


To make these options actually mean anything you will need to implement the logic behind them. Let's examine them.


1. Start the car: Starting the car as a first option makes sense, can't go anywhere without starting up first! But how do you know that the car is started? You need something to hold that value or state. Since this is a yes/no situation, you can use a boolean to hold that value and that becomes the first variable to be declared.
2. Select a gear: If the car is going to move then it has to be in the correct gear and there are three to choose from, but only two of them should allow the car to move, D (drive) and R (reverse). You need a variable to hold that value and since this is a text value you can use a String to hold the value.
3. Accelerate: You need to know your current speed before you can display the accelerated speed, so you need a variable to hold the speed value. The option to accelerate increases the speed, so you would add a set amount to the speed variable every time a user chooses that option. Since this is a number value you can use an int to store it.
4. Brake: When you use the brake the car slows down, so you only need to subtract a set amount from the speed variable to apply the brakes.
5. Exit: When the user selects this option the program will display a goodbye message and end.


You now have almost all the variables you need to be able to make the car function, but you also need a way to talk to your user. You need to declare a Scanner object to allow input from the keyboard. You will also need to capture the choice the user makes, and since this will be a menu number, you can also use an int to store this value.

 Note: When you encounter this icon, it's time to get into your IDE and start coding!

In your lab environment, open IntelliJ by double-clicking on the icon.

It's time to start the first task. Navigate to the *Main.java* file to write the Java program.

 It's time to get coding!

-  TODO 1: In the main method, declare and initialize the five variables to:
 - track if the car is on or off (Boolean)
 - hold the gear of the car (P, D, R)
 - track the current speed of the car (int)
 - store the user's menu choice (int)
 - a scanner object to read the user's input. (don't forget to import!)

```
boolean isEngineOn = false;
```


```
String gear = "P";


int speed    = 0;

int choice   = 0;

Scanner keyBoard = new Scanner(System.in);
```

You're off to a great start! The next step is to provide feedback to the user.

 It's time to get coding!

-  **TODO 2:** Press Enter a few times to create empty lines in the program.
 - Add print statements to indicate the current state of the car to the user. Note that this should be the first print statement displayed.


```
System.out.println("----- Car Dashboard -----");
```


```
System.out.println("Speed: " + speed);
```

Complete the print statements for the remaining variables you want to display.

HINT: What else do you normally see in a car dashboard that is also a declared variable above?

The next thing is that the user also needs to be informed about the options you have decided to give them, so you need to display a menu for them to choose from, your program has 5 options.

 It's time to get coding!


-  **TODO 3:** Add print statements to create the display menu for options available to the user.

```
System.out.println("Menu:");
```

```
System.out.println("1. Turn on/off the engine");
```

```
System.out.println("2. Change gear (P, D, R)");
```

Complete the menu using print statements to display them to your user. Remember to use the **Scanner**.


-  **TODO 4:** Prompt the user for their choice and store it in the **choice** variable.


```
choice = keyBoard.nextInt();
```

```
System.out.print("Enter your choice: ");
```

At this stage, it's a good idea to test-run your program (add a print statement to display the choice made) and make sure things are displaying the way you want them.

Each option you are displaying to your user needs to be backed up by logic and you need to display that reasoning to your user. The message you display along with the values of the variables that accompany it are going to be based on the menu choice that your user makes, so every choice needs to be checked to decide what to do. The menu has 5 options, so this can best be accomplished using a **switch** statement. The user enters a menu number, you capture that number in your **choice** variable and check this value in a **switch** statement.

 It's time to get coding!

-  **TODO 5:** Insert a switch statement to provide five menu options to the user. The user enters a menu number, which is captured in the **choice** variable. The value is then checked in a **switch** statement.

```
switch (choice) {

    case 1:

        isEngineOn = !isEngineOn;

        break;

    case 2:

        System.out.print("Enter gear (P, D, R): ");

        gear = keyBoard.next();

        break;

    case 3:

        if (isEngineOn && !gear.equals("P")) {

            speed += 10; // speed = speed+10


        } else {


            System.out.println("Cannot accelerate while engine is off or in Park (P) gear.");
```

```
}  
  
break;
```

Complete the remaining code to accommodate menu choices. Be careful and pay particular attention to case 4, this should follow the logic from case 3. You need to check that the engine is on and that the car is moving.

Test run your code, you will notice that the program only runs for one menu choice and then ends, you need to make it so the program will keep running until the user decides it's time for it to stop. This sounds like a good time for a while loop. Surround all of your code (not your variable declarations!) with a while loop, you can set the while condition to correspond to the menu choice of **Exit**. This also means that your **switch** does not need an exit option as choice will be checked to determine if the loop will run.

 It's time to get coding!

-  **TODO 6:** Insert a while loop to surround all of your code (not your variable declarations!). You can set the while condition to correspond to the menu choice of "Exit". This also means that your **switch** does not need an exit option as choice will be checked to determine if the loop will run.
 - Add a final print statement indicating that the simulator is shutting down.

```
while (choice !=5) {
```

Now that you have written a car simulator program, it's time to put your application to the test.

1. Run your code using the IDE.
2. Attempt to accommodate all possible manner of choices that a user may make. Remember your user didn't write the program and they only know what you tell them through your print statements.
3. Compare the output of your program to the expected results. For example:
 - a. What happens if the car is not moving and the user tries to apply the brakes?
 - b. Does the car have a top speed or is it 'to infinity and beyond'?