

University of New York Tirana Faculty of Engineering and Architecture Rruga e Kavajës, pranë 21 Dhjetorit (Sheshi Ataturk) Tirane, Shqipëri

Master of Science in Computer Science

Distributed Systems Manual for Laboratory Practice

Enterprise JavaBeans

PART II

Introducing Servlets and Facelets Connecting Servlets and Facelets with JavaBeans Connecting Servlets to Enterprise JavaBeans Connecting Enterprise JavaBeans to Databases A Web Banking Application with EJB and MySQL

Marenglen Biba, Ph.D Department of Computer Science, E-mail : <u>marenglenbiba@unyt.edu.al</u>

1. Document Purpose

For running the programs, a correct configuration of the running environment is necessary (path and classpath variables).

- Install Java SE (JDK) JDK8 https://www.oracle.com/java/technologies/javase/javase8-archive-downloads.html
- Install Java EE JDK7 <u>http://www.oracle.com/technetwork/java/javaee/downloads/java-ee-sdk-7-downloads-</u> <u>1956236.html</u>
- Install Netbeans 8.2 https://dlc-cdn.sun.com/netbeans/8.2/final/?pagelang=
- Install MySQL 5.0 and MySQL WorkBench 8.0

Set path and Path variables in the operating system

Click on Environment Variables.

Find the Path system variable and click Edit. Set the value of the variable to the directory where you have installed Java, for example:

D:\Program Files\Java\jdk1.8.0\bin

Edit environment variable		×
C:\Program Files\Java\jdk-1.8\bin		New
C:\ProgramData\Oracle\Java\javapath		
C:\Users\Admin\Desktop\ORACLE\bin		<u>E</u> dit
%SystemRoot%\system32		
%SystemRoot%		Browse
%SystemRoot%\System32\Wbem		
%SYSTEMROOT%\System32\WindowsPowerShell\v1.0\		Delete
%SYSTEMROOT%\System32\OpenSSH\		
		Movellp
		Move <u>op</u>
		Maus Daum
		Move D <u>o</u> wn
		Edit <u>t</u> ext
	OK	Cancel

Ensure that the required JDK software is installed on your system and that

the JAVA_HOME environment variable points to the JDK installation directory, not the Java Runtime Environment (JRE) software.

ironment Variables	>
lear variables for Admin	
Variable	Value
MOZ_PLUGIN_PATH	C:\Program Files (x86)\Foxit Software\Foxit PDF Reader\plugi
OneDrive	C:\Users\Admin\OneDrive
Path	C:\Users\Admin\AppData\Local\Programs\Python\Python312
TEMP	C:\Users\Admin\AppData\Local\Temp
TMP	C:\Users\Admin\AppData\Local\Temp
	New Edit Delete
	<u>N</u> ew <u>E</u> dit <u>D</u> elete
	<u>N</u> ew <u>E</u> dit <u>D</u> elete
ystem variables	<u>N</u> ew <u>E</u> dit <u>D</u> elete
ystem variables Variable	<u>N</u> ew <u>E</u> dit <u>D</u> elete
ystem variables Variable ComSpec	New Edit Delete Value ^ C:\Windows\system32\cmd.exe ^
ystem variables Variable ComSpec DriverData	New Edit Delete Value ^ C:\Windows\system32\cmd.exe ^ C:\Windows\System32\Drivers\DriverData ^
ystem variables Variable ComSpec DriverData JAVA_HOME	New Edit Delete Value ^ C:\Windows\system32\cmd.exe ^ C:\Windows\System32\Drivers\DriverData ^ C:\Program Files\Java\jdk-1.8 ^
ystem variables Variable ComSpec DriverData JAVA_HOME NUMBER_OF_PROCESSORS	New Edit Delete Value C:\Windows\system32\cmd.exe C:\Windows\System32\Drivers\DriverData C:\Program Files\Java\jdk-1.8 8
vstem variables Variable ComSpec DriverData JAVA_HOME NUMBER_OF_PROCESSORS OS	New Edit Delete Value C:\Windows\system32\cmd.exe C:\Windows\System32\Drivers\DriverData C:\Program Files\Java\jdk-1.8 8 Windows_NT
vstem variables Variable ComSpec DriverData JAVA_HOME NUMBER_OF_PROCESSORS OS Path	New Edit Delete Value C:\Windows\system32\cmd.exe C:\Windows\System32\Drivers\DriverData C:\Program Files\Java\jdk-1.8 8 Windows_NT C:\Program Files\Java\jdk-1.8\bin;C:\ProgramData\Oracle\Jav
vstem variables Variable ComSpec DriverData JAVA_HOME NUMBER_OF_PROCESSORS OS Path PATHEXT	New Edit Delete Value C:\Windows\system32\cmd.exe C:\Windows\System32\Drivers\DriverData C:\Program Files\Java\jdk-1.8 8 Windows_NT C:\Program Files\Java\jdk-1.8\bin;C:\ProgramData\Oracle\Jav .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
vstem variables Variable ComSpec DriverData JAVA_HOME NUMBER_OF_PROCESSORS OS Path PATHEXT	New Edit Delete Value C:\Windows\system32\cmd.exe C:\Windows\System32\Drivers\DriverData C:\Program Files\Java\jdk-1.8 8 Windows_NT C:\Program Files\Java\jdk-1.8\bin;C:\ProgramData\Oracle\Jav COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
vstem variables Variable ComSpec DriverData JAVA_HOME NUMBER_OF_PROCESSORS OS Path PATHEXT	New Edit Delete Value ^ C:\Windows\system32\cmd.exe ^ C:\Windows\System32\Drivers\DriverData ^ C:\Program Files\Java\jdk-1.8 8 Windows_NT C:\Program Files\Java\jdk-1.8\bin;C:\ProgramData\Oracle\Jav COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC ~ New Edjt Delete
variable ComSpec DriverData JAVA_HOME NUMBER_OF_PROCESSORS OS Path PATHEXT	New Edit Delete Value ^ C:\Windows\system32\cmd.exe ^ C:\Windows\System32\Drivers\DriverData ^ C:\Program Files\Java\jdk-1.8 8 Windows_NT C:\Program Files\Java\jdk-1.8\bin;C:\ProgramData\Oracle\Jav .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC ~ New Edit Delete
variable ComSpec DriverData JAVA_HOME NUMBER_OF_PROCESSORS OS Path PATHEXT	New Edit Delete Value C:\Windows\system32\cmd.exe C:\Windows\System32\Drivers\DriverData C:\Program Files\Java\jdk-1.8 8 Windows_NT C:\Program Files\Java\jdk-1.8\bin;C:\ProgramData\Oracle\Jav .COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC New Edjt Delete

Download and install the Netbeans IDE by double clicking the executable installation file.

Download and install Java EE SDK.

