Abstract Classes in Java

In C++, if a class has at least one pure virtual function, then the class becomes abstract. Unlike C++, in Java, a separate keyword *abstract* is used to make a class abstract.

```
// An example abstract class in Java
abstract class Shape {
    int color;

    // An abstract function (like a pure virtual function in C++)
    abstract void draw();
}
```

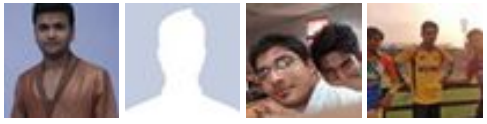
Following are some important observations about abstract classes in Java.

1) Like C++, in Java, an instance of an abstract class cannot be created, we can have references of abstract class type though.

```
abstract class Base {
    abstract void fun();
}
class Derived extends Base {
    void fun() { System.out.println("Derived fun() called"); }
}
class Main {
    public static void main(String args[]) {

        // Uncommenting the following line will cause compiler error a
        // line tries to create an instance of abstract class.
        // Base b = new Base();

        // We can have references of Base type.
        Base b = new Derived();
        b.fun();
```

Interview Experiences
Advanced Data Structures
Dynamic Programming
Greedy Algorithms
Backtracking
Pattern Searching
Divide & Conquer
Mathematical Algorithms
Recursion

```
    }
}
```

Output:

```
Derived fun() called
```

2) Like C++, an abstract class can contain constructors in Java. And a constructor of abstract class is called when an instance of a inherited class is created. For example, the following is a valid Java program.

```
// An abstract class with constructor
abstract class Base {
    Base() { System.out.println("Base Constructor Called"); }
    abstract void fun();
}
class Derived extends Base {
    Derived() { System.out.println("Derived Constructor Called"); }
    void fun() { System.out.println("Derived fun() called"); }
}
class Main {
    public static void main(String args[]) {
        Derived d = new Derived();
    }
}
```

Output:

```
Base Constructor Called
Derived Constructor Called
```

3) In Java, we can have an abstract class without any abstract method. This allows us to create classes that cannot be instantiated, but can only be inherited.

```
// An abstract class without any abstract method
abstract class Base {
    void fun() { System.out.println("Base fun() called"); }
}

class Derived extends Base { }

class Main {
```

Geometric Algorithms

Popular Posts

All permutations of a given string
Memory Layout of C Programs
Understanding “extern” keyword in C
Median of two sorted arrays
Tree traversal without recursion and without stack!
Structure Member Alignment, Padding and Data Packing
Intersection point of two Linked Lists
Lowest Common Ancestor in a BST.
Check if a binary tree is BST or not
Sorted Linked List to Balanced BST

```
    public static void main(String args[]) {
        Derived d = new Derived();
        d.fun();
    }
}
```

Output:

```
Base fun() called
```

4) Abstract classes can also have final methods (methods that cannot be overridden). For example, the following program compiles and runs fine.

```
// An abstract class with a final method
abstract class Base {
    final void fun() { System.out.println("Derived fun() called"); }
}

class Derived extends Base {}

class Main {
    public static void main(String args[]) {
       Base b = new Derived();
       b.fun();
    }
}
```

Output:

```
Derived fun() called
```

Exercise:

1. Is it possible to create abstract and final class in Java?

2. Is it possible to have an abstract method in a final class?

3. Is it possible to inherit from multiple abstract classes in Java?

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

Related Tpoics:

- Backtracking | Set 8 (Solving Cryptarithmetic Puzzles)
- Tail Recursion
- Find if two rectangles overlap
- Analysis of Algorithm | Set 4 (Solving Recurrences)
- Print all possible paths from top left to bottom right of a mXn matrix
- Generate all unique partitions of an integer
- Russian Peasant Multiplication
- Closest Pair of Points | O(nlogn) Implementation

Writing code in comment? Please use **ideone.com** and share the link here.

9 Comments GeeksforGeeks

Sort by Newest

Recent Comments

Abhi You live US or India?

Google (Mountain View) interview · 35 minutes ago

Aman Hi, Why arent we checking for conditions...

Write a C program to Delete a Tree. · 1 hour ago

kzs please provide solution for the problem...

Backtracking | Set 2 (Rat in a Maze) · 1 hour ago

Sanjay Agarwal bool tree::Root_to_leaf_path_given_sum(tree...

Root to leaf path sum equal to a given number · 1 hour ago

GOPI GOPINATH @admin Highlight this sentence "We can easily...

Count trailing zeroes in factorial of a number · 1 hour ago

newCoder3006 If the array contains negative numbers also. We...

Find subarray with given sum · 2 hours ago

Join the discussion…

puneet k agarwal • 19 days ago

Excersie:

1. No we cannot mark an abstract class as final
2. No we cant have a abstract method in final class as final class not be sub c
3. In java we can not have multiple inheritance due to the ambiguity problem th bodies

Reply • Share ›

Ghazanfar Ali Bhayo • 3 months ago

how to extend Super classes in abstract class????????

Reply • Share ›

vignesh ➔ Ghazanfar Ali Bhayo • 2 months ago

yeah Ghazanfar Ali Bhayo you can make your abstract class to extend

```
class A
{
public int a;
}
abstract class AbsA extends A
{
void assign_a()
{
a=10;
}
}
class MainClass extends AbsA
{
psvm(String args[])
{
```

```
AbsA obj=new Mainclass();
obj.assign_a();
sysout(a);
}
}
```

Reply • Share ›

Mani Deepak Vandrangi • 11 months ago

we need not write "abstract" keyword for abstract methods.plz correct me if I a

Reply • Share ›

puneet k agarwal Mani Deepak Vandrangi • 19 days ago

Yes you are correct. To make any method abstract you have to mark c

Reply • Share ›

Sandeep Jadhav Mani Deepak Vandrangi • 5 months ago

To specify a method as Abstract ,,, keyword is mandatory . take any ex

Reply • Share ›

Mani Deepak Vandrangi • 11 months ago

we need not write "abstract" keyword for abstract methods.plz correct me if I a

1 Reply • Share ›

mani • 11 months ago

we need not write "abstract" keyword for abstract methods.plz correct me if I a

Reply • Share ›

wgpshashank • a year ago

1. Is it possible to create abstract and final class in Java?
No,

When a class is declared final,it cannot be extended or subclassed.If any atte

compilation error .

An abstract class can never be instantiated.Its main mission is to be sub-class
abstract, the whole class must be declared abstract.However you can have n
class.You cannot create an object out of abstract classes,which will result in
compilation error, thus its duty of subclass to give definition of superclass's m
abstract, not following such result in complication error

So now if one declare the class both final and abstract then both will contradic
final an d abstract .

2. Is it possible to have an abstract method in a final class?

NO , since if you declare abstract method in class must be declared abstract
info please read above reasoning .

3. Is it possible to inherit from multiple abstract classes in Java?

No, Java Doesn't support multiple inheritance , but you can achieve this by imp

Feel free to comment !!!

9 ▲ | ▼ • Reply • Share ›