

Java 9 Underscore

In earlier versions of Java, underscore can be used as identifier and to create variable name also. But in Java 9 release, underscore is a keyword and can't be used as an identifier or variable name.

If we use the underscore character ("_") as an identifier, our source code can no longer be compiled.

Let's see some examples that explain, how the use of underscore is changed version after after.

In Java 7, we can use underscore like the following.

Java 7 Underscore Example

```
public class UnderScoreExample {  
    public static void main(String[] args) {  
        int _ = 10; // creating variable  
        System.out.println(_);  
    }  
}
```

And it produce the output without any warning and error.

Output:

```
10
```

Java 8 Underscore Example

If we compile the same program in Java 8, it will compile but throws a warning message.

```
public class UnderScoreExample {  
    public static void main(String[] args) {  
        int _ = 10;  
        System.out.println(_);  
    }  
}
```

Output:

```
UnderScoreExample.java:3: warning: '_' used as an identifier  
        int _ = 10;  
            ^  
(use of '_' as an identifier might not be supported in releases after Java SE 8)
```

Java 9 Underscore Example

In Java 9, program fails to compile and throws compile time error because now it is a keyword and can't be use as a variable name.

```
public class UnderScoreExample {  
    public static void main(String[] args) {  
        int _ = 10;  
        System.out.println(_);  
    }  
}
```

Output:

```
UnderScoreExample.java:3: error: as of release 9, '_' is a keyword, and may not be used as an identifier  
        int _ = 10;
```