If the .exe file does not start use the following command:



🛓 Java EE 7 SDK		_		×
C MAR I				
Introduction Installation Type	Introduction			7
Install Directory Update Tool				
Ready To Install Progress	Welcome to the Java EE 7 SDK installation.			
Config Results Summary	This installer will guide you through the installation process. You will s learn the latest Java EE 7 features, and you can get started with the F Java EE Tutorials. View sample application source code and then dep Server 4.0 to see them in action. You will find that Java EE 7 is a easy feature-rich platform for developing web and enterprise applications.	hortly be First Cup a loy to Gla -to-learn	able to and assFish	
Java.				
ORACLE [®]	Cancel Back	Ne	xt	-

🛓 Java EE 7 SDK		_		×
C HIND 2				
Introduction Installation Type Install Directory Update Tool Ready To Install Progress Config Results Summary	Installation Type Choose installation type. Typical Installation Installs a GlassFish Server management domain; ideal for development or non business critical use. Please make sure that the ports 4848 and 8080 are free. Custom Installation Not supported.	ıt I		
لان Java				
ORACLE.	Cancel Back	Nex	(t	

Choose directory for Glassfish:

🛓 Java EE 7 SDK		_		×
C HIMP I				
	Install Directory			
Introduction				
Installation Type				
Install Directory				
Update Tool				
Ready To Install				
Progress				
Config Results	GlassFish Server will be installed into the specified directory. You many	ay specify	а	
Summary	different directory or click the Browse to select a directory.			
	Installation Directory C:\glassfish4			
	If the specified installation directory does not exist, the installer will c	reate it for	you.	
Java-				
ORACLE [®]	Cancel Back	Ne	xt	

Click again Next:

🕌 Java EE 6 SDK			
	Update Configuration		
Welcome License Installation Directory Administration Settings Update Configuration JDK Selection Ready To Install Progress Product Registration Summary	If a connection is available, the installer will attempt to download and configure the Update Tool for this installation. Install Update Tool Proxy Host Proxy Port The Update Tool periodically checks for available updates and collects some system data. For more information about the data that is collected, see the <u>GlassFish Usag</u> Metrics page (http://wiki.glassfish.java.net/Wiki.jsp?page=UsageMetricsV3). Enable Update Tool	 e	
Sun Java	Cancel Back Next		



퉬 Java EE 7 SDK				×
C THE I	Progress			
Introduction				
Installation Type				
Install Directory	Java EE 7 SDK			
Update Tool				
Progress	Modular, Lightweight, Open			
Config Results	Modular architecture based on OSGi			
Summary	 Fast startup, less memory consumption Java EE 7 Certified 			
	Developed in Open Source			
Java ⁻	Installing GlassFish Server			
	9%Time Remaining 00:00:28			1
ORACLE				
	Cancel Back	Ne	xt	

🛓 Java EE 7 SDK Х C HHH **Config Results** Introduction The configuration has succeeded. Please see the output below. Installation Type Install Directory ۰ Domain domain1 allows admin login as user "admin" with no password. Login information relevant to admin user name [admin] Update Tool for this domain [domain1] stored at Ready To Install [C:\Users\Admin\.gfclient\pass] successfully. Make sure that this file remains protected. Progress Information stored in this file will be used by Config Results administration commands to manage this domain. Summary Command create-domain executed successfully. Starting domain Executing command :C:\glassfish4\glassfish\bin\asadmin.bat start-domain domain1 C:\glassfish4\glassfish\bin\asadmin.bat start-domain domain1 Attempting to start domain1.... Please look at the server log for more details..... ORACLE Cancel Configure again Next

실 Java EE 7 SDK			_		×
C HIND I					
	Summary				
Introduction					٦
Installation Type					
Install Directory					
Update Tool		Overall Status: Complete			
Ready To Install	Please see the <u>detailed</u>	summary report for an overview of this session,	including <u>next steps</u>	s for using	
Progress	this installation.Please se	ee the <u>log file</u> for detailed information.			
Config Results	2024-04-11-14-47-instal	I-summary html			
Summary	2024-04-11-14-47-instal	llog			
	Product Name	Status			
		Installed			
	Update Tool Bootstrap	Installed			
	GlassFish Server	Installed			
	Uninstallation Software	Installed			
2	Update Tool Bootstrap	Configured			
چن Java	GlassFish Server	Configured			
ORACLE.					_
	Cancel		Back	cit	

Only if "Overall Status" is "Complete", your installation has been performed appropriately.

In order for the examples of this tutorial to execute you need to set the PATH with the directory of Glassfish as follows:

Click on Environment Variables:

Find the Path variable and click Edit.

Edit environment variable		×
C:\Program Files\Java\jdk-1.8\bin		New
C:\glassfish4\glassfish\bin		
C:\ProgramData\Oracle\Java\javapath		<u>E</u> dit
C:\Users\Admin\Desktop\ORACLE\bin		
%SystemRoot%\system32		Browse
%SystemRoot%		_
%SystemRoot%\System32\Wbem		Delete
%SYSTEMROOT%\System32\WindowsPowerShell\v1.	.0\	Denete
%SYSTEMROOT%\System32\OpenSSH\		
		Move <u>U</u> p
		Move D <u>o</u> wn
		Edit <u>t</u> ext
	ОК	Cancel

In the variable value add the path of Glassfish.

Download and Install MySQL Server:

After you download use MySQL Server Instance Config Wizard



Choose the detailed configuration:



Choose developer machine:

MySQL Server Instance Configuration Wizard
MySQL Server Instance Configuration Configure the MySQL Server 5.0 server instance.
Please select a server type. This will influence memory, disk and CPU usage.
 this option for web/application servers. MySQL will have medium memory usage. Dedicated MySQL Server Machine This machine is dedicated to run the MySQL Database Server. No other servers, such as a web or mail server, will be run. MySQL will utilize up to all available memory.
< Back Next > Cancel

Choose multifunctional:



Choose Drive:

MySQL Server Instance Configuration Wizard
MySQL Server Instance Configuration
Press the [Modiry] button to change the InnoDB datafile settings.
InnoDB Tablespace Settings
Please choose the drive and directory where the InnoDB tablespace should be placed.
Drive Info
Volume Name: -
File System: -
Diskspace Used Free Diskspace
Modify < Back Cancel

Choose decision support:

MySQL Server In	stance Configuration Wizard 🛛 🛛 🔀
MySQL Server I Configure the M	nstance Configuration ySQL Server 5.0 server instance.
Please set the a	pproximate number of concurrent connections to the server. Ipport (DSS)/OLAP
&	Select this option for database applications that will not require a high number of concurrent connections. A number of 20 connections will be assumed.
🔿 Online Trai	nsaction Processing (OLTP)
	Choose this option for highly concurrent applications that may have at any one time up to 500 active connections such as heavily loaded web servers.
🔘 Manual Sel	tting
<u> </u>	Please enter the approximate number of concurrent connections.
	,
	< Back Next > Cancel

Perform the following checks:

MySQL Server Instance Configuration Wizard								
MySQL Server Instance Configuration Configure the MySQL Server 5.0 server instance.								
Please set the networking options.								
Enable this to allow TCP/IP connections. When disabled, only local connections through named pipes are allowed. Port Number: 3306 Add firewall exception for this port								
Please set the server SQL mode.								
🔽 Enable Strict Mode								
This option forces the server to behave more like a traditional database server. It is recommended to enable this option.								
< Back Next > Cancel								

Best Support For Multilingualism: Choose this option if you want to use utf8 as the default server character set. This is a Unicode character set that can store characters from many different languages.

MySQL Server Instance Configuration Wizard	
MySQL Server Instance Configuration Configure the MySQL Server 5.0 server instance.	\bigcirc
Please select the default character set. Standard Character Set	
Hello! Makes Latin1 the default charset. This character set is suited for English and other West European languages.	
🔿 Best Support For Multilingualism	
Make UTF8 the default character set. This is the recommended character set for storing text in many different languages.	
O Manual Selected Default Character Set / Collation	
Please specify the character set to use.	
Character Set: atin1	
< Back Next >	Iancel

Set the password for root:

MySQL Server I	Instance Configurati	on Wizard					
MySQL Server Instance Configuration							
Configure che	MySQL Server 5.0 server	instance.					
Please set the	security options.						
🔽 Modify Se	ecurity Settings						
	New root password:		Enter the root password.				
root	Confirm:		Retype the password.				
		🔲 Enable root a	access from remote machines				
🔲 Create An	Anonymous Account						
2	This option will create ar note that this can lead t	n anonymous accour 10 an insecure syster	it on this server. Please n.				
		< Back	Next > Cancel				

Press Execute:

MySQL Server Instance Configuration Wizard	
MySQL Server Instance Configuration Configure the MySQL Server 5.0 server instance.	\bigcirc
Ready to execute	
 Prepare configuration 	
 Write configuration file 	
 Start service 	
 Apply security settings 	
Please press [Execute] to start the configuration.	
< Back]Cancel

Restart the computer and the installation should be complete.

