Polymorphism and Abstract Classes

Refer to the code shown below in answering the following questions.

```java
public abstract class Vehicle {
    private double price;
    private int years;

    public void setPrice(double price) {
        this.price = price;
    }

    public void setYears(int years) {
        this.years = years;
    }

    public double getPrice() { return this.price; }
    public int getYears() { return this.years; }

    public abstract double value(double amount);
}

public class Bicycle extends Vehicle {
    private int gears;

    public Bicycle(int years, double price, int gears) {
        this.setYears(years);
        this.setPrice(price);
        this.gears = gears;
    }

    public int getGears(){
        return this.gears;
    }

    public double value(double amount) {
        return this.getPrice() - amount;
    }
    ...
}

public class Car extends Vehicle{
    private int passengers;

    public Car(int years, double price, int people){
        this.setYears(years);
        this.setPrice(price);
        passengers = people;
    }

    public int getPassengers(){
        return this.passengers;
    }
    ...
}
```
1. Implement the `value()` method for the `Car` class according to the following:
   - The **loss in value** for a `Car` object is equal to the amount passed as an argument to this method times the age of the car (i.e., years).
   - The value returned by this method is the **price** of the `Car` object minus its **loss in value** if that amount is greater than 0. Otherwise, the value returned is 0.

A class called `MyVehicles` is in the same package as the `Vehicle`, `Car`, and `Bicycle` classes. Assume all of the code shown below is in the `main()` method of the `MyVehicles` class.

2. For each of the following code segments, state whether or not the code will compile and, if it compiles, if it will run without throwing an exception. Then briefly explain your answers.

```java
Vehicle v1 = new Vehicle();

Compiles? Runs?:
Explanation:
```

```java
Car c = new Car(10, 9000, 6);
Vehicle v = new Car(6, 8800, 4);
c = v;
```

Compiles? Runs?:
Explanation:

```java
Bicycle bike = new Bicycle(3, 275, 20);
System.out.println(bike.getGears());
```

Compiles? Runs?:
Explanation: