

Programming Project 2

The second programming project involves writing a program that computes the sales tax for a collection of automobiles of different types. This program consists of four classes. The first class is the `Automobile` class, which contains the automobile's make and model, and purchase price, which is specified in whole dollars. It should have three methods:

1. A constructor that allows the make and purchase price to be initialized.
2. A method named `salesTax` that returns the base sales tax computed as 5% of the sales price.
3. A `toString` method that returns a string containing the make and model of the automobile, the sales price, and the sales tax, appropriately labeled.

The `Automobile` class has two subclasses. The first is `Electric`. It has an additional instance variable that contains its weight in pounds stored as an integer. It should have the same three methods:

1. A constructor that allows the automobile's make and model, purchase price and weight to be initialized.
2. An overridden method `salesTax` that returns the total sales tax. The sales tax for an electric automobile consists of the base sales tax of 5% that applies to all automobiles minus a discount. If the weight is less than 3000 pounds the discount is \$200. Otherwise it is \$150.
3. An overridden `toString` method that returns a string containing the make and model of the automobile, the sales price, sales tax and the weight, appropriately labeled.

The second subclass is `Hybrid`. It has an additional instance variable that contains the number of miles per gallon stored as an integer. It should have the same three methods:

1. A constructor that allows the automobile's make and model, purchase price and miles per gallon to be initialized.
2. An overridden method `salesTax` that returns the total sales tax. The sales tax for a hybrid automobile consists of the base sales tax of 5% minus a discount. If the number of miles per gallon is less than 40, the discount is \$100. Otherwise there is an additional discount of \$2 for every mile per gallon in excess of 40.
3. An overridden `toString` method that returns a string containing the make and model of the automobile, the sales price, sales tax and the number of miles per gallon, appropriately labeled.

No additional public methods should be included in any of the above three classes.

Finally there should be a fourth class named `Project2` that contains the main method. It should generate the GUI shown below:

After entering the information about the make and model, its sales price and the type of the automobile, clicking the *Compute Sales Tax* button should display the sales tax in the text field to its right. In addition, the object should be stored in an array of type `Automobile`. That array should allow space for up to the last five automobiles entered.

If non integer values are entering in any of the fields that require integers, an error message should be displayed in a `JOptionPane` window.

Clicking the *Clear Fields* button should clear all text fields.

Clicking the *Display Report* button should produce a report on the console that includes the information about all the automobiles currently stored in the array. An example of the report that should be displayed is shown below:

```
Make and Model: Toyota Prius
Sales Price: 30000
Sales Tax: 1390.00
Hybrid Vehicle
MPG: 45
```

```
Make and Model: Ford Fusion
Sales Price: 21000
Sales Tax: 1050.00
```

Be sure to follow good programming style, which means making all instance variables private, naming all constants and avoiding the duplication of code. Furthermore you must select enough different kinds of automobiles to completely test the program.

Grading Rubric:

The following grading rubric will be used to determine your grade:

Attribute	Meets
Automobile class	<p>20 points</p> <p>contains the automobile's make and model, and purchase price, which is specified in whole dollars. It should have three methods:</p> <ol style="list-style-type: none">1. A constructor that allows the make and purchase price to be initialized.2. A method named salesTax that returns the base sales tax computed as 5% of the sales price.3. A toString method that returns a string containing the make and model of the automobile, the sales price, and the sales tax, appropriately labeled.
Electric subclass	<p>20 points</p> <p>It has an additional instance variable that contains its weight in pounds stored as an integer. It should have the same three methods:</p> <ol style="list-style-type: none">1. A constructor that allows the automobile's make and model, purchase price and weight to be initialized.2. An overridden method salesTax that returns the total sales tax. The sales tax for an electric automobile consists of the base sales tax of 5% that applies to all automobiles minus a discount. If the weight is less than 3000 pounds the discount is \$200. Otherwise it is \$150.3. An overridden toString method that returns a string containing the make and model of the automobile, the sales price, sales tax and the weight, appropriately labeled.
Hybrid subclass	<p>20 points</p>

	<p>contains the number of miles per gallon stored as an integer. It should have the same three methods:</p> <ol style="list-style-type: none"> 1. A constructor that allows the automobile's make and model, purchase price and miles per gallon to be initialized. 2. An overridden method salesTax that returns the total sales tax The sales tax for a hybrid automobile consists of the base sales tax of 5% minus a discount. If the number of miles per gallon is less than 40, the discount is \$100. Otherwise there is an additional discount of \$2 for every mile per gallon in excess of 40. 3. An overridden toString method that returns a string containing the make and model of the automobile, the sales price, sales tax and the number of miles per gallon, appropriately labeled.
Project 2	<p>20 points</p> <p>contains the main method. It should generate the GUI</p> <p>After entering the information about the make and model, its sales price and the type of the automobile, clicking the Compute Sales Tax button should display the sales tax in the text field to its right. In addition, the object should be stored in an array of type Automobile. That array should allow space for up to the last five automobiles entered.</p> <p>If non integer values are entering in any of the fields that require integers, an error message should be displayed in a JOptionPane window.</p> <p>Clicking the Clear Fields button should clear all text fields.</p> <p>Clicking the Display Report button should produce a report on the console that includes the information about all the automobiles currently stored in the array. An example of the report that should be displayed is shown below:</p> <p>Make and Model: Toyota Prius Sales Price: 30000 Sales Tax: 1390.00</p>

	<p>Hybrid Vehicle MPG: 45</p> <p>Make and Model: Ford Fusion Sales Price: 21000 Sales Tax: 1050.00</p>
Documentation and Style guide	<p>20 points</p> <p>Tests cases provides coverage of code.</p> <p>Screen captures were provided and labeled for compiling your code, and running each of your test cases.</p> <p>Header comments include filename, author, date and brief purpose of the program.</p> <p>In-line comments used to describe major functionality of the code.</p> <p>Meaningful variable names and prompts applied.</p> <p>Class names are written in UpperCamelCase.</p> <p>Variable names are written in lowerCamelCase.</p> <p>Constant names are in written in All Capitals.</p> <p>Braces use K&R style.</p> <p>Declare all instance variables private.</p> <p>Avoids the duplication of code.</p>