

```

# Fahmi Omer
# ICS 4U1
# Classes

# Name: Fahmi Omer
# UWO Email: fomer4@uwo.ca
# Assignment #4 - Air Travel
# Program Description: Take input from files and create and use classes to
solve various questions about the given airports and flights such as
destinations, origins, flights in countries, in cities, and more

# Importing the Airport file so the Airport class can be used
from Airport import *

# Creating the flight class for use

class Flight:
    # Initializing all properties of the class
    def __init__(self, flightNo="", origin=Airport(),
destination=Airport()):
        if not isinstance(origin, Airport):
            raise TypeError(
                "The origin and destination must be Airport objects")
        if not isinstance(destination, Airport):
            raise TypeError(
                "The origin and destination must be Airport objects")

        self._flightNo = flightNo
        self._origin = origin
        self._destination = destination

    # Printing return line of the object when printed
    def __repr__(self):
        if self.isDomesticFlight():
            return "Flight: {} from {} to {}".format(self._flightNo,
self._origin._city, self._destination._city) + " {domestic}"
        else:
            return "Flight: {} from {} to {}".format(self._flightNo,
self._origin._city, self._destination._city) + " {international}"

    # Checking if self and other are the same flight
    def __eq__(self, other):
        if isinstance(other, Flight):
            if self._origin == other._origin and self._destination ==
other._destination:
                return True

```

```

        else:
            return False
    else:
        return False

# returning the number attribute of the flight
def getFlightNumber(self):
    return self._flightNo

# returning the origin airport attribute of the flight
def getOrigin(self):
    return self._origin

# returning the destination airport attribute of the flight
def getDestination(self):
    return self._destination

# Checking if the flight is within the country or outside of the
country
def isDomesticFlight(self):
    if self._origin._country == self._destination._country:
        return True
    else:
        return False

# creating function to allow changing the origin airport
def setOrigin(self, origin):
    self._origin = origin

# creating function to allow changing the destination airport
def setDestination(self, destination):
    self._destination = destination

```