OOP

IS612 Introduction To Coding

Spring 2022

Beautiful is better than ugly. Explicit is better than implicit. Simple is better than complex. Complex is better than complicated. Flat is better than nested. Sparse is better than dense. Readability counts. Special cases aren't special enough to

Although practicality beats purity. Errors should never pass silently. Unless explicitly silenced. In the face of ambiguity, refuse the temptation to guess. There should be one – and preferably only one — obvious way to do it. Although that way may not be obvious at first unless you're Dutch. Now is better than never. Although never is often better than right now. If the implementation is hard to explain, it's a bad

may be a good idea.

op s,191 - eapt ne honking great Namespaces are may be a good idea is easy to explain, it dea. If the implementation

now. If the implementation is hard to explain, it's a bad better than never. Although never is often better than right way may not be obvious at first unless you're Dutch. Now is and preferably only one — obvious way to do it. Although that ambiguity, refuse the temptation to guess. There should be one pass silently. Unless explicitly silenced. In the face of Although practicality beats purity. Errors should never break the rules.

op ugnoua jeroads Readability counts, Special cases aren't nested. Sparse is better than dense. than complicated. Flat is better than is better than complex. Complex is bette Explicit is better than implicit. Simple Beautiful is better than ugly.

## **CEX 3: OOP 2**

SIMPLE CLASSES AND OBJECTS

## Class Turboprop

### TurboProp

- String variant
- String modelNumber
- String aircraftTailNumber
- int crew
- int passengerCapacity
- List (Passenger) passengers
- float length
- float wingSpan
- float height
- int currentSpeed
- int currentAltitude
- boolean isLanded
- boolean is Airborne
- + static int aircraftCount = o
- + displayACInfo()...void
- + displayACStatus...void
- + displayPassengerInfo()...void
- + TakeOff()...void
- + Land()...void

Method	Description
displayACInfo	Displays the value of the variant, modelNumber, and aircraftTailNumber attributes (see output slide for format)
displayACStatus	Displays the value of the currentSpeed, currentAltitude, and the isAirborne attributes (see output slide for format)
displayPassengerInfo	Displays the passenger IDNumber and Seat Number of the Passenger object (see slide 4 for Passenger Class) and (see output slide for format)
TakeOff	if the isAirborne attribute is True displays 'Aircraft is already airborne'otherwise sets the currentSpeed to 75, the currentAltitude to 200, the isLanded to False, the isAirborne to True and displays '[aircraftTailNumber] has taken off'
Land	if the isAirborne attribute is False displays 'Aircraft is already landed'otherwise sets the currentSpeed to o, the currentAltitude to o, the isLanded to True, the isAirborne to False and displays '[aircraftTailNumber] is landed'

## Class Passenger

#### **Passenger**

- String IDNumber
- String SeatlNumber
- + static int passengerCount
- + displayPassengerInfo()...void

Method	Parameters	Description
displayPassengerInfo	None	Displays the value of the IDNumber and SeatNumber attributes (see output slide for format)

Constructor increments passengerCount after each object is constructed

Implement all accessors and mutators

# Tasks

Task	Requirements
1	Create the Turboprop Class
2	Create the Passenger Class
3	Create the following Passenger objects IDNumber = 'P-01' SeatNumber = '1A' IDNumber = 'P-02' SeatNumber = '1B' IDNumber = 'P-03' SeatNumber = '2A' IDNumber = 'P-04' SeatNumber = '2B'
4	Create the TurboProp object with the following parameters:  variant = 'BeechCraft 200' modelNumber = 'BC-001'  aircraftTailNumber = 'US-BC-01' crew = 2  passengerCapacity = 10 passengers = passengers created in Task 3  length = 43 ft 6 in wingSpan = 57 ft 6 in  height = 14 ft 4 in currentSpeed = 130 mph  currentAltitude = 5000 ft isLanded = False isAirborne = True
5	Call the displayACStatus() method on the TurboProp object created in Task 4
6	Call the displayPassengerInfo() method on the TurboProp object created in Task 4
7	Call the Land() method on the TurboProp object created in Task 4
8	Call the displayACStatus() method on the TurboProp object created in Task 4
9	Call the TakeOff() method on the TurboProp object created in Task 4
10	Call the displayACStatus() method on the TurboProp object created in Task 4

### Tasks

```
===== RESTART: C:/Users/msgth/Desktop/PACEPython/Exa
     Aircraft Status For: US-BC-01
Speed: 130
Altitude: 5000
In Flight True
Passenger ID: P-001 Task 5
Passenger Seat: 1A
Passenger ID: P-002
Passenger Seat: 1B
                   Task 6
Passenger ID: P-003
Passenger Seat: 2A
Passenger ID: P-004
Passenger Seat: 2B
US-BC-01 is landed TOSK 7
     Aircraft Status For: US-BC-01
Speed: 0
Altitude: 0
In Flight False
US-BC-01 has taken off TOSK 9
     Aircraft Status For: US-BC-01
                                             Task 10
Speed: 75
Altitude: 200
In Flight True
```