Experiment-8

1.1 Objective:

Application development using Enterprise Java Beans.

1.2 Learning Outcomes:

At the end learners will be able to:

- Create an EJB project.
- Understand the meaning of Stateless, Stateful and Singleton beans.
- Capable of using Jboss 7.1 application server and explain its importance.
- Explain the meaning of deploying the project in EJB environment.
- Understand and explain the process of server configuration and setting class path as appropriate.
- Create interfaces, client and server as appropriate for EJB application.

1.3 Resources required

• JDK, Net Beans, Eclipse, JBoss 7.1 application server.

1.4 Configuring JBoss7.1 Server

Goto server explorer and right click

Choose the server (I have selected JBoss As 7.1 since I have it in my machine)

Define a New Server Choose the type of server to	create	
Select the server type:	Download additional server a	dapter
type filter text		
JBoss AS 5.0 JBoss AS 5.1 JBoss AS 6.x JBoss AS 7.0 JBoss AS 7.1 JBoss AS 7.1	1	
Server's host name:	localhost	
Server name:	JBoss 7.1 Runtime Server (1)	
Server runtime environment:	JBoss 7.1 Runtime 👻	Add.
	Configure runtime environn	nents.
	Configure runtime environn	<u>nents</u>

Click "next"

New Server		
Create a new JBoss JBoss Application Se	s Server erver 7.1	Boss [®]
A JBoss Server manag It manages comman	ges starting and stopping instances of JBoss. Id line arguments and keeps track of which modules ha	ve been deployed.
If the runtime information and then Add to cre	mation below is incorrect, please press back, Installed R eate a new runtime from a different location.	untimes,
Home Directory	C:\Java\jboss-as-7.1.1.Final	
Execution Environm	nent Java Platform, Standard Edition 6.0	
JRE	Default JRE for JavaSE-1.6	
Server Behaviour Server is external Listen on all inte Expose your man	Ily managed. Assume server is started. erfaces to allow remote web connections nagement port as the server's hostname	
?	< Back Next > Finish	Cancel

Hit "<u>Next</u>"

gured on the server	
gure them on the server Configured:	
Add >	
< Remove	
Add All >>	
<< Remove All	
Next > Finish	Cancel
	gured on the server gure them on the server Configured: Add > < Remove Add All >> << Remove All Next > Finish

Both lists will be empty in the beginning if you are configuring the server first time or if there is no previous project available in the system.

Now, hit on "Finish"



Now, right click of JBoss7.1 RuntimeServer and choose start option.

public class EJBApplica ⊖ public static void HelloWorldBeanR	ti ma em	New Open Show In	► F3 Alt+Shift+W ►
System.out.prin }	tl	Сору	Ctrl+C
private static Hell Context context HelloWorldBeanR	ow m	Paste Delete Rename	Ctrl+V Delete F2
try { // 1. Obtai	ni 🎄	Debug	Ctrl+Alt+D
context = C	li 🚺	Start	Ctrl+Alt+R
String look	up 🔊	Profile	
bean = (Hel	10	Stop	Ctrl+Alt+S
} catch (Naming e.printStac	Ex kT	Publish Clean	Ctrl+Alt+P
}	Re-	Add and Remove	
(_	Monitoring	•
🗱 Servers 🖾 📮 Console > 🏭 JBoss 7.1 Runtime Server [Sto	op	Properties	Alt+Enter

Once, you will choose the start option from the context menu then your application server will start running and synchronized with the projects you are intended to develop.

