

## Apache ANT

### zip, unzip, OS family

```
<project name="test" default="zip">

    <target name="zip">
        <delete file="build.zip" />
        <zip basedir="build" destfile="build.zip"/>
        <delete dir="build" />
        <unzip src="build.zip" dest="build"/>

        <antcall target="checkOS"/>
    </target>

    <!--checking if for windows and non windows OS-->
    <target name="checkOS" depends="ifOSWindows, ifOSNotWindows" />

        <condition property="isOSUnix">
            <os family="unix" />
        </condition>

        <condition property="isOSWindows">
            <os family="windows" />
        </condition>

    <!-- if the OS is windows call the target run.script.windows -->
    <target name="ifOSWindows" if="isOSWindows">
        <echo>is windows.....</echo>
        <antcall target="runscript.windows"/>
    </target>

    <!-- if the OS is not windows call the target run.script.unix-->
    <target name="ifOSNotWindows" if="isOSUnix">
        <echo>is unix.....</echo>
        <antcall target="runscript.unix"/>
    </target>

    <target name="runscript.windows">
        <echo>This is a Windows machine.</echo>
        <exec dir="." executable="cmd" >
            <arg line="/c dir > dir_info"/>
        </exec>
    </target>

</project>
```

```

    <target name="runscript.unix">
      <echo>This is an Unix machine.</echo>
      <exec dir="." executable="/bin/sh">
        <arg line="-c ls -lrth > dir_list"/>
      </exec>
    </target>
  </project>

```

## OUTPUT

```

E:\Ant\Ant Exercises\Exercise7>ant -buildfile zip_unzip.xml
Buildfile: E:\Ant\Ant Exercises\Exercise7\zip_unzip.xml

zip:
  [delete] Deleting: E:\Ant\Ant Exercises\Exercise7\build.zip
  [zip] Building zip: E:\Ant\Ant Exercises\Exercise7\build.zip
  [delete] Deleting directory E:\Ant\Ant Exercises\Exercise7\build
  [unzip] Expanding: E:\Ant\Ant Exercises\Exercise7\build.zip into E:\Ant\Ant
Exercises\Exercise7\build

ifOSWindows:
  [echo] is windows.....

runscript.windows:
  [echo] This is a Windows machine.

ifOSNotWindows:

checkOS:

BUILD SUCCESSFUL
Total time: 1 second

```

## Mailing

```

<project name="test" default="test">

  <taskdef resource="net/sf/antcontrib/antcontrib.properties" />

  <target name="test">

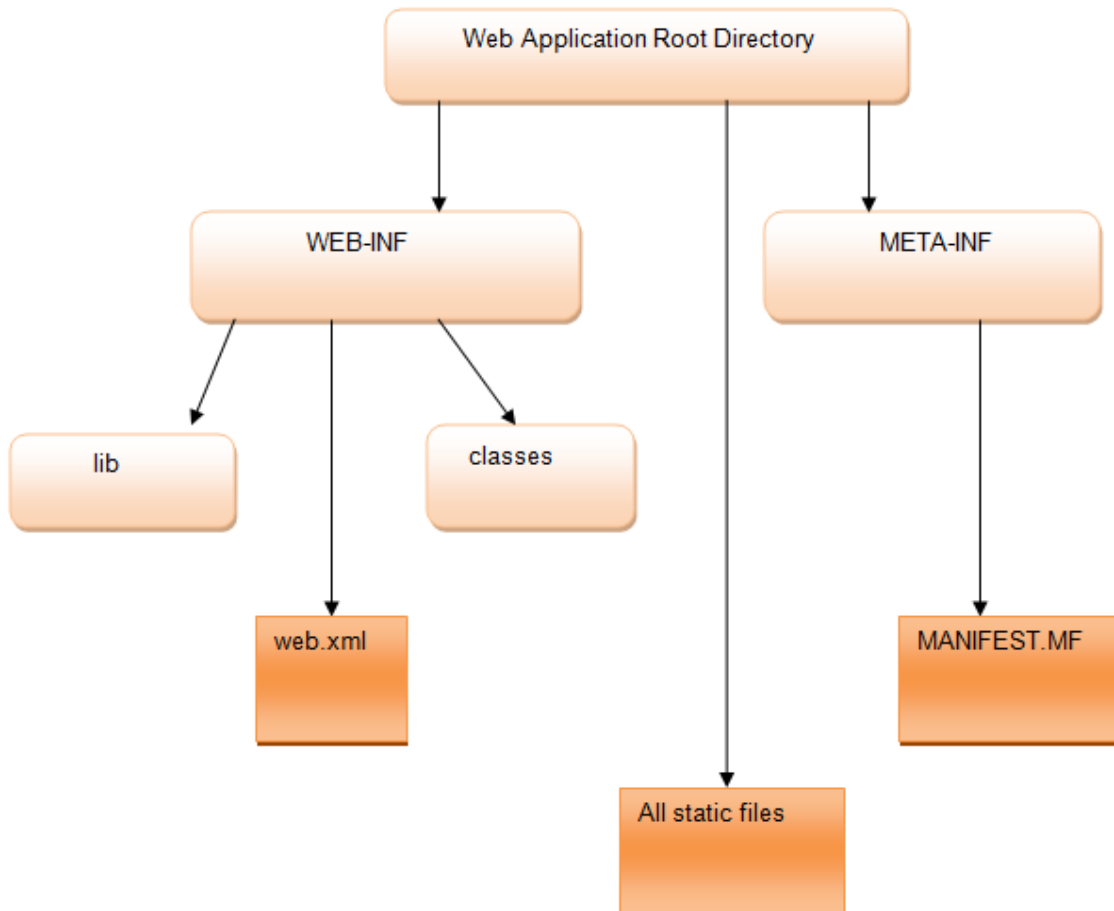
    <mail mailhost="smtp.gmail.com" mailport="465" subject="Test
Mail from Ant Build"
user="capitalinfosol@gmail.com " password="passwod"
from="capitalinfosol@gmail.com" tolist="capitalinfosol@gmail.com"
ssl="true" encoding="mime" messagemimetype="text/html">

    <message>Test Mail from Ant Build</message>
    <attachments>
      <fileset dir="src">

```

```
<include name="**/*.java"/>
</fileset>
</attachments>
</mail>
</target>
</project>
```

### Web Project File Structure



## Details –

**Web Application Root Directory** – This is the main or Root folder of web application. Usually name of this folder becomes your web application context. For example ,if our web application name is FirstWebApplication , then folder name will be FirstWebApplication and web application will be accessible via <http://localhost:8080/FirstWebApplication>

**WEB-INF**- This is the special directory under web application root directory. This is special because this is secured folder and files available within this folder will not be accessible to client directly. Which means say if this directory has one file “index.html” then this file cannot be accessed directly via <http://localhost:8080/FirstWebApplication/index.html>

**WEB-INF/lib**- All the required jar files or third party jar files will be placed inside this directory.

**WEB-INF/classes**- All the java code including servlets for the web application will go inside classes folder. We can also put our own classes into a JAR file, and place it here, rather than putting those classes in the classes directory.

**web.xml** – web.xml resides under WEB-INF folder and it is also known as deployment descriptor. All the configuration of web application like servlets configuration , filters configuration , welcome file list etc are configured in web.xml.

**META-INF/MAINIFEST.MF**- This is the manifest file

**Static Files** – All static files like HTML, css ,javascript will be placed directly under web application root directory. If we want to make these files secure, we need to place these files under WEB-INF directory.