



Prepare

Certify

Compete

Apply



Search



Prepare > Java > Object Oriented Programming > Java Abstract Class

Java Abstract Class ★

10 more points to get your next star!

Rank: 735016 | Points: 70/80

**Your Java Abstract Class submission got 10.00 points.**[Share](#)[Post](#)

You are now 10 points away from the 3rd star for your java badge.

[Try the next challenge](#) | [Try a Random Challenge](#)

Problem

Submissions

Leaderboard

Discussions

Editorial

A Java abstract class is a class that can't be instantiated. That means you cannot create new instances of an abstract class. It works as a base for subclasses. You should learn about Java Inheritance before attempting this challenge.

Following is an example of abstract class:

```
abstract class Book{
    String title;
    abstract void setTitle(String s);
    String getTitle(){
        return title;
    }
}
```

If you try to create an instance of this class like the following line you will get an error:

```
Book new_novel=new Book();
```

You have to create another class that extends the abstract class. Then you can create an instance of the new class.

Notice that setTitle method is abstract too and has no body. That means you must implement the body of that method in the child class.

In the editor, we have provided the abstract Book class and a Main class. In the Main class, we created an instance of a class called MyBook. Your task is to write just the MyBook class.

Your class mustn't be public.

Sample Input

```
A tale of two cities
```

Sample Output

```
The title is: A tale of two cities
```

Author

Shafaet

Difficulty

Easy

Max Score

10

Submitted By

223458

NEED HELP?

[View discussions](#)[View editorial](#)[View top submissions](#)

RATE THIS CHALLENGE



MORE DETAILS

[Download problem statement](#)[Download sample test cases](#)[Suggest Edits](#)

[Change Theme](#)

Language

Java 7



```
1  import java.util.*;
2  abstract class Book{
3      String title;
4      abstract void setTitle(String s);
5      String getTitle(){
6          return title;
7      }
8  }
9
10 //Write MyBook class here
11 class MyBook extends Book {
12     @Override
13     void setTitle(String s) {
14         title = s; // Set the title in the Book's 'title' field
15     }
16 }
17
18 public class Main{
19
20     public static void main(String []args){
21         //Book new_novel=new Book(); This line prHMain.java:25:
22         error: Book is abstract; cannot be instantiated
23         Scanner sc=new Scanner(System.in);
24         String title=sc.nextLine();
25         MyBook new_novel=new MyBook();
26         new_novel.setTitle(title);
27         System.out.println("The title is: "+new_novel.getTitle());
28         sc.close();
29     }
30 }
```

Line: 16 Col: 2

[Upload Code as File](#)☐[Test against custom input](#)[Run Code](#)[Submit Code](#)

You have earned 10.00 points!

You are now 10 points away from the 3rd star for your java badge.

67%

70/80



Congratulations

You solved this challenge. Would you like to challenge your friends?

[Next Challenge](#)

 **Test case 0**

Compiler Message

Success **Test case 1** **Test case 2** 

Input (stdin)

[Download](#)1 | **A tale of two cities**

Expected Output

[Download](#)1 | **The title is: A tale of two cities**[Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#)