If you will click of console then you can see the logs of JBoss 7.1 as enumerated below:

```
11:55:06,217 INFO [org.jboss.modules] JBoss Modules version 1.1.1.GA
11:55:06,704 INFO [org.jboss.msc] JBoss MSC version 1.0.2.GA
11:55:06,782 INFO [org.jboss.as] JBAS015899: JBoss AS 7.1.1.Final "Brontes" starting
11:55:08,357 INFO [org.jboss.as.server] JBAS015888: Creating http management service
using socket-binding (management-http)
11:55:08,400 INFO [org.xnio] XNIO Version 3.0.3.GA
11:55:08,411 INFO [org.xnio.nio] XNIO NIO Implementation Version 3.0.3.GA
11:55:08,421 INFO [org.jboss.remoting] JBoss Remoting version 3.2.3.GA
11:55:08,639 INFO [org.jboss.as.configadmin] JBAS016200: Activating ConfigAdmin
Subsystem
```

11:55:08,645 INFO [org.jboss.as.logging] JBAS011502: Removing bootstrap log handlers 11:55:08,691 INFO [org.jboss.as.connector.subsystems.datasources] (ServerService Thread Pool -- 27) JBAS010403: Deploying JDBC-compliant driver class org.h2.Driver (version 1.3) 11:55:08,733 INFO [org.jboss.as.security] (ServerService Thread Pool -- 44) JBAS013101: Activating Security Subsystem 11:55:08,813 INFO [org.jboss.as.security] (MSC service thread 1-4) JBAS013100: Current PicketBox version=4.0.7.Final 11:55:08,908 INFO [org.jboss.as.osgi] (ServerService Thread Pool -- 39) JBAS011940: Activating OSGi Subsystem 11:55:08,933 INFO [org.jboss.as.webservices] (ServerService Thread Pool -- 48) JBAS015537: Activating WebServices Extension 11:55:09,094 INFO [org.jboss.as.clustering.infinispan] (ServerService Thread Pool --31) JBAS010280: Activating Infinispan subsystem. 11:55:09,108 INFO [org.jboss.as.connector] (MSC service thread 1-4) JBAS010408: Starting JCA Subsystem (JBoss IronJacamar 1.0.9.Final) 11:55:09,163 INFO [org.jboss.as.naming] (ServerService Thread Pool -- 38) JBAS011800: Activating Naming Subsystem 11:55:09,238 INFO [org.jboss.as.naming] (MSC service thread 1-4) JBAS011802: Starting Naming Service 11:55:09,322 INFO [org.jboss.as.mail.extension] (MSC service thread 1-4) JBAS015400: Bound mail session [java:jboss/mail/Default] 11:55:09,599 INFO [org.jboss.ws.common.management.AbstractServerConfig] (MSC service thread 1-1) JBoss Web Services - Stack CXF Server 4.0.2.GA 11:55:09,892 INFO [org.jboss.as.server.deployment.scanner] (MSC service thread 1-1) JBAS015012: Started FileSystemDeploymentService for directory C:\Java\jboss-as-7.1.1.Final\standalone\deployments 11:55:09,948 WARN [org.jboss.as.server.deployment.scanner] (DeploymentScannerthreads - 1) JBAS015002: Deployment of 'GeniusHari3' requested, but the deployment is not present 11:55:09,950 WARN [org.jboss.as.server.deployment.scanner] (DeploymentScannerthreads - 1) JBAS015002: Deployment of 'EJBTesterLocal.java' requested, but the deployment is not present 11:55:09,955 WARN [org.jboss.as.server.deployment.scanner] (DeploymentScannerthreads - 1) JBAS015002: Deployment of 'EJBTester.java' requested, but the deployment is not present 11:55:09,957 WARN [org.jboss.as.server.deployment.scanner] (DeploymentScannerthreads - 1) JBAS015002: Deployment of 'DemoEJBEnterpriseProject.ear' requested, but the deployment is not present 11:55:09,959 INFO [org.jboss.as.server.deployment.scanner] (DeploymentScannerthreads - 1) JBAS015003: Found Demo1DynamicWebProject.war in deployment directory. To trigger deployment create a file called Demo1DynamicWebProject.war.dodeploy 11:55:09,961 INFO [org.jboss.as.server.deployment.scanner] (DeploymentScannerthreads - 1) JBAS015003: Found HelloWorldSessionBean.jar in deployment directory. To trigger deployment create a file called HelloWorldSessionBean.jar.dodeploy 11:55:09,963 ERROR [org.jboss.as.server.deployment.scanner] (DeploymentScannerthreads - 1) JBAS015010: The deployment scanner found a directory named META-INF that was not inside a directory whose name ends with .ear, .jar, .rar, .sar or .war. This is likely the result of unzipping an archive directly inside the C:\Java\jboss-as-7.1.1.Final\standalone\deployments directory, which is a user error. The META-INF directory will not be scanned for deployments, but it is possible that the scanner mayfind other files from the unzipped archive and attempt to deploy them, leading to errors. 11:55:09,967 ERROR [org.jboss.as.server.deployment.scanner] (DeploymentScannerthreads - 1) JBAS015010: The deployment scanner found a directory named WEB-INF that

