Fantasy Artifact Creation with Java

Estimated time: 1 hour
Goal: Learn about classes and objects in Java through creating and bringing your own magical artifact to life with your code!
Actively participate in this magical artifact creation adventure to get 1 point

Introduction

Welcome to the Realm of Enchantments where everyone of you possess the ability to create one magical artifact, creature, or item. Today, you will design your own unique magical creation and give it life through the power of Java programming.

Your creation can be **ANYTHING** you imagine:
- A legendary weapon (fire sword, frost bow, thunder staff)
- A magical creature (baby dragon, mystic cat, crystal fox)
- An enchanted item (magic potion, infinite backpack, time-traveling watch)
- A fantastical being (phoenix, unicorn, fairy)

Go wild and creative!

Part 1: Design Phase (5-10 minutes)

Before you start jumping into coding, plan your creation. You can either write down or sketch:

1) What is your creation?
2) What properties does it have? Think of 3-4 characteristics:
    - Physical traits (color, size, material)
    - Magical properties (power level, element type, special abilities)
    - Personality traits

3) What can it do? Think of 2-3 actions or behaviours:

- Can it glow, fly, speak, attack, heal or transform?
- What happens when the user interacts with it?

Part 2: Coding Phase (30 - 60 minutes)

Basic Requirements

You need to create TWO separate files:

File 1: YourCreationName.java (Your Class)

Your class must include:

1) At least 3 attributes that describe your creation

- Use private access modifier
- Choose appropriate data types (String, int, boolean, etc.)

2) A constructor that initializes all attributes

- Takes parameters for at least 3 attributes
- Assigns parameter values to instance variables

3) At least 2 methods that make your creation "do something"

- These should print descriptive messages
- Can change attribute values or perform actions
- Must be meaningful to your creation

File 2: App.java (Main Program)

Your App.java file must include:

1. A main method that:
   - Creates at least one instance of your creation class
   - Calls all the action methods to demonstrate what it can do
   - Shows your creation with clear console output

Make sure to include comments in the beginning of each method & class.

