

COLLECTIONS

```
<%@ page import="java.util.*" %>
<%@ page import="netcharts.server.api.*" %>
<%@ page import="netcharts.server.util.*" %>
<%
    try
    {
        // You can declare and initialize a collection this way...
        NSWebToolKit objTK[] = new NSWebToolKit[5];
        objTK[1] = new NSWebToolKit("aaMMOChartspre" + "\\\" + "Testbed");

        /*
         *
         */

        // Or this way
        NSWebToolKit objTKauto[] =
        {
            new NSWebToolKit("aaMMOChartspre" + "\\\" + "Testbed"),
            new NSWebToolKit("aaMMOChartspre" + "\\\" + "Testbed"),
            new NSWebToolKit("aaMMOChartspre" + "\\\" + "Testbed"),
            new NSWebToolKit("aaMMOChartspre" + "\\\" + "Testbed"),
            new NSWebToolKit("aaMMOChartspre" + "\\\" + "Testbed")
        };

        out.println("Either way, it works.");
    }
    catch(Exception e)
    {
    }
%>
```

JAVA [OPTIONS] CLASSNAME [PARMS]

JAVAC [OPTIONS] FILENAME.JAVA

VARIABLES \Rightarrow CALL STACK
OBJECTS \Rightarrow GC HEAP



BUILDING AN OBJECT MODEL : LINKING TOP-LEVEL OBJECTS TO INNER OBJECTS

parent

```
public class objTopLevel extends Object
{
    parm1 lngParm;
    public objTopLevel(parm1 lngPassThru)
    {
        lngParm = lngPassThru;
    }
}
```



child

```
public class objChild extends objTopLevel
{
    private boolean boolIsEmpty;
    public objChild()
    {
        this(null);
    }
    public objChild(parm1 lngPassThru)
    {
        super(lngPassThru);
    }
}
```