was not inside a directory whose name ends with .ear, .jar, .rar, .sar or .war. This is likely the result of unzipping an archive directly inside the C:\Java\iboss-as-7.1.1.Final\standalone\deployments directory, which is a user error. The WEB-INF directory will not be scanned for deployments, but it is possible that the scanner mayfind other files from the unzipped archive and attempt to deploy them, leading to errors. 11:55:10,030 INFO [org.jboss.as.remoting] (MSC service thread 1-4) JBAS017100: Listening on localhost/127.0.0.1:4447 11:55:10,070 INFO [org.apache.coyote.http11.Http11Protocol] (MSC service thread 1-3) Starting Coyote HTTP/1.1 on http-localhost-127.0.0.1-8080 11:55:10,237 INFO [org.jboss.as.connector.subsystems.datasources] (MSC service thread 1-3) JBAS010400: Bound data source [java:jboss/datasources/ExampleDS] 11:55:10,349 INFO [org.jboss.as.remoting] (MSC service thread 1-4) JBAS017100: Listening on /127.0.0.1:9999 11:55:10,691 INFO [org.jboss.as] (Controller Boot Thread) JBAS015951: Admin console listening on http://127.0.0.1:9990 11:55:10,705 INFO [org.jboss.as] (Controller Boot Thread) JBAS015874: JBoss AS 7.1.1.Final "Brontes" started in 4972ms - Started 133 of 208 services (74 services are passive or on-demand)

Project Descriptor

- We are going to create a simple EJB HelloWorld Stateless bean project and a remote Java application client which will call/invoke the bean.
- This is "HelloWorld" example explain the how to develop, deploy and run EJB3 Session Bean (Stateless and Statefull) in JBoss application server.
- For testing this "HelloWorld" example we write a remote Java Application client (main() method)
- For simplicity, the session bean and the client to access the session bean are created in the same project.

Creating New EJBProject

- Open Eclipse IDE and create a new EJB project which can be done in three ways
 - Right click on Project Explorer \rightarrow New \rightarrow EJBProject
 - File menu \rightarrow New \rightarrow EJB Project
 - Click on the down arrow on New icon on toolbar \rightarrow EJB project

2	File->	New	_		
	-		3	New icon in toolbar	
			-		
Edt	Nau otarch	Project Run Windo		Help	
4	000.	0-4- 3-		- 004- 0 2 C- + 20 5 9-5	
nect	Explorer C.L.				
	-	8 2 4			
				The second second method is not a second sec	
			1	Project Explorer	
			1	Project Explorer	
	-		1	Project Explorer	
	-		1	Project Explorer	
	New		1	Project Explorer	
	New Show In	AR-Shift-W+	1	Project Explorer	
	New Show In		1	Project Explorer Project. Application Client Project Connector Project	
	New Show In Copy Constituted bit	AR-SMR-W- CH-C	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Project Explorer Project. Project. Project. Consector Project Consector Project Opnamic Wab Project Dynamic Wab Project	
0.68	New Show In Copy Copy Qualified No Perform	AR-SMR-W+ Col-C	1 0 1 4 4 9 4	Project Explorer Project Application Client Project Consector Project Consector Project Consector Project EB Project EB Project	
(9) (1)	New Show In Copy Copy Qualified Re Patte Instea	Alte-Shifte-W + Colle-C me Children	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Project Explorer Project Interview Project Interview Project P	
x965	New Show In Copy Copy Qualified Na Paste Dates	Alt-Shife-W + Col-C Col-C Col-V Color	L PAGEOR	Project Explorer Project Explorer Application Client Project Consector Project Dynamic Web Project Etil Project Etil Project Static Web Project Static Web Project	
×9 65	New Show In Copy Copy Qualified His Patte Delote Delote Seport	AR-SAR-W+ ColeC ColeC ColeC ColeX Detes	0 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	Project Explorer Project Explorer Application Client Project Connector Project Connector Project Eth Project Eth Project Eth Project Static Web Project Static Web Project Static Web Project	

