```
// Candice Morris
// SCS131
/**
 * @author barrygp
 * @version 1.0
 * @created 22-Apr-2018 10:22:37 PM
 */
public class TravelAgent {

        /**
         * Array of Customers
         */
        private Customer[] customers;
        /**
         * Array of Flights
         */
        private Flight[] flights;

        public final int MAX_FLIGHTS = 10;

        public final int MAX_CUSTOMERS = 100;

        // TODO: implement static createCustomer similar to
createFlight below.
    static public Customer createCustomer (String orig, String dest,
String depDT, String arrDT){
        Customer ctm = null;
        ctm = new Customer (orig, dest, depDT, arrDT);
        return ctm;
    }
        /**
         * Create a new Flight factory method
         */
        static public Flight createFlight(String orig, String dest,
String depDT, String arrDT){
            Flight flt = null;
            flt = new Flight(orig, dest, depDT, arrDT);
            return flt;
        }

        public TravelAgent(){
            flights = new Flight[MAX_FLIGHTS];
            customers = new Customer[MAX_CUSTOMERS];
        }

        /**
         *
         * @exception Throwable
         */
        public void finalize()
```

```
  throws Throwable{

}

/**
 * Create the Itinerary for the passed in Customer Name.
 *
 * @param custName
 */
public Itinerary createCustomerItinerary(String custName){
        return null;
}

/**
 * Array of Customers
 */
public Customer[] getCustomers(){
    // TODO:
    // REIMPLEMENT THIS METHOD TO RETURN COPY OF
    // CUSTOMER ARRAY. (SEE getFlights METHOD)
public void setCustomers(String newVal){
    origination = newVal;
}
public String getCustomers(){
        return customers;
}



/**
 * Array of Customers
 *
 * @param newVal
 */
public void setCustomers(Customer[] newVal){
        customers = newVal;
}

/**
 * Add customer to the customers array if array has room
 */
public void addCustomer(Customer cust){
    // TODO:
    // IMPLEMENT THIS METHOD TO ADD cust TO
    // CUSTOMER ARRAY. (SEE addFlight METHOD)
    public Cusotmer [] customers;
    customer.addCustomer(Customer crust);

}
```

```
        public Flight[] getFlights(){
                // Let's make a copy so as not to return reference
                // to the private variable.

                // Let's create a temporary flight array for return
                Flight[] tmpFltsArr = new Flight[MAX_FLIGHTS];

                // Let's loop through the flights array
                for(int i = 0; i < MAX_FLIGHTS; i++){
                    // Let's get the instance of the primary flight
                    Flight primaryFlightInstance = flights[i];
                    // Let's create a copy of this primary flight
using the
                    // Flight copy constructor. if the flight is not
null
                    if(primaryFlightInstance != null){
                        Flight fltCopy = new
Flight(primaryFlightInstance);
                        // Let's put this newly created copy into the
temporary
                        // flights array that will be returned.
                        tmpFltsArr[i] = fltCopy;
                    }
                }
                 return tmpFltsArr;
        }

        /**
         * Add flight to the array if array has room
         */
        public void addFlight(Flight flt){
            // Let's iteratye through the array to
            // find the first available spot.

            for(int flightIndex=0;flightIndex < MAX_FLIGHTS;
flightIndex++){

                // Check if first avail spot
                if(flights[flightIndex] == null){
                    // Let's add the flight to the Flights array
                    flights[flightIndex] = flt;
                    // Being we are in here means we found a spot so
                    // let's break out of the loop
                    break;
                }
            }
        }

        /**
```

```
         *
         * @param newVal
         */
        public void setFlights(Flight[] newVal){
                flights = newVal;
        }

        /**
         * Override toString() method
         */
        @Override
        public String toString() {
            // Let's declare our return string
            String outputStr = "Travel Agent Output\n\n";

            // Let's add the flights to our output string
            outputStr += "FLIGHTS:\n";
            for(int i=0; i < MAX_FLIGHTS; i++){
                if(flights[i] != null){
                    outputStr += flights[i];
                }
            }

            // Let's add a new line for readablity
            outputStr += "\n";

            // Let's add the customers to our output string
            outputStr += "CUSTOMERS:\n";
            for(int i=0; i < MAX_CUSTOMERS; i++){
                if(customers[i] != null){
                    outputStr += customers[i];
                }
            }
            return outputStr;
        }
}
```