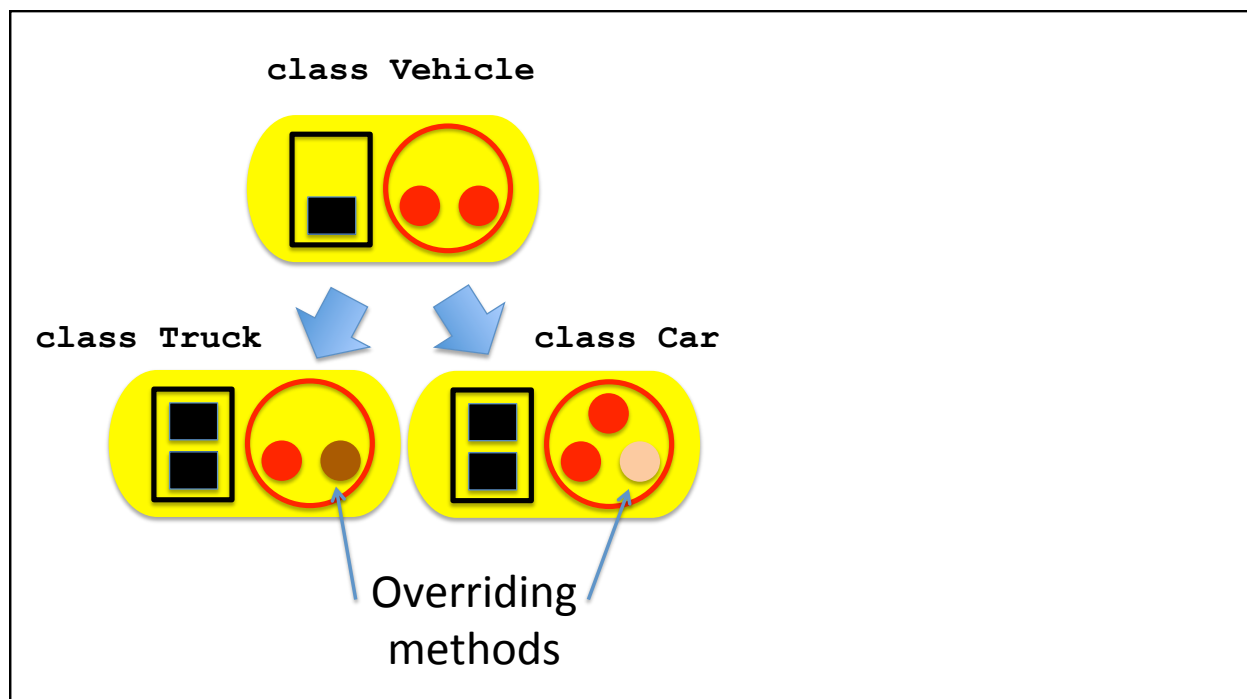
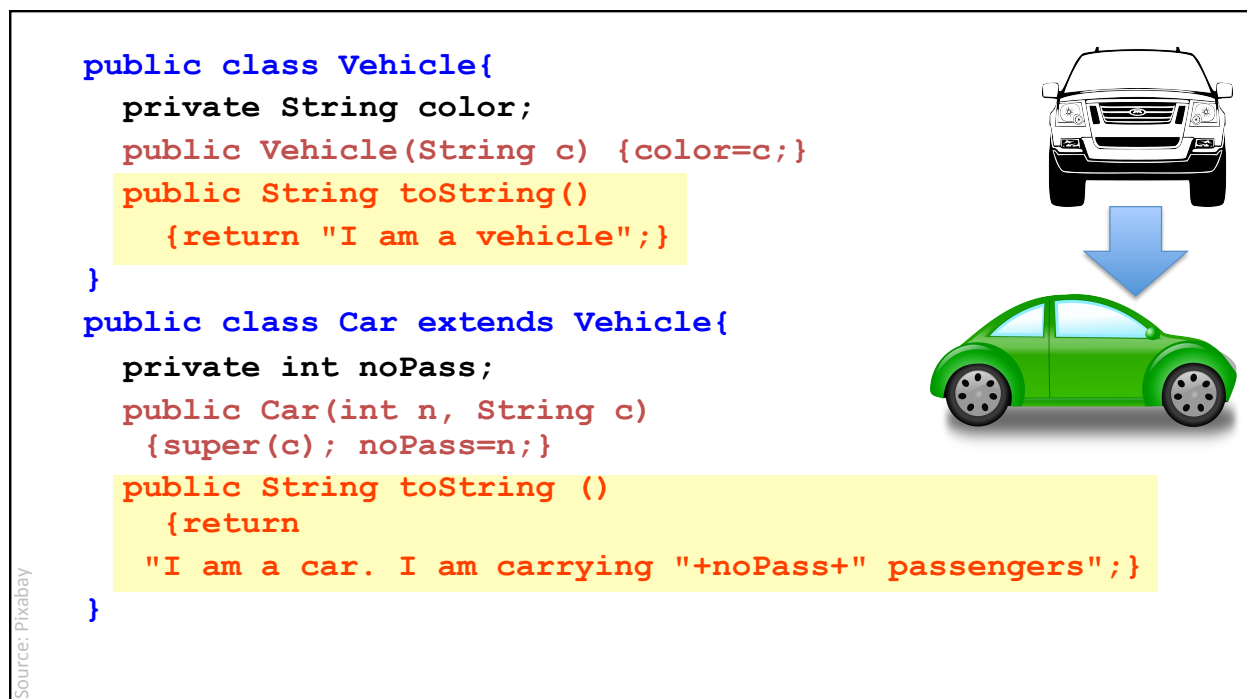
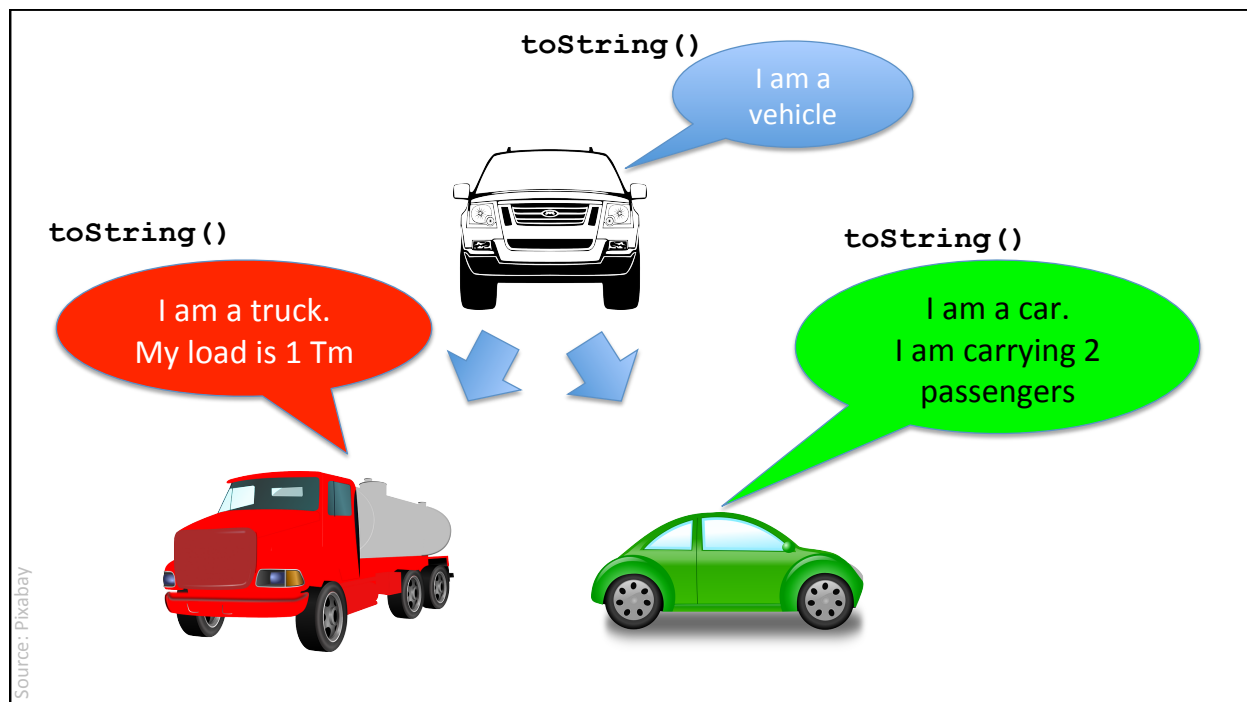
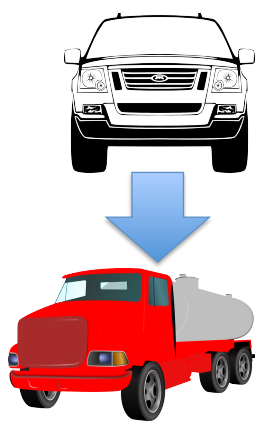