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	New Open Open Open With Copy Paste Delete Move Rename Import Export Export Refresh Unmark as Deployable	JPA Project Enterprise Application Project Dynamic Web Project EJB Project Connector Project Application Client Project Static Web Project Project Servlet Session Bean (EJB 3.x) Message-Driven Bean (EJB 3.x) Web Service Folder	
-41	Validate Show in Remote Systems view Run As Debug As	File Example Other	Ctrl+N

• Enter the project name as "HelloWorldSessionBean" and make sure the JBoss7.1 Runtime has been selected with EJB 3.0 Module version.

New BB Project	
JB Project Create an EJB Project and add it to a new or poisting Enterprise Applicatio	n. 🌔
Project name: HelloWorldSessionBean	
Project location	
Use default location	
Location: C:\Users\iByteCode\Desktop\EJ83JBoss7.1CR1BCodes\Hellc	Browge
Target runtime	
JBoss7.1 Runtime 👻	New <u>Runtime</u>
EIB module version	
3.0	-
Configuration	
<custom></custom>	Modify
Hint: Get started quickly by selecting one of the pre-defined project cor	figurations.
EAR membership	
Add project to an EAR	
EAR project name: HelloWorldSessionBeanEAR +	New Project
Working sets	
Add project to working sets	
Working sets:	Select
© theopentuto	rials.com

- Click Next \rightarrow Next \rightarrow and Finsh
- You will see an EJB project in the Project Explorer view.



Creating Session Bean and Bean Interface

• Right click on EJBModule \rightarrow New \rightarrow Session Bean (EJB 3.x)

🔁 Navigator 🕅	- 8)	DemoEJBSession	fullBe	🚺 HelloWorldBean.java 🗍	HelloW
	⇔⇔@ 🖪 🕏 🏱	remote.conne	ction	provider.create.options.org.	knio.Opt
▲ 送 HelloWor ▷ ➢ .settin	ldSessionBean Igs	remote.conne	ction	s=default	
	New		•	JPA Project	
X .	Go Into		8	Enterprise Application Project	
⊿ 🗁 Servi	Сору			EJB Project	
	Paste			Connector Project Application Client Project	
× . ×	Delete Move Rename			Static Web Project Project	
2 2	Import Export		6	Servlet Session Bean (EJB 3.x)	
8	Refresh			Message-Driven Bean (EJB 3.x) Web Service	
ال≉	Mark as Deployable			Folder	
	Validate Show in Remote Systems view			Example	
	Debug As			Other C	trl+N

- Enter Java package name as com.ADJP.GeniusHari
- Enter the class name HelloWorldBean
- Select the State type as Stateless

- Check the Remote Business Interface and enter the name as com.ADJP.GeniusHari.HelloWorldBeanRemote
- Click Finish

Specify class f	ile destination.	
<u>P</u> roject:	HelloWorldSessionBean	-
Source fol <u>d</u> er:	/HelloWorldSessionBean/ejbModule	Br <u>o</u> wse
Java pac <u>k</u> age:	com.ADJP.GeniusHari	Browse
Class na <u>m</u> e:	HelloWorldBean	
Superclass:		Brows <u>e</u>
State type:	Stateless	-
Create busin	ess interface	
Remote	com.ADJP.GeniusHari.HelloWorldBeanRemote	
Local	com.ADJP.GeniusHari.HelloWorldBeanLocal	
No-interfa	ace	

Coding Bean and the Interface

- Open Bean Interface and type the following code and save the file
- Interface can be either @Remote or @local. In this example we have used @Remote.

```
package com.ADJP.GeniusHari3;
import javax.ejb.Remote;
```