Install the MySQL Workbench.

You can create the database in two ways:

- 1. By commands in the MySQL console
- 2. By graphical user interface in MySQL Workbench

Click on local instance with the right and click Query Database.



Click with the right and select create schema.



Give a name to the database: and press Apply.



Click with the right on the Tables options and select Create Table:



Choose a name for the Table:

new_table	2								
	Name:	account			The name of the table. It is recommended to use only alpha-numeric characters. Spaces should be avoided and be replaced by $_$				
	Collation:	Schema Default		~	The charset/collation specifies which language specific characters can be stored in the table and their sort order. Common choices are Latin1 or UTF8.				
	Engine:	Server Default		~	The database engine that is used for the table. This option affects performance, data consistency and much more.				
	Comments:								
Table	Columns Ind	exes Foreign Keys	Triggers	Partitioning) Options				
DBMS fee	DBMS feedback messages will go here upon applying changes.								

Click on Columns and add the columns for the table. At the end click Apply.

new_table						×
Column Name	Datatype	PK NN UQ	BIN UN ZF AI Default		Column Details	
? jidaccount	INT				Collation:	
					Table Default	~
					Comments:	
						<u>~</u>
l						
Table <mark>Columns</mark> Indexes	Foreign Keys Triggers	Partitioning	Options			
DBMS feedback messages will go	here upon applying change	5.		App	oly Revert	Close

Following the above procedure create three tables:

Table Account Fields: IdAccount (int), Balance (float)

Table Customer Fields: idCustomer(int), Name (Varchar), surname (Varchar)

Table AccountCustomer Fields: idAccount, IdCustomer

In AccountCustomer both fields are primary keays as follows:

accountcustomer	Tark , accord						×
accountcustomer Column Name idAccount idCustomer	Datatype INT(11) INT(11)	PK NN UQ	BIN UN ZF A	I Default	Column Collatio Table D Comme	Details n: lefault nts:	
J							~
Table <mark>Columns</mark> Indexes F	oreign Keys Triggers	Partitioning	Options				
DBMS feedback messages will go her	e upon applying changes.			Å. •	Apply	Revert	Close

2. Running the Web examples

You will be given a folder with the following examples:

- Hello: a web application that shows with a facelet a greeting that is read from a java bean
- Hello2: a web application that reads a string from the user in the browser through a servlet, sends it to another servlet that returns back a response
- GuessNumber: a web application that reads a number through a facelet and compares the number with a random number generated with a java bean. Then it sends back the response to another facelet.

To run the examples go to Open Project as follows:

I N	etBe	ans ID	E 6.8			
File	Edit	View	Navigate	Source	Refactor	Run
¢	New	Project	t	Ctrl+Shif	t+N	
ф.	New	File		Ctrl+N		
	Оре	n Proje	ct	Ctrl+Shif	t+O	
	Ope	n Recer	nt Project		•	
	Оре	n Kenai	Project			
	Clos	e Proje	ct			
	Оре	n File				
	Ope	n Recer	nt File		•	
	Proj	ect Gro	up		•	
	Proj	ect Prop	perties			
	Impo	ort Proj	ect		•	
	Save	е		Ctrl+S		
	Save	e As				
	Save	e All		Ctrl+Shif	t+S	
	Page	e Setup				
	Print			Ctrl+Alt+	-Shift+P	
	Print	to HTN	4L			
	E×it					

Open the Web folder:

🗊 Open Projec	:t		
	Look in: 🛅 v	web	🤣 📂
My Recent Documents	uessr 	number	Project Name: hello
			♥ Open as <u>M</u> ain Project
Desktop			Open <u>R</u> equired Projects:
My Documents			
My Computer			
My Network	File <u>n</u> ame:	cuments and Settings\Geni\Desktop\ADV-OP5Y5 2010-2011\M	Y SLIDES\LESSON 9\web\hello Open Project
Places	Files of <u>type</u> :	Project Folder	

And choose the project that you want to open.

Example 1: Connecting a Bean with a Facelet

The project is composed as follows:

One Bean that prints the String "Hello World" as shown below:



And one Facelet that takes the message of the bean and shows it in the web page.



If we want to run the application, it must be first build, then deployed and finally run. To properly build copy the given directory "bp-project" in the same directory where the "web" folder is.

Then perform in turn: Clean and Build, Deploy and Run. When it is executed the following window will appear:



Hello World!

Example 2: Hello2

Once you open the web application you will see the following:



There are two servelets: one to get the input from the user GreetingServlet and one to send the response to the user ResponseServlet:

Once we build, deploy and run the application we will have the following:

🕙 н	🕹 Hello - Mozilla Firefox										
<u>F</u> ile	<u>E</u> dit	<u>V</u> iew	Hi <u>s</u> tory	<u>B</u> ookmarks	<u>T</u> ools <u>H</u> elp						
<		- C	×	☆ 😵	http://localhost:8080/hello2/greeting						
🔎 М	lost Visi	ted 📄	Getting S	tarted <u> L</u> a	test Headlines						
	Hello +										
ļ	×										

Hello, my name is Duke. What's yours?



2

If we write something the in textfield, the greeting servlet will read the data and send it to the response servlet that will send a response to the client as follows:



Hello, my name is Duke. What's yours?

Peter Pan						
Submit	Reset					

If we press submit the following appears:

🕹 Hello - Mozilla Firefox										
<u>F</u> ile	<u>E</u> dit	<u>V</u> iew	Hi <u>s</u> tory	<u>B</u> ook	marks	<u>T</u> ools	<u>H</u> elp			
<	$\left \right\rangle$	- C	×		e.		http://	/localhost:8080/hello2/greeting?username=F		
<u>)</u> M	🙍 Most Visited 📄 Getting Started 🔝 Latest Headlines									
	Hello						÷			
	4									

Hello, my name is Duke. What's yours?



Hello, Peter Pan!

Example 3: Guess number

Open the project:

🗊 Open Project 🔀		
My Recent Documents	Look in: web uessnumber hello hello2	 ✓ Ø Ø Project Name: guessnumber ✓ Open as Main Project
Desktop My Documents		Open <u>R</u> equired Projects:
My Computer		
My Network Places	File name: and Settings\Geni\Desktop\ADV-OPSYS 2010-2011\M* Files of type: Project Folder	Y SLIDES\LESSON 9\web\guessnumber Open Project Cancel

In the project there is one bean and two facelets: one to read the value and one to display the response from the bean.

🗊 guessnumber - NetBeans IDE 6.8			
<u>File Edit View N</u> avigate <u>S</u> ource Ref <u>a</u> ctor	<u>R</u> un <u>D</u> ebug <u>P</u> rofile Tea <u>m</u> <u>I</u> ools <u>W</u> indow <u>H</u> elp		
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1			
Proj 🔍 × Files Services 💩 UserNumberBean.java 🗴 🎯 greeting.xhtml 🗴			
🖃 🛞 guessnumber	[] ·] ·] ·] ·] ·] ·] ·] ·] ·]		
🖨 🕞 Web Pages	html		
	19 🗄 <h:inputtext< td=""></h:inputtext<>		
	20 id="userNo"		
	<pre>21 value="#(userNumberBean.userNumber)"></pre>		
	22 🖸 <f:validatelongrange< td=""></f:validatelongrange<>		
	23 minimum="#(userNumberBean.minimum)"		
UserNumberBean.java	24 - maximum="#{userNumberBean.maximum}"/>		
	25 -		
🗊 \overline 📷 Configuration Files	26		
🗄 🖓 🙀 Server Resources	<pre>27 <h:commandbutton action="response.xntml" id="submit" value="submit"></h:commandbutton> 28 (h:commandButton id="submit" value="submit" action="response.xntml"/></pre>		
	20 - <n:message showbecall-"laise"<="" showsummary-"true"="" th=""></n:message>		
	30 font-femily: New Century Schoolbook! serif:		
	31 font-style: obligue:		
	32 text-decoration: overline"		
	33 id="errors1"		
	34 - for="userNo"/>		
	35 -		
	36 -		
	37 -		
	38 -		
1			

Once you build, deploy and run the application you will see the following:



Hi, My name is Duke. I am thinking of a number between 0 and 10. Can you guess it ?

