Question - 1
Bike(Avg Price & Name)

Create a class **Bike** with the below attributes:
bikeNo – int
bikeName - String
category - String
price - double

The above attributes should be private, write getters, setters and parameterized constructor as required.

Create a class **Solution** with the main method.
Implement two static methods -
1. findAveragePriceByCategory
2. searchBikeByName in Solution class.

findAveragePriceByCategory
This method will take two input parameters - an array of Bike objects and a String parameter (for the category). The method will return the average price of the Bike from an array of Bike objects. If no Bike with the given category is present in the array of Bike objects, then the method should return 0.

searchBikeByName
Create a static method searchBikeByName in the Solution class. This method will take an array of Bike objects and Name as input and returns the Bike object having the mentioned Name if found else return null if not found.

These methods should be called from the main method.

Write code to perform the following tasks:
1. Take the necessary input variable and call findAveragePriceByCategory.
For this method - The main method should print the average of the given attribute of Bike object as it is if the returned value is greater than 0, or it should print "No Bikes found with the mentioned Category"
2. Take the necessary input variable and call searchBikeByName.
For this method - The main method should print the Bike object details as it is, if the returned value is not null, or it should print "No Bike found with mentioned Name."
Note: All String comparison needs to be case insensitive You can use/refer the below-given sample input and output to verify your solution.

Sample Input (below) Description:
The 1st input taken in the main section is the number of Bike objects to be added to the list of Bike.
The next set of inputs is: bikeNo, bikeName, category, and price for each Bike object taken one after the other and is repeated for the number of Bike objects given in the first line of input.
The last two lines of inputs will be the arguments that need to be passed as a parameter to the methods.

Consider below sample input and output to test your code:

Sample Input 1:
```
4
101
Pulsar
Sports Bike
70000
```

```
102
KTM
Sports Bike
100000
103
Vespa
Scooter
75000
104
Kawasaki
Dirt Bike
73000
Sports Bike
Kawasaki
```

Sample Output 1:

```
Average Price-85000.0
number-104
name-Kawasaki
category-Dirt Bike
price-73000.0
```

Sample Input 2:

```
4
2001
Harley Davidson
Cruiser
175000
2002
BMW
Sports bike
154000
1003
Kawasaki
Cruiser
188000
2004
Suzuki
Cruiser
117000
Cruiser
Jupiter
```

Sample Output 2:

```
Average Price-160000.0
No Bike found with mentioned Name.
```