ABSTRACT METHODS and CLASSES



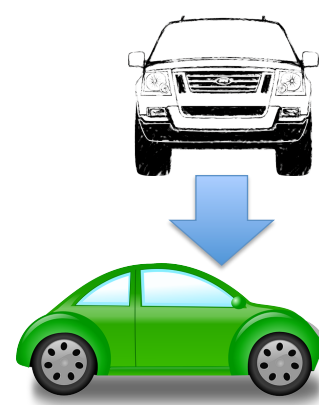


```
public class Vehicle{
    private String color;
    public Vehicle(String c) {color=c;}
    public String toString()
        {return "I am a vehicle";}
}
public class Truck extends Vehicle{
    private int load;
    public Truck(int n, String c)
        {super(c); load=n;}
    public String toString ()
        {return
            "I am a truck. My load is "+load+" Tm";}
}
```



Source: Pixabay

```
public abstract class Vehicle{
    private String color;
    public Vehicle(String c) {color=c;}
    public abstract String toString();
    ... // other methods
}
public class Car extends Vehicle{
    private int noPass;
    public Car(int n, String c)
        {super(c); noPass=n;}
    public String toString ()
        {return
            "I am a car. I am carrying "+noPass+" passengers";}
}
```



Source: Pixabay

Abstract Method



- Method without body
- `public abstract String toString();`
- Constructors, static methods, final methods cannot be abstract

Source: Pixabay

Abstract Class



- Class where **some** methods are abstract and **some** are not
- `public abstract class Vehicle{...}`
- An abstract class cannot be instantiated
- An abstract class can be extended to a **class** or to an **abstract class**

Source: Pixabay