@Remote

public interface HelloWorldBeanRemote { public String SayHello();



- Open Bean and type the following code and save file
- Bean type can either be @Stateless or @Stateful. In this example we have used @Stateless

```
package com.ADJP.GeniusHari3;
import com.ADJP.GeniusHari3.HelloWorldBeanRemote;
import javax.ejb.Stateless;
/**
* Session Bean implementation class HelloWorldBean
```

*/ @Stateless public class HelloWorldBean implements HelloWorldBeanRemote {

```
/**

* Default constructor.

*/

public HelloWorldBean() {

}

public String SayHello()

{

return "Hello World";
```



Now, the stateless session bean has been created. The next step is to deploy the bean on the server.

Deploying EJB project

- Now, we need to deploy the Stateless session bean "HelloWorldBean" on server (JBoss 7.1)
- Deploying the project can be done in two ways:
- Right click on the EJB project →Run As → Run on Server. Select the existing "JBoss 7.1 Runtime Server" and click Finish.
- Right click on JBoss 7.1 Runtime Server" available in Server view → Add and Remove... → Select the EJB JAR file from the left pane and click Add → and then Finish.



Start/Restart Server

Right click on JBoss 7.1 Runtime Server from servers view and click on Start if it has not yet been started.

If the project is deployed properly with global JNDI mapping then you will see the following message in the console.

Creating Client

- The next step is to write a remote Java client application (with main()) for accessing and invoking the EJBs deployed on the server
- Client uses for a proxy of your bean and invokes method on that proxy.

Creating JNDI InitialContext

Obtaining a Context using InitialContext

- All naming service operations are performed on some implementation of the javax.naming.Context interface. Therefore, the starting point of interacting with the naming service is to obtain a Context by providing the properties specific to the server implementation being used. In our case it is, JBoss Application Server.
- To create a javax.naming.InitialContext, we need to initialize it with properties from the environment. JNDI verifies each property's value by merging the values from the following two sources,
 - Using parameterized constructor of InitialContext which takes properties of supplied environment
 - jndi.properties resource files found on the classpath.

The following utility class is used to create InitialContext for JBoss AS and can be reused in all applications. Otherwise the code written in this class should be repeated in all clients.

- Right click on EJBModule -> New -> Class
- Enter the package name as **com.ADJP.ClientHari**
- Enter the Class name as **ClientUtility**
- Click on Finish

💽 New Java Class	2 million 2 married and	- · · ×
Java Class Create a new Java o	class.	C
Source folder:	HelloWorldSessionBean/ejbModule	Browse
Package:	com.ADJP.ClientHari	Browse
Enclosing type:		Browse
Superclass:	abstract final static	Browse
Interfaces:		Add
		Remove
which method stub	s would you like to create?	
	Constructors from superclass	
	Inherited abstract methods	
Do you want to add	comments? (Configure templates and default value here)	
	Cenerate comments	

```
package com.ADJP.ClientHari;
import java.util.Properties;
import javax.naming.Context;
import javax.naming.InitialContext;
import javax.naming.NamingException;
public class ClientUtility {
    private static Context initialContext;
    private static final String PKG INTERFACES =
"org.jboss.ejb.client.naming";
    public static Context getInitialContext() throws NamingException {
        if (initialContext == null) {
            Properties properties = new Properties();
            properties.put(Context.URL_PKG_PREFIXES, PKG_INTERFACES);
            initialContext = new InitialContext(properties);
        }
        return initialContext;
    }
}
```



