Tutorial #11: Arrays of Primitive Type Review & Arrays of Objects

Question 1:

What will be the output of the following code?

A.

```
class Parray1
{
   public static void main(String[] args)
   {
      int i;
      int a[] = {5, 2, 3, 1, 1, 0, 2, 1, 0, 1};
      for (i = 0; (i < 10); i++)
      {
         if (a[i] == 0)
            break;
         if (i % 3 == 0)
            continue;
         System.out.println("a[" + i + "]:" + a[i]);
      }
   }
}
```

B.

```
class Parray2
{
   public static void main(String[] args)
   {
      int[] data = {1,3,5,8,11,15};
      int sum = 0;
      for(int i = 1; i < data.length; ++i)
      {
         sum = sum + data[i] - data[i-1];
         System.out.println("sum = " + sum);
      }
   }
}
```

C.

```
class Parray3
{
   public static int sumIf (int[] a, boolean[] b)
   {
      int sum = 0;
      for (int i = 0; i < a.length; ++i)
         if(b[i])
            sum = sum + a[i];
      return sum;
   }

   public static void main(String[] args)
   {
      int[] data = {1, 2, 3, 4, 5, 6, 7};
      boolean[] filter = {true, false, true, true, false, true, true};
      System.out.println("data:" + sumIf(data, filter));
      for(int i = 0; i < filter.length; ++i)
         filter[i] = !filter[i];
         System.out.println("data:" + sumIf(data, filter));
   }
}
```

Question 2:

Write a static method called `sum2` that has two parameters called `row` and `n`. The parameter `row` is an array of numbers of type double; `n` is an integer which will be greater than or equal to 0. The method will return the sum of the first `n` elements of the array `row`.

Question 3:

Assume the following class:

```
public class Airplane {
      private int nbOfPassengers;
      private double weight;
      private int maxSpeed;

       . . .
}
```

A.

Complete the class Airplane by adding

- A default constructor
- A constructor with 2 parameters which will hold data for the `weight` and `maxSpeed` attributes.
- A constructor with 3 parameters.
- All accessor and mutator methods
- An equals method which returns `true` if all attributes are the same and `false` otherwise
- A `toString` such that it returns the string

```
Plane with capacity of <nbOfPassengers> passengers, weighing <weight>kg can travel
at a maximum speed of <maxSpeed>km/hr.
```

where `<nbOfPassengers>` , `<weight>` and `<maxSpeed>` are the values stored in the attributes of an object.

B.

Write a driver class which:

- Contains a static method called `getAverageWeight()` that takes an array of `Airplane` objects as parameter and computes and returns the average weight of the airplanes in the array.
- Contains a static method called `findFasterAirplane()` that takes an array of `Airplane` objects as parameter and returns the fastest airplane in the array. If several airplanes have the same maximum speed, return null.
- Contains a `main()` method which will create an array capable of holding 5 `Airplane` objects using each constructor at least once. Prompt the user for the data for the first 3 objects. For the 4th object ask only for the weight and maximum speed. For the last object in the array use the default constructor.
- Display the content of the array of `Airplane` objects.
- The object that was created with the default constructor, should be assigned attribute values from 3 different objects in the array. (You decide which one).
- Using the `findFasterAirplane()` method find the fastest airplane and display its information.
- Using the `getAverageWeight()` method find the average weight of the airplanes in the array and display its information.
- Compare any 2 objects in the array to see if they are equal using the *equals* method.
- Compare any 2 objects in the array to see if they are equal using the accessor methods (not the equals() methods)
- Have 2 objects contain the same data and then using the *equals* method compare them.