'''

MSc Digital & Technology Solutions - CETM65 Software Engineering Principles-

Author: Alan John Heslop

Student Number: bh83dl

Date: 11/10/2020

---------------------------------------------------------------------------------

EDUCATION OOP

- SCHOOL DETAILS

- here we want to be able to define the school in which a student (child) attends and the location of the school (assuming they could be different)

- CHILD DETAILS

- here we are able to define the child who is a child of school - the basic information will be getting the details of the child.

- Overview

- Overview's main idea is to be multiple inheritence and have it's own details requested (grades and attendance) - then this will be consolidated as one final string to output

'''

class School(): #defines a class in python, this is the school class

def \_\_init\_\_(self, schoolName, location):

self.schoolName = schoolName # set attributes for the class

self.location = location # set attributes for the class

#print('(Initialized School Class: {})'.format(self.schoolName)) # simply a debugging statement to see if the schoolname has correctly passed in a string

def getSchoolName(self):# Defining the method schoolname

return self.schoolName

def getSchoolLocation(self):#Defining the method schoollocation

return self.location

def \_\_str\_\_(self): # I am using str to return the school name and location

return f"School Name: {self.schoolName}. School Location: {self.location}"

# repr method below, showing 2 options of printing the item

def \_\_repr\_\_(self): # I am using the reper method below to pass the same details as the string

#return f" {self.schoolName}, {self.location}"

return "school name: {}, Location {}".format(self.schoolName, self.location)

# Prints using the REPR function too.

# instantiating the items for printing

sch = School("Royal Grammar School", "Newcastle Upon Tyne")

print("Below is an example of passing \_\_repr\_\_")

print(repr(sch)) # Print the repr string

print()

#2nd class about the child attending the School

class Child(School): # Child class is using inheritance from the School Class

def \_\_init\_\_(self, name, age, yeargroup, schoolName, location):

self.name = name

self.age = age

self.yeargroup = yeargroup

super().\_\_init\_\_(schoolName, location)

self.\_\_password = 1 #private arrtribute for the users password

self.\_\_pass = 20

self.word = 30

print ({self.name})

#Showing that data is being passed correctly on-screen

#print('(Initialized Child Class: {})'.format(self.name))

#print('(Initialized Child Class: {})'.format(self.age))

#print("{}".format(self.name))

def getChildName(self):

return self.name

def getChildAge(self):

return self.age

def getChildYearGroup(self):

return self.yeargroup

#this string is displaying the information

def \_\_str\_\_(self):

return f"{self.name} is {self.age} and is currently is yearx {self.yeargroup} and the school name is {self.location } {super().\_\_str\_\_()}"

def \_\_repr\_\_(self):

return f"{self.\_\_class\_\_.\_\_name\_\_}{self.\_\_dict\_\_}"

# all the attributes associated with the 't'

#instantiation

chi = Child("Kieran", 12, 8, "Royal Grammar School", "Newcastle Upon Tyne")

print(repr(chi))

print()

# 3rd class will show the attendance, grade and bring back all data

class Overview(Child, School): #multiple inheritence

def \_\_init\_\_(self, attendance, grade, name, age, yeargroup, schoolName, location):

self.attendance = attendance

self.grade = grade

super().\_\_init\_\_(name, age, yeargroup, schoolName, location)

#print('(Initialized Overview Class: {})'.format(self.name))

def getAttendance(self):

return self.attendance

def getGrade(self):

return self.grade

# inherited

# \_\_Str\_\_ informal representation

def \_\_str\_\_(self): # STR method to print out an easily readable sentence.

return f"{self.name}s attendance has been oustanding, acheiving an overall attendance of: {self.attendance}. He is currently Aged: {self.age} and is in YearGroup: {self.yeargroup} and his overall grade for the year is: {self.grade} - he attends the school in {self.location} and the name of the school is {self.schoolName}"

def \_\_repr\_\_(self):

return f"{self.\_\_class\_\_.\_\_name\_\_}{self.\_\_dict\_\_}"

# Final instantiating

# Person 1

p1 = Overview("80%", "A", "Kieran", 8, 12, "Royal Grammar School", "Newcastle Upon Tyne")

print(p1)

print()

# Person 2

p2 = Overview("20%", "B", "Alan Heslop", 31, "University", "Sunderland University", "Sunderland")

print(p2)

print()

#access repr string

print()

print("\_\_repr\_\_ output")

print()

print(repr(p1))

# the manual way of inputting the data to the screen, this was the first method to test the data passed

#print(p1.getSchoolName())

#p1.name = "alan"

#p1.age = "12"

#p1.yeargroup = 8

#p1.location = "Newcastle"

#print()

#print(Overview.\_\_bases\_\_)

#returns a list of all the members in the specified object

#print(dir(p1))