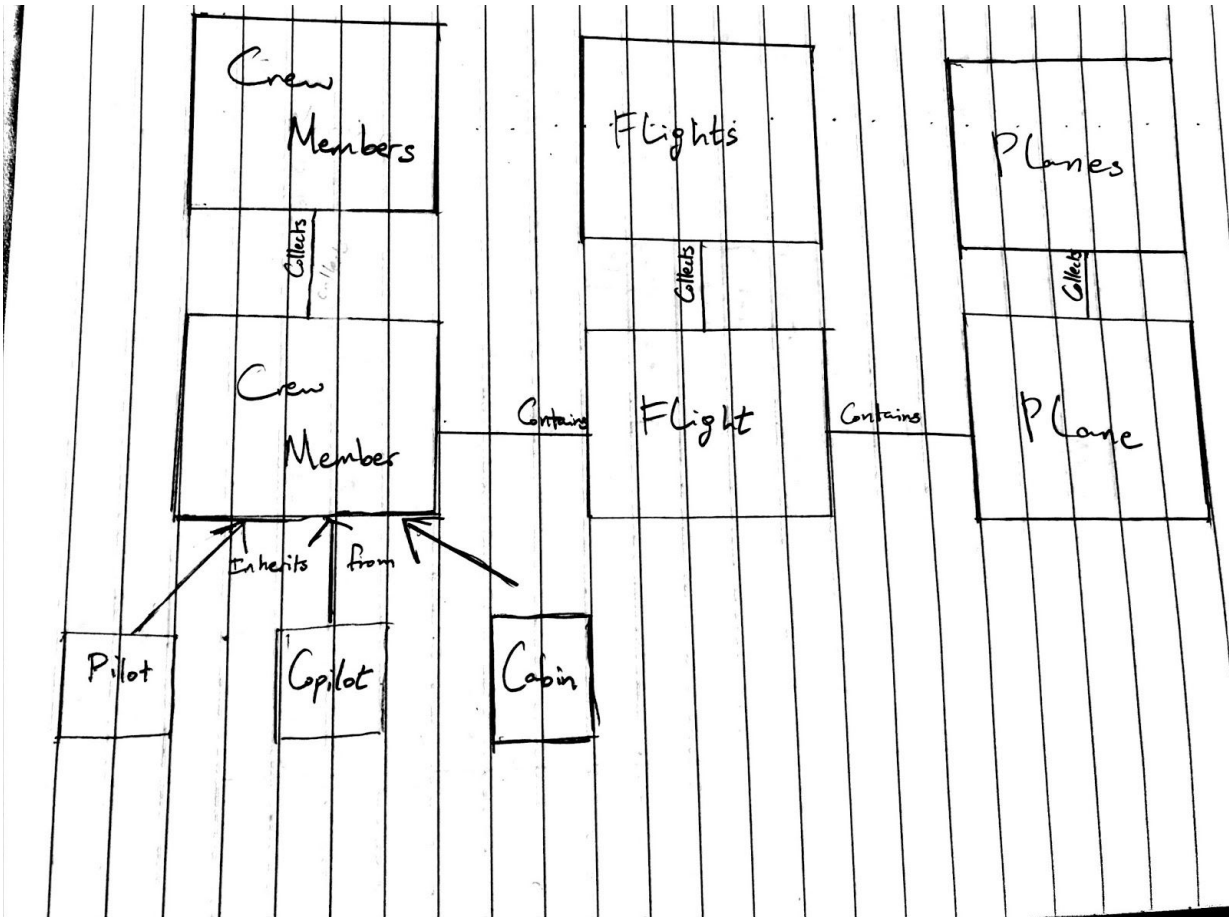
SUDARSHANA JAGADEESHI

HOMEWORK 5

CSCE 1040

MEAN GREENS AIRLINE DESIGN

Class Relationships(Changed from HW 4)



Class Contents

Note: * indicates base class members not unique to that derived class

Plane

Make (e.g. Boeing)
Model (eg 737)
Tail Number (eg B171A)
Number of Seats (eg 137)
Range (e.g. 800 miles)
Status (out, in, repair)

Set/get make
Set/get model
Set/get tail number
Set/get number of seats
Set/get range

