

# Properties class in Java

The **properties** object contains key and value pair both as a string. The `java.util.Properties` class is the subclass of `Hashtable`.

It can be used to get property value based on the property key. The `Properties` class provides methods to get data from properties file and store data into properties file. Moreover, it can be used to get properties of system.

## Advantage of properties file

**Recompilation is not required, if information is changed from properties**

**file:** If any information is changed from the properties file, you don't need to recompile the java class. It is used to store information which is to be changed frequently.

## Methods of Properties class

The commonly used methods of `Properties` class are given below.

Method	Description
<code>public void load(Reader r)</code>	loads data from the Reader object.
<code>public void load(InputStream is)</code>	loads data from the InputStream object
<code>public String getProperty(String key)</code>	returns value based on the key.
<code>public void setProperty(String key,String value)</code>	sets the property in the properties object.
<code>public void store(Writer w, String comment)</code>	writes the properties in the writer object.
<code>public void store(OutputStream os, String comment)</code>	writes the properties in the OutputStream object.

<code>storeToXML(OutputStream os, String comment)</code>	writers the properties in the writer object for generating xml document.
<code>public void storeToXML(Writer w, String comment, String encoding)</code>	writers the properties in the writer object for generating xml document with specified encoding.

## Example of Properties class to get information from properties file

To get information from the properties file, create the properties file first.

### db.properties

```
user=system
password=oracle
```

Now, lets create the java class to read the data from the properties file.

### Test.java

```
import java.util.*;
import java.io.*;
public class Test {
    public static void main(String[] args)throws Exception{
        FileReader reader=new FileReader("db.properties");

        Properties p=new Properties();
        p.load(reader);

        System.out.println(p.getProperty("user"));
        System.out.println(p.getProperty("password"));
    }
}
```

```
Output:system
        oracle
```

Now if you change the value of the properties file, you don't need to compile the java class again. That means no maintenance problem.

## Example of Properties class to get all the system properties

By `System.getProperties()` method we can get all the properties of system. Let's create the class that gets information from the system properties.

### Test.java

```
import java.util.*;
import java.io.*;

public class Test {
    public static void main(String[] args)throws Exception{

        Properties p=System.getProperties();
        Set set=p.entrySet();

        Iterator itr=set.iterator();
        while(itr.hasNext()){
            Map.Entry entry=(Map.Entry)itr.next();
            System.out.println(entry.getKey()+" = "+entry.getValue());
        }

    }
}
```

#### Output:

```
java.runtime.name = Java(TM) SE Runtime Environment
sun.boot.library.path = C:\Program Files\Java\jdk1.7.0_01\jre\bin
java.vm.version = 21.1-b02
java.vm.vendor = Oracle Corporation
java.vendor.url = http://java.oracle.com/
path.separator = ;
java.vm.name = Java HotSpot(TM) Client VM
file.encoding.pkg = sun.io
user.country = US
user.script =
sun.java.launcher = SUN_STANDARD
.....
```

## Example of Properties class to create properties file

Now lets write the code to create the properties file.

### Test.java

```
import java.util.*;
import java.io.*;

public class Test {
    public static void main(String[] args)throws Exception{

        Properties p=new Properties();
        p.setProperty("name","Sonoo Jaiswal");
        p.setProperty("email","sonoojaiswal@javatpoint.com");

        p.store(new FileWriter("info.properties"),"Javatpoint Properties Example");

    }
}
```

Let's see the generated properties file.

### info.properties

```
#Javatpoint Properties Example
#Thu Oct 03 22:35:53 IST 2013
email=sonoojaiswal@javatpoint.com
name=Sonoo Jaiswal
```

[< prev](#)[next >](#)

## Share this page



## Latest 4 Tutorials



CouchDB



Docker



Rails



RichFaces