Submit

Insert a number:



Press submit:
🕹 Guess Number Facelets Application - Mozilla Firefox
<u>File Edit View History Bookmarks Tools Help</u>
🕜 🕞 🗸 🏠 📽 📋 http://localhost:8080/guessnumber/faces/greeting.xhtml;jsessionid=3010c29121be3242decc1dc8ceb4
🖻 Most Visited 📋 Getting Started 🔊 Latest Headlines
Guess Number Facelets Application

Sorry, 7 is incorrect.

Back

The numbers do not match.

3. Developing a web application with a Servlet and an EJB

In order to create the web Application perform the following steps in NetBeans: Go to new in Netbeans:

🗊 Neti	Bea	ns ID	E 6.8					
File Ec	tit ۱	∕iew	Navigate	Source	Refactor	Run	Debug	Profile
- 💾 N	lew F	Project		Ctrl+Shif	t+N			~
1 N	lew F	ile		Ctrl+N				
23 o	Dpen	Projec	:t	Ctrl+Shif	t+0	vices		Start
	pen	Recer	nt Project		•			
C	pen	Kenai	Project					
C	lose	Proje	t					
C	Dpen	File						
C)pen	Recer	nt File		•			
F	rojec	t Grou	qu		•			
F	rojec	t Prop	perties					
I	mpor	t Proje	ect		•			
S	jave			Ctrl+S				
S	jave (As						
	jave i	All		Ctrl+Shif	t+S			
F	age :	Setup						
P	rint.			Ctrl+Alt+	-Shift+P			
F	Print t	o HTM	1L					
E	Xit							
						1		

Select Java Web as follows:

New Project		
<u>Steps</u>	Choose Project	
 Choose Project 	Categories: Java JavaFX Java Web Java Web Java EE Java ME Maven NetBeans Modules Samples	Projects: Web Application Web Application with Existing Sources Web Free-Form Application
	Description: Creates an empty Web application in IDE-generated build script to build, run	a standard IDE project. A standard project uses an , and debug your project.
	< <u>B</u> ack	Next > Einish Cancel Help

Rew Web Application		
Steps	Name and Location	
 Choose Project Name and Location Server and Settings Frameworks 	Project Name: HelloWorld Project Location: ngs\Geni\Desktop\ADV-OPSYS 2010-2011\MY SLIDES\LESSON 9\Test Project Ender: Desktop\ADV-OPSYS 2010-2011\MY SLIDES\LESSON 9\Test\HelloWorld	rowse
	Use <u>D</u> edicated Folder for Storing Libraries	rowse
	Different users and projects can share the same compilation libraries (see Help for details).	
	< <u>B</u> ack Next > Einish Cancel	Help

New Web Application			×
Steps	Server and Set	ttings	
 Choose Project Name and Location 	<u>A</u> dd to Enterprise	Application:	
 Server and Settings Frameworks 	<u>S</u> erver:	GlassFish v3 Domain 🗸 🖌 Add]
		Use dedicated library folder for server JAR files	
	Java EE Version:	Java EE 6 Web 💌	
	Context <u>P</u> ath:	/HelloWorld	
			_
		< <u>Back</u> Next > Einish Cancel Help	J

New Web Application	
Steps 1. Choose Project 2. Name and Location 3. Server and Settings 4. Frameworks	Frameworks Select the frameworks you want to use in your web application. Spring Web MVC 2.5 JavaServer Faces Struts 1.3.8 Hibernate 3.2.5
	< <u>B</u> ack Next > <u>Finish</u> Cancel <u>H</u> elp

The initial content is this one:



If we run the application:

🗊 HelloWa	orld - NetBeans IDE 6.8	
File Edit V	/iew Navigate Source Refactor Ru	un Debug Profile
ት 🔁	💾 🖣 🤊 (°) 🗖	<u>ک</u>
Proj 🕫	× Files Services	🗊 index.jsp 🗙 📃
⊫ ⊕ Hell ⊨ ि	New	🕼 🗟 • 🗟 • 🛛
⊕(Build	1 🖓 <%
	Clean and Build	2 Doc
T L	Clean	3 Cre
	Generate Javadoc	4 Aut
÷	Run	5
÷	Deploy	6
÷	Debug	7 <%@page
	Profile	8 DUCIY</th
	Test RESTful Web Services	10
	Test Alt+F6	11 🖵 <html></html>
	Set as Main Project	12 📮 🛛 <he< th=""></he<>
	Open Required Projects	13
	Close	14
		_15 -
	Rename	16 🗐 <bo< th=""></bo<>
	Move	17
	Copy	18 -
	Delete	$19 \stackrel{\text{L}}{\sim} $
	Find	20
	Share on Kenai	
	Versioning 🕨	lutput
	Local History	👌 Java DB Datab

You will see the following:

🕹 JSP Page - Mozilla Firefox										
<u>F</u> ile	<u>E</u> dit	<u>⊻</u> iew	Hi <u>s</u> tory	<u>B</u> ookr	marks	<u>T</u> ools	Help			
)>	- C	×		¥		http://localhost:8080/HelloWorld/			
🔎 М	ost Visit	ted 📄	Getting S	itarted	🔊 La	test Head	dlines			
	JSP Pa	ige					÷			

Hello World!

If we change the text as follows:



Press Refresh in the browser as follows:

sr 🌏	🕹 JSP Page - Mozilla Firefox										
Eile	<u>E</u> dit	<u>V</u> iew	Hi <u>s</u> tory	<u>B</u> ook	marks	<u>T</u> ools	Help				
)>	- C	×		¥		http://localhost:8080/HelloWorld/				
🔎 М	🙍 Most Visited 📄 Getting Started 🔝 Latest Headlines										
	JSP Pa	ige					+				

Hello World!

:

After refreshing the web page you will get:

🕹 JSP	🕹 JSP Page - Mozilla Firefox									
<u>Eile E</u> o	dit <u>V</u> iew	Hi <u>s</u> tory	<u>B</u> ookmarks	Tools Help						
$(\mathbf{\cdot})$	>)•	2 ×	☆ જ	http://localhost:8080/HelloWorld/						
🙍 Most Visited 📄 Getting Started 🔊 Latest Headlines										
📄 JSF	Page			*						

Hello World Again! Do Not Worry Be Happy

Let us now add a servlet to the project:



Rew Servlet		
Steps	Name and L	ocation
 Choose File Type Name and Location Configure Servlet Deployment 	Class <u>N</u> ame:	TestServlet
	<u>P</u> roject:	HelloWorld
	Location:	Source Packages
	Pac <u>k</u> age:	server
	<u>C</u> reated File:	PSYS 2010-2011\MY SLIDES\LESSON 9\Test\HelloWorld\src\java\server\TestServlet.java
		< <u>B</u> ack Next > Einish Cancel Help

New Servlet									
Steps	Configure Servlet Deployment								
 Choose File Type Name and Location Configure Servlet Deployment 	Register the Servlet with the application by giving the Servlet an internal name (Servlet Name). Then specify patterns that identify the URLs that invoke the Servlet. Separate multiple patterns with commas.								
	Class Name: server.TestServlet								
	Servlet Name: TestServlet								
	URL Pattern(s): /TestServlet								
	Initialization Parameters:								
	Name	Value	New						
			<u>E</u> dit						
			Delete						
		< Back Next > Finish	Cancel <u>H</u> elp						

