

ClassNotFoundException Vs NoClassDefFoundError

In Java, both **ClassNotFoundException** and **NoClassDefFoundError** occur when a particular class is not found at run time. But, they occur at different scenarios. **ClassNotFoundException** is an exception which occurs when you try to load a class at run time using **Class.forName()** or **loadClass()** methods and mentioned classes are not found in the classpath. On the other hand, **NoClassDefFoundError** is an error which occurs when a particular class is present at compile time but it was missing at run time. In this tutorial, we will see the differences between ClassNotFoundException Vs NoClassDefFoundError in java and when they occur.

ClassNotFoundException In Java :

ClassNotFoundException is a run time exception which is thrown when an application tries to load a class at run time using **Class.forName()** or **loadClass()** or **findSystemClass()** methods and the class with specified name are not found in the classpath. For example, you may have come across this exception when you try to connect to MySQL or Oracle databases and you have not updated the classpath with required JAR files. In most of time, this exception occurs when you try to run an application without updating the classpath with required JAR files.

For example, below program will throw ClassNotFoundException if the mentioned class "**oracle.jdbc.driver.OracleDriver**" is not found in the classpath.

```
1
2     public class MainClass
3     {
4         public static void main(String[] args)
5         {
6             try
7             {
8                 Class.forName("oracle.jdbc.driver.OracleDriver");
9             }
10            catch (ClassNotFoundException e)
11            {
12                e.printStackTrace();
13            }
14        }
15    }
```

If you run the above program without updating the classpath with required JAR files, you will get the exception like below,

```
1     java.lang.ClassNotFoundException: oracle.jdbc.driver.OracleDriver
2     at java.net.URLClassLoader.findClass(Unknown Source)
3     at java.lang.ClassLoader.loadClass(Unknown Source)
4     at sun.misc.Launcher$AppClassLoader.loadClass(Unknown Source)
5     at java.lang.ClassLoader.loadClass(Unknown Source)
```

```
5 at java.lang.Class.forName0(Native Method)
6 at java.lang.Class.forName(Unknown Source)
7 at pack1.MainClass.main(MainClass.java:17)
8
```

NoClassDefFoundError In Java :

NoClassDefFoundError is an error which is thrown when Java Runtime System tries to load the definition of a class and class definition is no longer available. The required class definition was present at compile time but it was missing at run time. For example, compile the below program.

```
1
2 class A
3 {
4 }
5 public class B
6 {
7     public static void main(String[] args)
8     {
9         A a = new A();
10    }
```

When you compile the above program, two .class files will be generated. One is **A.class** and another one is **B.class**. If you remove the **A.class** file and run the **B.class** file, Java Runtime System will throw NoClassDefFoundError like below,

```
1 Exception in thread "main" java.lang.NoClassDefFoundError: A
2 at MainClass.main(MainClass.java:10)
3 Caused by: java.lang.ClassNotFoundException: A
4 at java.net.URLClassLoader.findClass(URLClassLoader.java:381)
5 at java.lang.ClassLoader.loadClass(ClassLoader.java:424)
6 at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:331)
7 at java.lang.ClassLoader.loadClass(ClassLoader.java:357)
```

Below is the quick recap of above findings.

Difference Between ClassNotFoundException Vs NoClassDefFoundError In Java :

ClassNotFoundException	NoClassDefFoundError
It is an exception. It is of type java.lang.Exception.	It is an error. It is of type java.lang.Error.
It occurs when an application tries to load a class at run time which is not	It occurs when java runtime system doesn't find a class definition, which is

updated in the classpath.

present at compile time, but missing at run time.

It is thrown by the application itself. It is thrown by the methods like `Class.forName()`, `loadClass()` and `findSystemClass()`.

It is thrown by the Java Runtime System.

It occurs when classpath is not updated with required JAR files.

It occurs when required class definition is missing at run time.

ClassNotFoundException

It occurs when an application tries to load a class at run time which is not updated in the class path.

It is of type `java.lang.Exception`.

It is an exception.

It occurs If class path is not updated with required JAR files.

It is thrown by the methods like `Class.forName()`, `loadClass()` and `findSystemClass()`.

NoClassDefFoundError

It occurs when java runtime system failed to load a class definition, which is present at compile time but missing at run time.

It is of type `java.lang.Error`

It is an Error.

It occurs when required class definition is missing at run time.

It is thrown by the Java Runtime System.