Set/get status

Planes
Collection of plane objects

Add a plane
Edit a planes information
Delete a plane
Search for a plane
Print a list of all the planes
Print the detailed information for a single plane

Crew Member
Name
ID number
Status (available, on leave, sick)

Set/get Name
Set/get ID number
Set/get status

Pilot
Name*
ID number *
Status (available, on leave, sick)*
5 character Rating code
Cumulative hours as a pilot

Set/get Name*
Set/get ID number*
Set/get status*
Set/get character rating code
Set/get cumulative hours as a pilot

Copilot
Name*
ID number *
Status (available, on leave, sick)*
4 character Rating code
Cumulative hours as a copilot

Set/get Name*
Set/get ID number*

Set/get status*
Set/get character rating code
Set/get cumulative hours as a copilot

Cabin
Name*
ID number *
Status (available, on leave, sick)*
Position

Set/get position

Crew Members
Collection of crew member objects

Add a crew member
Edit a crew member's information
Delete a crew member
Search for a crew member
Print a list of all the crew members
Print the detailed information for a single crew member

Flight
Plane ID (Tail number)
Pilot ID
CoPilot ID
Crew IDs for 3 Cabin Crew Members
Start Date/Time with TZ
End Date/Time with TZ
Starting Airport code (3 letters)
Ending Airport Code (3 letters)
Number of Passengers
Status (active, cancelled, completed)

Set/get plane ID
set/get pilot ID
set/get copilot ID
set/get crew IDs
set/get start time
set/get end time
set/get starting airport
set/get ending airport
set/get number of passengers

set/get status

Flights
Collection of flight objects

Add a flight
Edit a flights information
Delete a flight
Search for a flight
Print a list of all flights
Print the detailed information for a single flight

Clean useless flights
Update completed flights
Print assignment schedule for a specific plane
Print assignment schedule for a specific crew member

Function Pseudocode

In main:
1. Print menu
Prints to the screen 4 options: 1) manage planes, 2) manage crew members 3)Manage flights 4) quit. Under each option there are suboptions labeled a, b, c) to add, delete, search, clean, etc.

2. Load Data
Prompt the user for the name of an input file
Open that file
If the file could not be opened, state so.

Read that data in to the collections

3. Save Data
Prompt the user for the name of an output file
Create that file

Write the data from all the collections to the file

In class planes:
4.Add a plane
Prompt the user for all data fields associated with a plane

Search through all the members in the collection to find a matching copilot ID.

If a matching ID exists, identify whether that crew member is a pilot

Create a new plane object in the collection
Populate this new object with the responses

5.Edit a planes information
Prompt the user for the plane ID
Search through all the planes in the collection to find a matching ID.

If no hits are found, return that the plane could not be found and that nothing was edited
If a hit is found, prompt the user for all the data fields.

If the user enters nothing for a specific data field, keep the old value.
Else, update the value with the user input.

Finally, quickly echo any changes. i.e. “member Range was updated to 900.”

6.Delete a plane
Prompt the user for the plane ID
Search through all the planes in the collection to find a matching ID.

If no hits are found, return that the plane could not be found
If a hit is found, print that a hit was found, and ask to confirm deletion

If yes, delete.
If no, return to the menu.

7.Search for a plane
Prompt the user for the plane ID
Search through all the planes in the collection to find a matching ID.

If no hits are found, return that the plane could not be found
If a hit is found, print that a hit was found

8.Print a list of all the planes
Loop over all the planes in the collection and print all the data fields associated with each plane

9.Print the detailed information for a single plane
Prompt the user for the plane ID
Search through all the planes in the collection to find a matching ID.

If no hits are found, return that the plane could not be found
If a hit is found, print that a hit was found, and print all the data fields for that plane only

In class crew members:

10.Add a crew member

Prompt the user for all data fields associated with a crew member

Search through all the crew members in the collection to find a matching ID.
If a matching ID exists, print an error message and reprompt for a new ID.

If the type is a pilot, prompt for the pilot-specific data fields.
If the type is a copilot, prompt for the copilot-specific data fields

Create a new crew member object in the collection
Populate this new object with the responses

11.Edit a crew members information

Prompt the user for the crew member ID
Search through all the crew members in the collection to find a matching ID.

If no hits are found, return that the crew member could not be found and nothing was edited
If a hit is found, prompt the user for all the data fields.