Creating client class

- Right click on ejbModule -> New -> Class
- Enter the package name as **com.ibytecode.client**
- Enter the Class name as **EJBApplicationClient**
- Check the main() method option
- Click on Finish

lava Class		P
Create a new Java c	:lass.	G
Source folder:	HelloWorldSessionBean/ejbModule	Browse
Package:	com.ADJP.client	Browse
Enclosing type:		Browse
Name:	EJBApplicationClient1	
Modifiers:	public Odefault Oprivate Oprotected abstract Infinal static	
Superclass:	java.lang.Object	Browse
Interfaces:		Add
Which mathed stub		n
which method stub	would you like to create:	
	Constructors from superclass	
	Inherited abstract methods	
Do you want to add	comments? (Configure templates and default value here)	

package com.ADJP.client; import javax.naming.Context; import javax.naming.NamingException; import com.ADJP.GeniusHari3.HelloWorldBeanRemote; import com.ADJP.GeniusHari3.HelloWorldBean; import com.ADJP.ClientHari.ClientUtility;

public class EJBApplicationClient {

```
public static void main(String[] args) {
    HelloWorldBeanRemote bean = doLookup();
    System.out.println(bean.SayHello()); // 4. Call business logic
}
```

```
private static HelloWorldBeanRemote doLookup() {
    Context context = null;
    HelloWorldBeanRemote bean = null;
    try {
        // 1. Obtaining Context
        context = ClientUtility.getInitialContext();
        // 2. Generate JNDI Lookup name
        String lookupName = getLookupName();
        // 3. Lookup and cast
        bean = (HelloWorldBeanRemote) context.lookup(lookupName);
    } catch (NamingException e) {
        e.printStackTrace();
    }
    return bean;
}
```

```
private static String getLookupName() {
```

```
/*
```

The <u>app</u> name is the EAR name of the deployed EJB without .ear suffix. Since we haven't deployed the application as a .ear, the <u>app</u> name for us will be an empty string */

```
String appName = "";
```

/* The module name is the JAR name of the deployed EJB
without the .jar suffix.
*/
String moduleName = "HelloWorldSessionBean";

/*AS7 allows each deployment to have an (optional) distinct name. This can be an empty string if distinct name is not specified. */

String distinctName = "";

// The EJB bean implementation class name
String beanName = HelloWorldBean.class.getSimpleName();

```
// Fully qualified remote interface name
final String interfaceName = HelloWorldBeanRemote.class.getName();
```

```
// Create a look up string name
String name = "ejb:" + appName + "/" + moduleName + "/" +
distinctName + "/" + beanName + "!" + interfaceName;
```

return name;

}

Setting up EJB client context properties

An EJB client context is a context which contains contextual information for carrying out remote invocations on EJBs. This is a JBoss AS specific API. The EJB client context can be associated with multiple EJB receivers. Each EJB receiver is capable of handling invocations on different EJBs.

Create a file "**jboss-ejb-client.properties**" in the classpath of the application. We can place it in ejbModule folder of our application. The jboss-ejb-client.properties contains the following properties:

remote.connectionprovider.create.options.org.xnio.Options.SSL_ENABLED=false

remote.connections=default

remote.connection.default.host=localhost
remote.connection.default.port = 4447
remote.connection.default.connect.options.org.xnio.Options.SASL_POLICY_NOANONYMOUS=fa
lse



Adding JAR files required for the client to run the client application



Select the client application (EJBApplicationClient) under Java Application from left pane and open the Classpath tab from right side pane. If you don't see your client application, run it once. Select "User Entries" and click on "Add External JARs"

Run Configurations		×
 Run Configurations Create, manage, and run configuration Run a Java application Type filter text Apache Tomcat Tomcat v6.0 Server at localhost Eclipse Application New_configuration Eclipse Data Tools Generic Server Generic Server(External Launch) HTTP Preview J2EE Preview Java Applet New_configuration (1) Java Application ElBApplicationClient New_configuration (4) 	ns Name: EJBApplicationClient ⓒ Main (∞= Arguments () IRE (∞) Classpath () Source () Environment () Common Classpath: ▲ Sootstrap Entries A JRE System Library (JDK) ▲ (∞) User Entries >) HelloWorldSessionBean (default classpath) @ jboss-client.jar - C:\Java\jboss-as-7.1.1.Final\bin\client\	Up Up Down Remove Add Projects Add JARs Add JARs Edit
Ju JUnit Ju Nint Ju New_configuration (2) Ju JUnit Plug-in Test ⊕ OSGi Framework Ju Task Context Test ス XSL		Restore Default Entries

Add the following JAR files.

Run the client

