=lambda fd:fd.foodPrice)

    return foodlist

template.py

from model import Food

import view as v

import os

choice=0

while choice!=10:

    print('-$-'\*3,'FOOD STORAGE SYSTEM','-$-'\*3)

    print('\n\t1.Add Food')

    print('\t2.Update Food')

    print('\t3.Delete Food')

    print('\t4.Show Food List')

    print('\t5.Show Food by Id')

    print('\t6.Show Food by Name')

    print('\t7.Show Food by Type')

    print('\t8.Sort Food List by Name')

    print('\t9.Sort Food List by Price')

    print('\t10.Exit')

    choice=int(input('\nEnter Your Choice='))

    os.system('cls')

    if choice==1:

        fd=Food() # object with default value

        # Here replace the value with user inputs.

        fd.foodid=int(input('\tEnter Food ID='))

        fd.foodName=input('\tEnter Food Name=')

        fd.foodType=input('\tEnter Food Type(Veg/Nonveg)=')

        fd.foodPrice=float(input('\tEnter Food Price='))

        v.save(fd)

    # Update Food

    elif choice==2:

        foodid=int(input('\tEnter Food ID to Update='))

        food=v.getFood(foodid)

        if food!=None:

            print('\tEnter Updated Details=')

            print('\t',food)

            newfood=Food()

            newfood.foodName=input('\tEnter Food Name=')

            newfood.foodType=input('\tEnter Food Type(Veg/Nonveg)=')

            newfood.foodPrice=float(input('\tEnter Food Price='))

            v.update(foodid,newfood)

            print('\tFood is Updated.')

        else:

            print('\tInvalid Food Id........!')

    # Delete Food

    elif choice==3:

        foodid=int(input('\tEnter Food ID to Delete='))

        food=v.getFood(foodid)

        if food!=None:

            v.delete(foodid)

            print('\tFood Deleted Successfull.')

        else:

             print('\tInvalid Employee Id........!')

    # Show All Food

    elif choice==4:

        flist = v.getAll()

        print('-+-'\*10,"Food List",'-+-'\*10)

        print()

        for food in flist:

            print('\t\t',food)

    # Show Food by Id

    elif choice==5:

        foodid=int(input('\tEnter Food ID to Search='))

        item=v.getFood(foodid)

        if foodid!=None:

            print('\tItem Found')

            print('\t',item)

    #Show Food List by Name

    elif choice==6:

        foodName=input('\tEnter Food Name to Search=')

        item=v.getfoodbyname(foodName)

        if foodName!=None:

            print('\tItem Found')

            print('\t',item)

    #Show Food by Type

    elif choice==7:

        foodType=input('\tEnter Food Type to Search(Veg/Non-Veg)=')

        item=v.getfoodbytype(foodType)

        if foodType!=None:

            for f in item:

                print(f)

    # Sort food by name

    elif choice==8:

        namesort=v.sortbyname()

        print("="\*5,"Sorted Food by Name","="\*5)

        for f in namesort:

            print(f)

    # Sort food by price

    elif choice==9:

        pricesort=v.sortbyprice()

        print("="\*5,"Sorted food by Price","="\*5)

        for f in pricesort:

            print(f)

    elif choice==10:

        print('\t\tThank You....!')

    else:

        print('\tInvalid Choice')