CUSTOMER CLASS

*// Ethan Sylvester 101479568 | Amanda Gurney 101443253 | Taylor Martin 100849882*

*// Group MA1 22 | Assignment 2*

*// COMP 2129 | CRN: 15646*

using System;

using System.Collections.Generic;

*// Ethan Sylvester 101479568 | Amanda Gurney 101443253 | Taylor Martin 100849882*

*// Group MA1 22 | Assignment 2*

*// COMP 2129 | CRN: 15646*

using System.Linq;

using System.Security.Cryptography.X509Certificates;

using System.Text;

using System.Threading.Tasks;

namespace Group22\_Project

{

*internal* class Customer

{

*public* *static* int customerIDRef = 1;

*public* int CustomerID { get; set; }

*public* string FirstName { get; set; }

*public* string LastName { get; set; }

*public* string PhoneNumber { get; set; }

*public* int NumOfBookings { get; set; }

*// Constructor*

*public* Customer(string fname, string lname, string phone)

{

FirstName = fname;

LastName = lname;

PhoneNumber = phone;

NumOfBookings = 1;

customerIDRef++;

CustomerID = customerIDRef;

}

*// Set customerID automatically.*

*public* void IncrementBookings()

{

NumOfBookings++;

}

*public* *override* string ToString()

{

return $" -- Customer ID: {CustomerID} | Name: {FirstName} {LastName} | Phone: {PhoneNumber} | Number Of Bookings: {NumOfBookings}";

}

}

}

FLIGHT CLASS

*// Ethan Sylvester 101479568 | Amanda Gurney 101443253 | Taylor Martin 100849882*

*// Group MA1 22 | Assignment 2*

*// COMP 2129 | CRN: 15646*

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Group22\_Project

{

*internal* class Flight

{

*public* *static* int FlightNumberRef = 1000;

*public* int FlightNumber { get; }

*public* int MaxPassengers { get; set; }

*private* Customer[] customersArray = new Customer[10];

*public* Flight(int maxPassengers)

{

*// FlightNumberRef should increment with every instance, and then*

*// the flight number is taken from the value so that it is automatically*

*// a unique 4 digit ID*

FlightNumberRef++;

FlightNumber = FlightNumberRef;

MaxPassengers = maxPassengers;

}

*// This function handles extending the array 5 customers at a time to reduce*

*// memory use while also not using a ton of resources to copy the array repeatedly*

*public* void AddCustomer(Customer passenger)

{

bool extendArray = true;

for (int i = 0; i < customersArray.Length; i++)

{

if (customersArray[i] == null)

{

extendArray = false;

customersArray[i] = passenger;

break;

}

}

if (extendArray)

{

if (customersArray.Length == MaxPassengers)

{

*// MAX CUSTOMERS ACHIEVED. PRINT RESULT. NO MORE CUSTOMERS ALLOWED*

return;

}

else if (customersArray.Length + 5 > MaxPassengers)

{

int growth = MaxPassengers - customersArray.Length;

Customer[] tempArray = new Customer[customersArray.Length + growth];

for (int i = 0; i < customersArray.Length; i++)

{

tempArray[i] = customersArray[i];

}

customersArray = tempArray;

}

else

{

Customer[] tempArray = new Customer[customersArray.Length + 5];

for (int i = 0; i < customersArray.Length; i++)

{

tempArray[i] = customersArray[i];

}

customersArray = tempArray;

}

}

}

*public* string ToString()

{

string returnString = $"Flight Number: {FlightNumber}: Number of Passengers: {GetPassengerCount()} out of {MaxPassengers}\n";

foreach (Customer customer in customersArray)

{

if (customer != null)

returnString += customer.ToString() + "\n";

}

return returnString;

}

*public* int CustomerListLength()

{

return customersArray.Length;

}

*public* int GetPassengerCount()

{

int count = 0;

foreach (Customer customer in customersArray)

{

if (customer != null)

{

count++;

}

}

return count;

}

*public* Customer[] GetCustomerArray()

{

return customersArray;

}

}

}