The following is created:



You will see in the Servlet a TODO part generated for you automatically:

j in	dex.jsp 🗙 🖄 TestServlet.java 🗴
(- - - -
28	* @throws IOException if an I/O error occurs
29	L */
30	<pre>protected void processRequest(HttpServletRequest request, HttpServletRespons</pre>
31	throws ServletException, IOException {
32	response.setContentType("text/html;charset=UTF-8");
33	<pre>PrintWriter out = response.getWriter();</pre>
34	try (
35	/* TODO output your page here
36	<pre>out.println("<html>");</html></pre>
37	<pre>out.println("<head>");</head></pre>
38	out.println(" <title>Servlet TestServlet</title> ");
39	<pre>out.println("");</pre>
40	out.println(" <body>");</body>
41	<pre>out.println("<h1>Servlet TestServlet at " + request.getContextPath (</h1></pre>
42	<pre>out.println("");</pre>
43	<pre>out.println("");</pre>
44	*/
45	} finally {
46	<pre>out.close();</pre>
47	}
48	L)
49	

After building and deploying, if you run the application by changing the browser address you will see a blank page:



If you uncomment the TODO part you will get:



Servlet TestServlet at /HelloWorld

If you want to set deploy on save you can go to Properties:



Then Run:

Project Properties - HelloWorld			×
Categories:			
Sources	Server:	GlassFish v3 Domain	*
Frameworks	Java EE Version:	Java EE 6 Web	
Elbraries	Context Path:	/HelloWorld	
Compiling	✓ Display Brows	er on Run	
Occumenting	Specify the URL	relative to the context path to run:	
Run Enrmatting	Relative URL:		
- Formacing		(e.g. /admin/login.jsp)	
	Deploy on Save If selected, files are compiled and deployed when you save them. This option saves you time when you run or debug your application in the IDE		
	· · · · · · · · · · · · · · · · · · ·		
	<u>V</u> M Options:		
		(used for running main classes or unit tests; e.gXms10m)	
		OK Cancel <u>H</u> elp	

And check Deploy on Save. From now on every time you save the project it is automatically deployed in the server.

Adding an EJB to the Web Application

Go to New as follows and choose Session Bean:



New Session Bean			\mathbf{X}
Steps	Name an	d Location	
 Choose File Type Name and Location 	EJB <u>N</u> ame:	TestBean	
	<u>P</u> roject:	HelloWorld	
	Location:	Source Packages	~
	Pac <u>k</u> age:	server	~
	Session Ty Sta Sta Sin Create Int Re Loc	pe: teless teful gleton erface: note al	
		< <u>B</u> ack Next > <u>Finish</u> Cancel <u>H</u> elp	

You will see the following:



Now lets add some code to the EJB that we created:

Click with the right of the mouse within the class and choose Insert Code:



Choose Add Business Method:



The following window will appear:

🗊 Add Busi	ness Method					
<u>N</u> ame:	businessMethod					
Return <u>T</u> ype:	void					Browse
Parameters	Exceptions					
Name		Туре		Final		Add
						Remove
						Up
						Down
Use in Interfa	ce: 💿 <u>L</u> ocal	○ R <u>e</u> mote	⊖ <u>B</u> oth			
					ОК	Cancel

Write the following in the window:

🗊 Add Busi	ness Method				
<u>N</u> ame:	SendHello				
Return <u>T</u> ype:	String				Browse
Parameters	Exceptions				
Name		Түре	Final		Add
name		java.lang.String			- Remove
		02			<u>Femove</u>
Use in Interfa	ce: 💿 Local	○ R <u>e</u> mote ○ <u>B</u> oth			
				ОК	Cancel

You will see the following code generated:

	7
Packages	
ver	8 import javax.ejp.stateless;
TestBean.java	
TestServlet.java	
ickages	
<	12 * @author Geni
varios	13 4 */
ina Danaa	14 @Stateless
ise beans	15 public class TestBean {
Iration Files	16
	17 public String SendHello(String name) (
	18 return null;
	19
	20
	21 // Add business logic below. (Right-click in editor and choose
	22 // "Insert Code > Add Business Method")
	23
	24
	25
	26 }

```
13 ×/
14
     @Stateless
15
     public class TestBean {
16
17 🖃
         public String SendHello(String name) {
             return "Name: " + name;
18
19
         }
20
21
         // Add business logic below. (Right-click in editor and choose
22
         // "Insert Code > Add Business Method")
23
24
25
```

In the class of the servlet add the following code:

@EJB TestBean bean;



In the code below add:

out.println("<h1>" + bean.SendHello("The Boss") + "</h1>");

· · · ·					· · · · ·
Services	index.j:	sp 🗴 🙆 TestServlet.java 🛛 🗴 🙆 TestBean.java	x		
	🔛 🐼 ·	· 3 · Q 7 7 8 7 8 7 8 9 9	••••••••••••••••••••••••••••••••••••••		
	29	* @ param response servlet resp	server.TestBean		
	30	* @throws ServletException if :			
es	31	* @throws IOException if an I/	<pre>public String SendHello()</pre>	<u>Strinq</u> name)	
	32 -	*/			
ean.java	33	protected void processRequest (H	Javadoc not found. Either Javadoc	documentation for this item does not exist	or you
ervlet.java	34 -	throws ServletException, IOException,	nave not added specified Javadoc Managor	in the Java Platform Manager or the Library	(
	35	response.setContentType("te:	Mahayer.		
	36	PrintWriter out = response.			
	37	try (
ns	38	// TODO output your page			
Files	39	out.println(" <html>");</html>			
	40	out.printin(" <nead>");</nead>			
	41	out.printin(" <title>Ser</title>			
	42	out.printin("");			
	43	out.printin(" <body>");</body>			
	44	out.printin(" <ni>servie)</ni>			
	40	out println("<" + bea	an. [+ ~~/ ni>~);	Character and	
	40	out println(");	SendHello(String name)	String	
	47	ouc.princin();	Ustring()	String	
	40) finelly (equals(object obj)	Close (2)	
	50	out close():	<pre>getClass() </pre>	class(/>	
	51	3	<pre>nashcode()</pre>	Inc	
	52	}	<pre>onciry() oncirg()</pre>	void	
	53	,	<pre>o motifyAff()</pre>	void	
	E A D	The Court of mathematic Clinks on a	wait(long timeout)	void	
gator 🖶 🛪	Output		<pre>wait(long timeout int</pre>	nanos) woid	
~	0.0		ware(rong cimeouc, inc	nanos) voru	

When you run the application (In the browser you should add: <u>http://localhost:8080/HelloWorld/TestServlet</u>.) or if you just refresh the browser you will get:



Servlet TestServlet at /HelloWorld

Name: The Boss

We have completed the application and shown how a servlet can be connected with an Enterprise Java Bean.

4. Developing a web application with a Servlet and an EJB that connects to a database.

Following the schema below, now we will try to build an application where the EJB is connected to a database.



Take the project that we developed in Section 3 and perform the following operations.