If the user enters nothing for a specific data field, keep the old value.
Else, update the value with the user input.

Finally, quickly echo any changes. i.e. “member Range was updated to 900.”

12.Delete a crew member

Prompt the user for the crew member ID
Search through all the crew members in the collection to find a matching ID.

If no hits are found, return that the crew member could not be found
If a hit is found, print that a hit was found, and ask to confirm deletion

If yes, delete.
If no, return to the menu.

13.Search for a crew member

Prompt the user for the crew member ID
Search through all the crew members in the collection to find a matching ID.

If no hits are found, return that the crew member could not be found
If a hit is found, print that a hit was found

14.Print a list of all the crew members
Loop over all the planes in the collection and print all the data fields associated with each plane

15.Print the detailed information for a single crew member
Print the detailed information for a single crew member
Prompt the user for the crew member ID
Search through all the crew members in the collection to find a matching ID.

If no hits are found, return that the crew member could not be found
If a hit is found, print that a hit was found, and print all the data fields for that crew member only

In class flights:

16.Add a flight
Find the size of the members collection
Find the size of plane collection
If the size of the members collection is less than 5 or if the size of the plane collection is less than 1, stop and do not add a flight.

Prompt the user for all data fields associated with a flight

Search through all the planes in the planes collection to find a matching ID.
If a matching ID exists, print an error message and reprompt for a new ID.

Search through the flight collection and store all entries with the user-entered plane.
If the starting time the user inputs is during an active flight, state so and do not add that flight.

Search through all the members in the collection to find a matching pilot ID.
If a matching ID exists, identify whether that crew member is a pilot

If no ID is found, or the crew member is not a pilot, state so and do not add that flight.

Search through all the members in the collection to find a matching copilot ID.
If a matching ID exists, identify whether that crew member is a pilot

If no ID is found, or the crew member is not a copilot, state so and do not add that flight.

Create a new plane object in the collection
Populate this new object with the responses

17.Edit a flights information
Prompt the user for the flight ID
Search through all the flights in the collection to find a matching ID.

If no hits are found, return that the flight could not be found and that nothing was edited
If a hit is found, prompt the user for all the data fields.

Add the populated flight to the collection

18.Delete a flight
Prompt the user for the flight ID
Search through all the flights in the collection to find a matching ID.

If no hits are found, return that the flight could not be found
If a hit is found, print that a hit was found, and ask to confirm deletion

If yes, delete.
If no, return to the menu.

19.Search for a flight
Prompt the user for the flight ID
Search through all the flights in the collection to find a matching ID.

If no hits are found, return that the flight could not be found
If a hit is found, print that a hit was found

Print a list of all the flights
Print all the data fields for all flights in the collection.

20.Print a list of all the flights
Loop over all the flights in the collection and print all the data fields associated with each flight

21. Print a list of all flights by status
Loop over all the flights in the collection
If the flight is active, print it's data

Loop over all the flights again
If the flight is completed, print it's data

Loop over all the flights a third time
If the flight is cancelled, print it's data
21.Print the detailed information for a single flight
Prompt the user for the flight ID
Search through all the flights in the collection to find a matching ID.

If no hits are found, return that the flight could not be found
If a hit is found, print that a hit was found, and print all the data fields for that flight only

22.Clean useless flights
Loop over all flights
If the flight has a status of “completed” or “cancelled”
Delete that flight from the collection

23. Update completed flights
Loop over all flights

Get the current time from the user in the format in a list of 6 integers
Compare the first integer of the current time with the first integer of the ending time of the flight
If they are the same, move on to the next integer. If those are the same, move on again, etc.

If the current time is greater at any point, examine the next flight
If the current time is less at any point, update the status to “Completed”

24. Print the assignment schedule for a specific plane
Prompt the user for the plane ID

Loop over the collection and find all the flights that have a plane ID that matches the one the user provided.
If the user-provided ID is not in the collection, stop and state that the plane was not found

Loop over all the hits and print each flight and the start and end time. Then print all the crew members on that plane and their roles.

25. Print the assignment schedule for a specific crew member
Prompt the user for the crew member ID

Loop over the collection and find all the flights that have a crew member ID that matches the one the user provided.
If the user-provided ID is not in the collection, stop and state that the crew member was not found

Loop over all the hits and print each flight and the start and end time