```
/**
* Assignment 02 Part 1
* COMP-249
* Written by:
* @author Yan Pilon (40246124)
* @author Miskat Mahmud (40250110)
* Due Date: 04 March 2023
*/
```

Here is the method to find the least and the most expensive aircraft

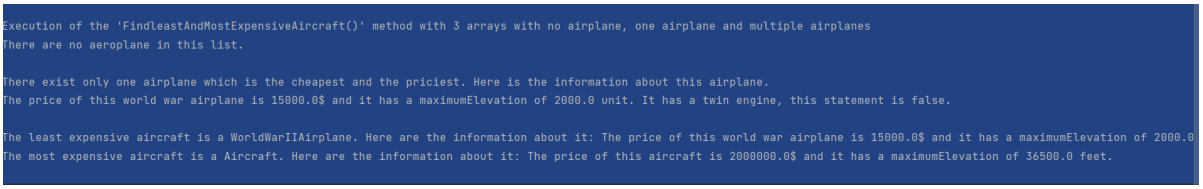
```
static void FindleastAndMostExpensiveAircraft(Object [] arr){
       /**
        * using default constructor to create two new aircraft objects
        */
       Aircraft priciest = new Aircraft();  //new aircraft
       Aircraft cheapest = new Aircraft();  //new aircraft

       double max = 0;
       double min = 0;

       for(int i=0; i< arr.length; i++){
           if(!(arr[i] instanceof Aircraft) & (arr[i].getClass() !=
Aircraft.class)){
               continue;
           }
           if(arr[i] instanceof Aircraft | arr[i].getClass() ==
Aircraft.class){

               Aircraft A = (Aircraft)  arr[i];
               if (max == 0){
                   priciest = A;
                   max = A.getPrice();
               }
              else if(max<A.getPrice()){
                   priciest = A;
                   max = A.getPrice();
               }

              if (min == 0){
                  cheapest = A;
                  min = A.getPrice();
              }
              else if(min>A.getPrice()){
                  cheapest=A;
                  min=A.getPrice();
              }

           }

       }

       //print statements
```

```
        if (max == 0   | min == 0 ){
            System.out.println("There are no aeroplane in this list.");
        return;
        }

        if(cheapest == priciest){
            System.out.println("There exist only one airplane which is the
cheapest and the priciest. Here is the information about this airplane.\n" +
cheapest.toString());
        }
        else {
        System.out.println("The least expensive aircraft is a " +
cheapest.getClass().getSimpleName()+". Here are the information about it: " +
cheapest.toString() + "\nThe most expensive aircraft is a "  +
priciest.getClass().getSimpleName() +". Here are the information about it: "+
priciest.toString());}

}
```

Here two new airplanes were created for cheapest and priciest airplane, and two new variables for the price of the airplane. After going through the loop, the program checks whether each object meets the condition of being an airplane. If they do so, each of the object's price is compared to find the least and the most expensive aircraft. After that, the variables and the two new created objects are updated according to the provided array.

Invocations from 3 different arrays with no airplane, one airplane and multiple airplanes:

```
System.out.println();
Object [] arr0 = {WT1, WT2, T1, T2, M1, M2, TM1, MW1, MW2, F1, F2, WT1, M2,
TM1, F2};
Object [] arr1 = {WT1, WT2, T1, T2, M1, M2, TM1, TM2, MW1, MW2, F1, F2, WW1,
M1, T1, WT2, F2, M1};
Object [] arr2 = {WT1, WW1, T1, T2, M1, AC1, M2, TM1, TM2, MW1, MW2, F1, F2,
AC1, AC2, WW2, AC2 };

System.out.println("Execution of the 'FindleastAndMostExpensiveAircraft()'
method with 3 arrays with no airplane, one airplane and multiple airplanes");
FindleastAndMostExpensiveAircraft(arr0);
System.out.println("");
FindleastAndMostExpensiveAircraft(arr1);
System.out.println("");
FindleastAndMostExpensiveAircraft(arr2);
```



Here is the output of the method where the method is printing according to the given arrays, The program is working correctly because it takes an array of created objects and then it checks for whether they are Airplane objects. If they are not, then the loop is continued until it ends and since the two variables do not get updated and remain 0, the method prints accordingly. Same way, it checks for airplane objects and if it finds any, the variables and two newly created objects are updated and print statements are being executed accordingly.