First of all change the properties of the Java DB as follows:

NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window I



Open Properties:

🗊 Java DB Prope	Java DB Properties					
Specify the folder where Java DB is installed and the folder where you will keep your databases. The database location folder will be used as the value of the derby.system.home property.						
Java DB Installation:	C:\Program Files\glassfish-4.1.1\javadb	B <u>r</u> owse				
Database Location: Users\Admin\AppData\Roaming\NetBeans\Derby Brows						
	ОК	Cancel				

Change the above fields as follows:

Java DB Prope	rties	×				
Specify the folder where Java DB is installed and the folder where you will keep your databases. The database location folder will be used as the value of the derby.system.home property.						
Java DB Installation:	C:\glassfish4\javadb	Browse				
Database <u>L</u> ocation:	C:\Users\Admin\AppData\Roaming\NetBeans\8.2	Browse				
	ОК	Cancel				

Then create a sample DB:

NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window H



Name the database as "sample" as follows:

🗊 Create Java D	×	
Database <u>N</u> ame:	sample	
Database Location: C:\Users\Admin\AppData\Roaming\NetBeans\8.2		Proper <u>t</u> ies
		OK Cancel
		Calicer

Then connect the database as follows:

NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window



You can now open the database as follows:

Projects	Files	Services ×		-
🖃 🗐 Dat	abases			^
📄 · 🗊	Java DB			
	🗐 sample	e		
🕀 - 🔁	Drivers			
	jdbc:derb	y://localhost:	1527/sample [app on APP]	
	🚽 🔲 Tal	bles		
	.	CUSTOMER		
	.	DISCOUNT_C	CODE	
	🕂 ·· 🗾	MANUFACTU	RER	
	🔁 · 📰	MICRO_MAR	KET	
	⊡	PRODUCT		
	🔁 🗮	PRODUCT_C	DDE	
	<u>ن</u>	PURCHASE_C	ORDER	
6	🗄 🚞 Vie	ws		
6	🗄 🚞 Pro	cedures		
	👌 Other	schemas		
🛨 - 💦	jdbc:myso	l://localhost:	3306/bank?zeroDateTimeBehavior=convertToNull [root on Defau	,
÷ 💦	jdbc:myso	l://localhost:	3306/mysql?zeroDateTimeBehavior=convertToNull [root on Defa	1
🗄 🚇 We	b Services	5		
🗄 🛅 Ser	vers			
🗄 📶 Ma	en Repos	itories		
<u> </u>	•			V

Then create an Entity EJB as follows:



In the window below, choose JDBC/Sample:

:

New Entity Classes from Data	New Entity Classes from Database				
Steps	Database Tables				
 Choose File Type Database Tables Entity Classes Mapping Options 	 Data Source: Database Schema 	jdbc/sample <no database="" in="" project="" schemas="" the=""></no>			
	Available <u>T</u> ables:	S <u>e</u> lected Tables:			
	CUSTOMER DISCOUNT_CODE MANUFACTURER MICRO_MARKET PRODUCT PRODUCT_CODE PURCHASE_ORDER	Add > < Remove << Remove All			
		✓ Include Related Tables			
	Select at least one	table.			
		< <u>Back</u> Next > Einish Cancel Help			

From the table select one table, for example Customer:

🗊 New Entity Classes from Database						
Steps	Database Tables					
 Choose File Type Database Tables Entity Classes Mapping Options 	💿 Data Source:	jdbc/sample				
	O Database <u>S</u> chema Available <u>T</u> ables:	<no database="" in="" project="" schemas="" the=""></no>				
		S <u>e</u> lected Tables:				
	MANUFACTURER MICRO_MARKET PRODUCT PRODUCT_CODE PURCHASE_ORDER	Add > Add All >> << Remove All				
		Include Related Tables				
		< Back Next > Einish Cancel Help	2			

New Entity Classes from Databas	2				3	
Steps	Entity Classes					
1. Choose File Type 2. Database Tables	Specify the names and the location of the entity classes.					
3. Entity Classes 4. Mapping Options	<u>C</u> lass Names:	Database Table	Class Name	Generation Type		
		CUSTOMER	Customer	New		
		DISCOUNT_CODE	DiscountCode	New		
		MICRO_MARKET	MicroMarket	New		
	Project:	Test				
	Location:	Source Packages		•	-	
	Pad <u>k</u> age: server					
	Concerne Named Query Annabeliana for Desciptort Fields					
	Generate Named Query Annotations for Persistent Helds					
	Generate JAXB Annotations					
	V Create Pers	iistence <u>U</u> nit				
			< Back Next >	Einish Cancel Help		

Click "Next".
New Entity Classes from Database							
Steps	Mapping Options						
 Choose File Type Database Tables Entity Classes Mapping Options 	Specify the default mapping options. Association Eetch: default Collection Type: java.util.Collection						
	Hully Qualified Database Tables						
	Use Column Names in Relationships						
	Use Defaults if Possible						
	Generate Fields for Unresolved Relationships						
	< <u>Back</u> Next > Einish Cancel Help						

Click "Finish".

File Edit View Navigate Source Refactor Run Deiter Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr Image: Services Image: Services Image: Services Image: Services Image: Services Image: Pr	HelloWorld - NetBeans IDE 6.8	
Pr Files Files Services HelloWorld Web Pages WEB-INF index.jsp index.jsp index.jsp Source Packages DiscountCode.java DiscountCode.java TestBean.java TestPackages Test Packages Libraries Libraries Enterprise Beans Configuration Files	<u>File E</u> dit <u>V</u> iew <u>N</u> avigate <u>S</u> ource Ref <u>a</u> ct	or <u>R</u> un <u>D</u> e
Pr Files Files Services HelloWorld Web Pages WEB-INF index.jsp index.jsp index.jsp Source Packages Source	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
HelloWorld Web Pages HelloWorld	Pr I × Files Services	index.j:
Web Pages WEB-INF Index.jsp <	🖃 🌐 🛞 HelloWorld	
WEB-INF 30 index.jsp 31 Source Packages 32 Output 33 Output 33 Output 34 Output 35 Output 35 Output 35 Output 36 TestServlet.java 36 TestPackages 38 Output 39 Output 40 Output 41 Output 42 43 44	🖨 🖓 Web Pages	29
Index.jsp 31 Source Packages 32 Image: Server 33 Image: Server 34 Image: Server 35 Image: Server 36 Image: Server 36 Image: Server 36 Image: Server 36 Image: Server 38 Image: Server 38 Image: Server 39 Image: Server 40 Image: Server 42 Image: Server 42	🗊 🛅 WEB-INF	30
Source Packages 32 Server 33 ObscountCode.java 34 DiscountCode.java 35 TestBean.java 36 TestServlet.java 37 Test Packages 38 Test Libraries 39 Test Libraries 39 Test Configuration Files 41 Configuration Files 42	index.jsp	31
Image: Server 33 Image: Customer, java 34 Image: DiscountCode, java 35 Image: DiscountCode, java 35 Image: DiscountCode, java 36 Image: DiscountCode, java 37 Image: DiscountCode, java 38 Image: DiscountCode, java 39 Im	Source Packages	32
Customer.java 34 ⊂ DiscountCode.java 35 35 36 36 37 38 37 38 38 39 39 39 39 39 39 39 39 39 39	erver	33
DiscountCode.java 35 TestBean.java 36 37 36 37 37 38 37 38 39 €@ Test Packages 38 39 €@ Test Libraries 40 €@ Enterprise Beans 41 €@ Configuration Files 42 43	🔤 Customer,java	34 -
IestBean.java 36 TestServlet.java 37 Image: Test Packages 38 Image: Libraries 39 Image: Test Libraries 39 Image: Test Libraries 40 Image: Test Libraries 41 Image: Test Configuration Files 42 43 44	DiscountCode.java	35
Image: Section Viet. Java 37 Image: Section Viet. Java 37 Image: Section Viet. Java 38 Image: Section Viet. Java 39 Image: Section Viet. Java 39 Image: Section Viet. Java 40 Image: Section Viet. Java 40 Image: Section Viet. Java 41 Image: Section Viet. Java 42 Image: Section Viet. Java 43 Image: Section Viet. Java 44	TestBean.java	36
Itest Packages 38 Itest Packages 39 Itest Libraries 40 Itest Libraries 40 Itest Configuration Files 41 Itest Configuration Files 42 43 44	Test Deskierviet. java	37
Image Libraries 39 Image Libraries 40 Image Libraries 41 Image Libraries 42 Image Libraries 42 Image Libraries 43 Image Libraries 44		38
the files interprise Beans 40 the files 41 the files 42 43 44		39
Configuration Files		40
42 43 44		41
43	Configuration nies	42
44		43
		44