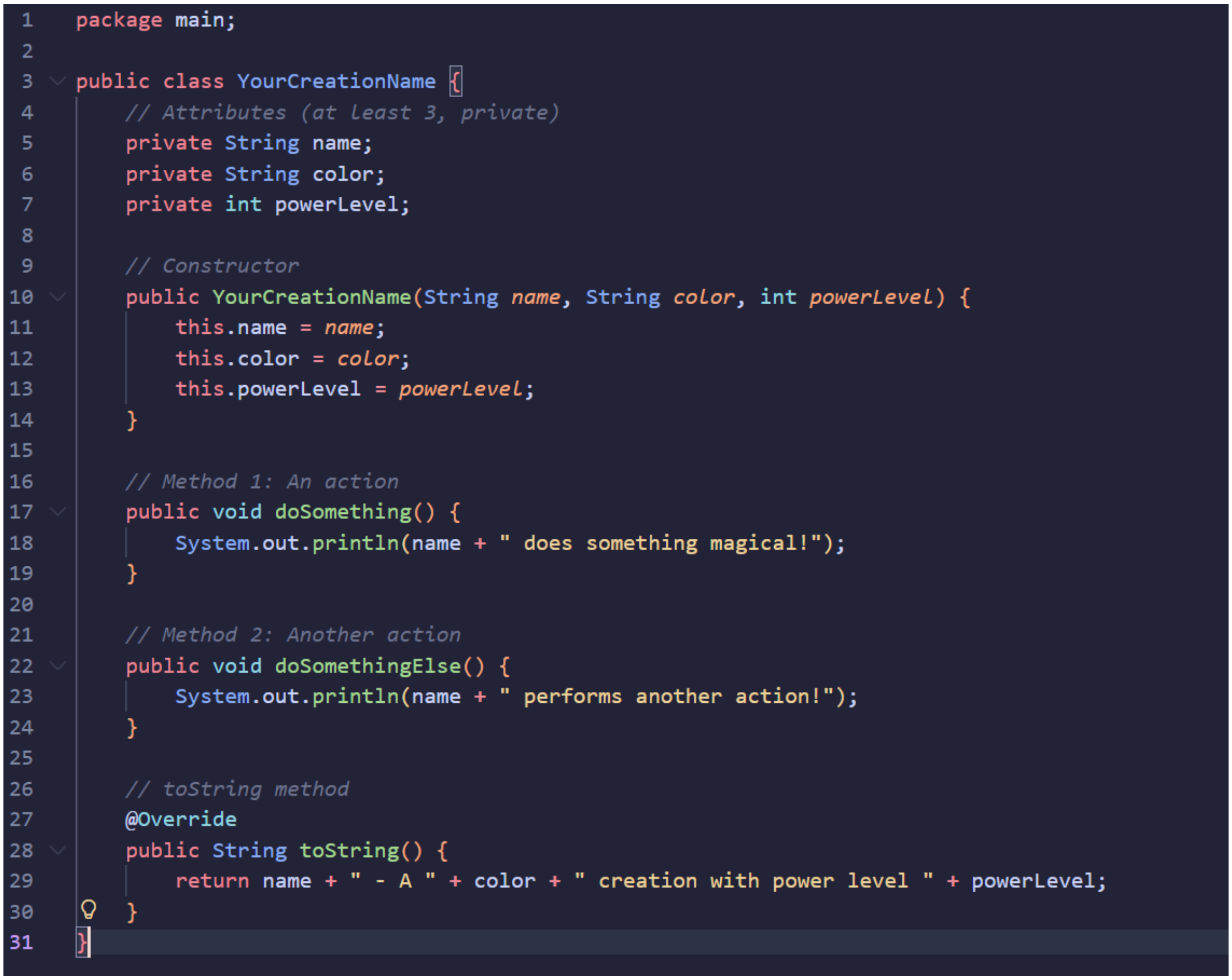
Bonus Ideas

In case you're way too OP (overpowered) and want to add extra touches to your magical creation, you can do these bonuses. Note that they do not give you extra points, unfortunately.

- Add a method that modifies an attribute (e.g. levelUp(), changeColor(), repair())
- Create multiple instances of your creation with various properties in main
- Add conditional logic in a method (if statements based on attribute values)
- Add a method that takes a parameter (like castSpell(String target) or fly(int distance))

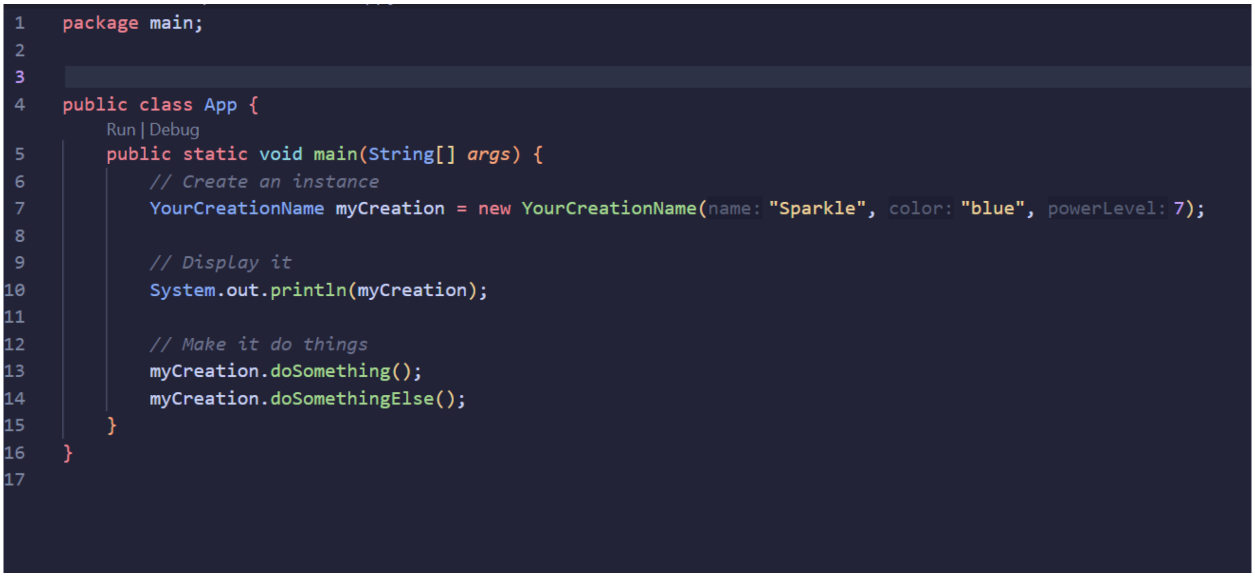
Example Basic Code Structure

YourCreationName.java:



```java
package main;

public class YourCreationName {
    // Attributes (at least 3, private)
    private String name;
    private String color;
    private int powerLevel;

    // Constructor
    public YourCreationName(String name, String color, int powerLevel) {
        this.name = name;
        this.color = color;
        this.powerLevel = powerLevel;
    }

    // Method 1: An action
    public void doSomething() {
        System.out.println(name + " does something magical!");
    }

    // Method 2: Another action
    public void doSomethingElse() {
        System.out.println(name + " performs another action!");
    }

    // toString method
    @Override
    public String toString() {
        return name + " - A " + color + " creation with power level " + powerLevel;
    }
}
```

App.java:



```java
package main;


public class App {
    public static void main(String[] args) {
        // Create an instance
        YourCreationName myCreation = new YourCreationName(name: "Sparkle", color: "blue", powerLevel: 7);

        // Display it
        System.out.println(myCreation);

        // Make it do things
        myCreation.doSomething();
        myCreation.doSomethingElse();
    }
}
```