import os as os

os.system('cls')

class Fraction:

# The init method initiates variables. It feels redundant, but I see how modifying the line

#like I did in line 12 can be useful.

def \_\_init\_\_(self,numerator, denominator):

self.numerator = numerator

self.denominator = denominator

# The string method now manipulates initialized variables into a string

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

return f"{str(self.numerator)}/{str(self.denominator)}"

# The show method prints out the string method

def show(self):

print(f"The fraction is {str(self.numerator)}/{str(self.denominator)}")

#print("The contents of the fraction are:", num,"and", den)

def add(self, other):

#Get the common denominator

other.numerator = other.numerator

other.denominator = other.denominator

newden = other.denominator\*self.denominator

#Get the numerators

self.newnum = self.numerator\*other.denominator

other.newnum = other.numerator\*self.denominator

#Add them up

newnum = self.newnum + other.newnum

print(f"The sum is {newnum}/{newden}")

def eq(self, other):

prod1 = self.numerator\*other.denominator

prod2 = self.denominator\*other.numerator

return prod1 == prod2

# Validation and Verification (V&V)

fract1 = Fraction(1,2)

fract2 = Fraction(3,4)

fract3 = Fraction(5,6)

fract4 = Fraction(4,24/5)

fract4\_decimal = Fraction(4,4.8)

fract5 = Fraction(-5,6)

fract\_irrational\_exactvalue = Fraction(1, 1/3)

fract\_irrational\_decimalvalue\_16 = Fraction(1, 0.3333333333333333) # sixteen places

fract\_irrational\_decimalvalue\_15 = Fraction(1, 0.333333333333333) # fifteen places

fract1.show() # The fraction is 1/2

fract4.show() # The fraction is 4/4.8

# .show method works and converts to decimal form!

fract1.add(fract2) # The sum is 10/8

fract1.add(fract5) # The sum is -4/12

# .add method works with positive and negative numbers!

print(fract1.eq(fract3)) # Expected false, got false

print(fract3.eq(fract4)) # Expected true, got true

print(fract3.eq(fract4\_decimal)) # Expected true, got true

print(fract\_irrational\_exactvalue.eq(fract\_irrational\_decimalvalue\_16)) # Expected true, got true

print(fract\_irrational\_exactvalue.eq(fract\_irrational\_decimalvalue\_15)) # Expected true, got false

# .eq method works with equivalent decimal and fraction values.

# Irrational values must be rounded out to the sixteenth decimal place to be counted equivalent to the fraction

The fraction is 1/2

The fraction is 4/4.8

The sum is 10/8

The sum is -4/12

False

True

True

True

False

PS C:\Users\store>