In the servlet add the following code:



Add imports:



Now in the code we should call the EJB. Perform the following code changes in the servlet:

	world - NetDeans IDE 0.8		
File Edit	: view Navigate Source Refacti	or Run Debug Profile Team Tools Window:	и нер
° 1		🔄 🝸 🔞 🕨 🔝	
Pr	0 × Files Services	🗊 index.jsp 🗴 🐼 TestServlet.java 🔹 🖉	TestBe javax.persistence.EntityManagerFactory
⊟∰ H	TelloWorld		
<u>.</u>	👌 Web Pages		public EntityManager createEntityManager()
G	∎	37 protected void proce	ssReq
	index.jsp	38 - throws ServletExcept	ion, Create a new application-managed EntityManager. This method returns a new
	🗃 Source Packages	39 response.setCont	ent TyjEntityManager instance each time it is invoked. The isupen method will return the
Ē	server	40 Printwriter out	= resjon die returnet instance.
	🚳 Customer.java		
		42 // 1000 outp	uc yo
		43 Out.printin(Keturns:
	🔤 TestServlet.java	44 Out.printin("cheat Throws:
⊡ …[🗃 Test Packages	45 Out.println("(he. IllegalStateException - if the entity manager factory has been closed
⊡ …[Libraries	47 out println(" / IE " / IE / IE //
÷6	🗃 Test Libraries	48 out println("ch1x
÷6	👌 Enterprise Beans	49 out.println(" <hi></hi>
<u>ا</u> …ا	Configuration Files	50 Customer cus	t = emf.
		51 out.println("
		52 out.println("
		53	createEntityManager(Man man) EntityManager
		54) finally (equals(Object obj) boolean
		55 out.close();	getCache () Cache
		56 }	getClass() Class
		57 - }	<pre>getCriteriaBuilder() CriteriaBuilder</pre>
		58	<pre> getMetamodel() Metamodel </pre>
		59 + HttpServlet methods	Clic 🔵 getPersistenceUnitUtil() PersistenceUnitUtil
		94	<pre>getProperties() Map<string, object=""></string,></pre>
	•		
	40 PrintWrite	<pre>er out = response.getWriter();</pre>	
	41 try {	· · · · · · · · · · · · · · · · · · ·	
а 👘	42 // 100	contin ("chtml>").	Parameters:
	4J Out.pr	<pre>cintln("<head>");</head></pre>	Returns:
	45 out.pr	cintln(" <title>Servlet TestServle'</title>	the new query instance
	46 out.pr	cintln("");	Throws:
	47 out.pr	<pre>sintln("<body>");</body></pre>	java.lang.IllegalArgumentException - if a query has not been defined
	48 out.pr	<pre>sintln("<h1>Servlet TestServlet at</h1></pre>	with the given name or if the query string is found to be invalid
	49 out.pr	<pre>sintln("<h1>" + bean.SendHello("T)</h1></pre>	11
	50 Custom	ner <mark>cust</mark> = emf.createEntityManage	: () .
	51 out.pr	<pre>:intln("");</pre>	😑 clear () void 🛆
	52 out.pr	<pre>:intln("");</pre>	○ close() void
	53	1	Contains (Object entity) boolean
	55 out al	lose().	Query
	56 3		<pre>oreateNativeOvery(String salString)</pre>
	57 }		createNativeQuery(String sqlString, Class result(less)
	58		createNativeQuery(String sqlString, String resultSetMann Ouerv
	59 ± HttpServlet me	ethods. Click on the + sign on th	G createQuery (CriteriaQuery <t> criteriaQuery) TypedOuerv<t></t></t>
	94		createQuery (String qlString) Query
	95 }		<pre>createQuery(String qlString, Class<t> resultClass) TypedQuery<t></t></t></pre>
			🖕 😑 detach (Object entity) void

From Customer take a query for example "Customer.findAll":

 index.)sp index.)sp Source Packages server SiscountCode.java DiscountCode.java TestBean.java TestServlet.java Libraries index. Test Libraries index. Enterprise Beans Configuration Files 	15 16 17 18 19 = 20 21 22 23 24 25 26 27 28 29 30 21	<pre>import javax.persistence.NamedQueries; import javax.persistence.NamedQuery; import javax.persistence.Table; /**</pre>	mer c WHEF HERE c.zi WHERE c.ne tomer c W
	30	<pre>@NamedQuery(name = "Customer.findByAddressline1", query = "SELECT c FROM Cus</pre>	tomer c WF
	31	<pre>@NamedQuery(name = "Customer.findByAddressline2", query = "SELECT c FROM Cus</pre>	tomer c WF
	32	<pre>@NamedQuery(name = "Customer.findByCity", query = "SELECT c FROM Customer c</pre>	WHERE c.ci

Choose getResultList().

	I¢	- 🛃 - 🖣	I • Q 🖓 🖓 层) 🔗 🈓 😫 🖞	2 🔘 🛙			nuhli	c List metResultList()		
IE .	37 protected void processRequest (HttpServletRequest request,							paori	o <u>hibo</u> genebuletie ()		
	38 - throws ServletException, IOException {						Execut	e a SELECT query and return the query results	; as an untyped List.		
sp	39		response.	setContentType('text/h	tml;charset=	=UTF-8");				
kages	40		PrintWrit	er out = respon	se.getW	riter();					
chomor iouo	41		try (F	Retur	ns:		
scomert.java	42		// TO	DO output your	page her	re			a list of the results		
tRess isus	43		out.p	cintln(" <html>"</html>	;		1	Throw	's:		
stDean, java	44		out.p	cintln(" <head>"</head>	;				IllegalStateException - if called for a Java	Persistence query languag	e
	45		out.p	rintln(" <title></title>	Servlet	TestServlet	t")		UPDATE or DELETE statement		
lez	46		out.p	cintln("	');				QueryTimeoutException - if the query execu	tion exceeds the query	
~	47		out.p	cintln(" <body>"</body>	;				timeout value set and only the statement is ro	lled back	
Boong	48		out.p	cintln(" <h1>Ser</h1>	/let Tes	stServlet at	t " + reque		iransactionRequiredException - II a IULK f	noue nas been set anu men	9
op Filos	49		out.p	rintln(" <h1>" +</h1>	bean.Se	endHello("Th	he Boss") +		Is no dansaction	advina faila and the	V
UTFILES	50		Custo	mer cust = emf.	reateE	ntityManager	r().createNa	medQ	uery("Customer.findAll").		
	51		out.p	cintln("	');			Γ	🔵 executeUpdate ()	i	.nt 🔼
	52		out.p	cintln("	');				🔵 getClass()	Class<	.2>
	53								🔵 getFirstResult ()	i	nt
	54		} finally	(🔵 getFlushMode ()	FlushModeTy	pe
	55		out.c	lose();					🥥 getHints()	Map <string, objec<="" th=""><th>:t> 🗏</th></string,>	:t> 🗏
	56		}						🥥 getLockMode ()	LockModeTy	pe
	57	L	}						🥥 getMaxResults ()	i	.nt
	58								🔵 getParameter(String name)	Parameter<	.?>
	59	+	HttpServlet n	ethods. Click c	n the +	⊦ sign on th	e left to ea	dit	getParameter(int position)	Parameter<	.2>
	94								🔵 getParameter (String name, Class	<t> type) Parameter<</t>	.T>
	95	}							getParameter(int position, Clas	s <t> t Parameter<</t>	.T>
avigator 🛛 🖶 🗙	Outr	out					= x itad	ke	getParameterValue(Parameter <t></t>	param)	Т
~	N C	pac			.			K3	🔵 getParameterValue(String name)	Obje	.ct
tring 🔨	W	Java DB	Database Process	GlassFish v3 Dom	ain × H	ielloworia (run) ×	MySQL Server Co	ommar	getParameterValue(int position)	Obje	ct
:pServletRequest re 🔽		run-dis	play-browser:						🥥 getParameters()	Set <parameter<?< th=""><th>>></th></parameter<?<>	>>
>	<u></u>	run:							🧉 getkesultList ()	Li	st

