Solution Review: Playing with Cars and Engines!

This lesson provides the solution to the challenge, "Playing with Cars and Engines!" with an explanation.

Solution

```java
// Car class
class Car {

  // Declaring data members and methods
  private int id;
  private String model;
  private String color;

  public void carFeatures() { // Function to print out car features
    System.out.println("Car Model: " + model);
    System.out.println("Car Color: " + color);
  }

  public void setModel(String model) {
    this.model = model;
  }

  public void setColor(String color) {
    this.color = color;
  }

}

// Toyota Class, which is a child class of Car class.
class Toyota extends Car {

  // Inherits all properties of Car class
  public void setStart() {
    // Declaring an engine object and calling the start() function of the engine.
    ToyotaEngine engine = new ToyotaEngine();
    engine.start();
  }

}
```

```java
// Engine class
class ToyotaEngine {

  // This function simply prints out on screen that Engine has been started!
  public void start() {
    System.out.println("Engine has been started.");
  }

  // This function simply prints out on screen that Engine has been stopped!
  public void stop() {
    System.out.println("Engine has been stopped.");
  }

}

class Main {

  public static void main(String[] args) {
    // Declaring and initializing Toyota object
    Toyota t = new Toyota();
    t.setModel("Fortuner");
    t.setColor("Silver");

    t.carFeatures();
    t.setStart();
  }

}
```

Explanation

- Line **25**: Extended `Toyota` class from `Car` class.
- Line **28**: In the `setStart()` function, we've declared an instance of `ToyotaEngine()` which is now bounded with the instance of `Toyota` class.
- Line **55**: We are declaring a `Toyota` object, named **t**.
- Line **59**: Calling `carFeatures()` of the **"t"** object, after initializing its *model* and its *color* above.
- Line **60**: We are then calling the `start()` function of `ToyotaEngine()` through the instance of `Toyota` i.e. **t**.