And then the get the first element in the results:



The full line code is:

```
out.println("<h1>Servlet TestServlet at " + request.getContextPath () + "</h1>");
out.println("<h1>" + bean.SendHello("The Boss") + "</h1>");
Customer cust = (Customer)emf.createEntityManager().createNamedQuery("Customer.findAll").getResultList().get(0);
out.println("</body>");
out.println("</html>");
```

Now we need to print the results in the web page:



The line of printing is:

out.println("<h1>Servlet TestServlet at " + request. out.println("<h1>" + bean.SendHello("The Boss") + " Customer cust = (Customer)emf.createEntityManager(). out.println("<h2>" + cust.getName() + "</h2>"); out.println("</body>"); out.println("</html>");

Now if you run the application:

🕙 Se	ervlet	TestS	ervlet -	Mozilla Fire	fox		
Eile	<u>E</u> dit	<u>V</u> iew	Hi <u>s</u> tory	<u>B</u> ookmarks	<u>T</u> ools <u>H</u> elp		
<	>	- C	×	☆ 😵	http://localhost:8080/HelloWorld/TestServlet		
🧖 M	🔊 Most Visited 📄 Getting Started 🔊 Latest Headlines						
	Servle	t Test	5ervlet		+		

Servlet TestServlet at /HelloWorld

Name: The Boss

JumboCom

If you want to change the code in order to get the city of the customer:

```
out.println("<body>");
out.println("<h1>Servlet TestServlet at " + request.get
out.println("<h1>" + bean.SendHello("The Boss") + "</}
Customer cust = (Customer)emf.createEntityManager().cre
out.println("<h2>" + cust.getName() + "</h2>");
out.println("<h2>" + cust.getCity() + "</h2>");
out.println("</body>");
out.println("</html>");
```

When you refresh the browser:

🙂 Se	ervlet	TestSe	ervlet -	Mozilla Fire	efox			
Eile	<u>E</u> dit	<u>V</u> iew	Hi <u>s</u> tory	<u>B</u> ookmarks	<u>T</u> ools <u>H</u> elp			
<		- C	×	☆ જ	http://localhost:8080/HelloWorld/TestServlet			
🔎 М	🚈 Most Visited 📄 Getting Started 🔝 Latest Headlines							
	Servle	t Tests	Servlet		*			

Servlet TestServlet at /HelloWorld

Name: The Boss

JumboCom

Fort Lauderdale

Some more changes:

```
but.println( \sbody> ),
out.println("<h1>Servlet TestServlet at " + request.getContextPath () + "</h1>"
out.println("<h1>" + bean.SendHello("The Boss") + "</h1>");
Customer cust = (Customer)emf.createEntityManager().createNamedQuery("Customer.
out.println("<h2> The name of the customer is: " + cust.getName() + "</h2>");
out.println("<h2> The city of the customer is: " + cust.getCity() + "</h2>");
out.println("</body>");
out.println("</html>");
```

When you refresh the browser:

🕹 Se	ervlet	TestSe	ervlet -	Mozilla Fire	fox		
<u>F</u> ile	<u>E</u> dit	⊻iew	Hi <u>s</u> tory	<u>B</u> ookmarks	<u>T</u> ools <u>H</u> elp		
<	$\left \right\rangle$	- C	×	☆ જ	http://localhost:8080/HelloWorld/TestServlet		
🔎 М	🧖 Most Visited 📄 Getting Started 🔊 Latest Headlines						
	Servle	t TestS	iervlet		+		

Servlet TestServlet at /HelloWorld

Name: The Boss

The name of the customer is: JumboCom

The city of the customer is: Fort Lauderdale

The database to which we are connected can be seen at:

🗊 NetBeans IDE	6.8			
<u>File E</u> dit <u>V</u> iew <u>N</u>	avigate <u>S</u> ource R	Ref <u>a</u> ctor <u>R</u> un	<u>D</u> ebug	Profile
1 1	496			~
Projects	Files	Servic	es	4 0 ×
Databases MySQL S Java DB Java DB Jorivers jdbc:der Jdc:der Servers Servers Hudson Build Kenai Instan Issue Tracke	Server at localhost:3 /by://localhost:1527/ sql://localhost:3306/ ss ders icces ers	306 [root] (disc /sample [app on /bank [root on D	onnected) APP] efault sch) nema]



🗊 NetBeans IDE 6.8	
File Edit View Navigate Source	: Refactor Run Debug Profile
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
Projects Files	Services 🐠 🛪
MySQL Server at localhos Java DB Drivers Jidbc:derby://localhost:15	st:3306 [root] (disconnected)
CUSTOMER CUSTOMER DISCOUNT MANUFACT MICRO_MA COMPACT PRODUCT DUDCL	View Data Execute Command Add Column Refresh Delete
	Grab Structure Recreate Table
 Image: SQLJ Image: SQLJ Image: SYS Image: SYSCAT Image: SYSCS_DIAG Image: SYSCS_UTIL 	Properties

You will see the following editor of SQL and the data of the table:

🗊 NetBeans IDE 6.8	NetBeans IDE 6.8							
<u>File Edit View Navigate Source Refactor Run Debug Profi</u>	ile Tea <u>m T</u>	ools <u>W</u> indow <u>H</u> elp						
- 🔁 🚰 🔩 🔊 🤁 🛄 🗵	T 👸	D 📅 - 🕞 -			Q • 5			
Projects Files Services @ ×	🛛 🖄 TestSe	rvlet.java 🗙 🙆 TestBean.java 🗙 🙆	Customer.java 🗙	🕻 📝 index.jsp 🗙 📑 SQL Command 1 🗙				
Databases	Connection	jdbc:derby://localhost:1527/sample [ap	p on APP]	I I I I I I I I I I I I I I I I I I I	- 🛛 - 🔍 🔜 🖓 📇			
MySQL Server at localhost:3306 [root] (disconnected)	1 Bel	ect * from APP.CUSTOMER						
Java DB								
Jubc:derby://localhost:152//sample [app on APP]								
PRODUCT_CODE								
DURCHASE_ORDER								
🗈 💼 Views								
Procedures								
I NULLID	select * fro	om APP.CUSTOME ×						
B = B SQL3		🖩 🖩 🗶 I 😂 K K K 🖌 🖌 I	Page Size: 20	Total Rows: 13 Page: 1 of 1				
	#		710	NAME	ADDRESSI INE1			
	# 1		33015	TumboCom	111 E. Las Olas Blvd			
	2	2 M	33055	Livermore Enterprises	9754 Main Street			
SYSFUN	3	25 M	75200	Oak Computers	8989 Qume Drive			
SYSIBM	4	3 L	12347	Nano Apple	8585 Murray Drive			
B	5	36 H	94401	HostProCom	65653 El Camino			

5. Developing a web banking application

We will develop a web banking application that connects to the MySQL database.

Perform the following in order:

In order to connect Java with MySQL we need the following connector:

mysql-connector-java-5.1.15-bin.jar

Add the connector to the libraries of the project in Netbeans as follows:

WebBankingApp	- NetBeans IDE 7.1		of second dis-second state (second distance)	
File Edit View Nav	vigate Source Refactor Run	in Debug	Profile Team Tools Window Help	Q Search (Ctrl+I)
1 🔁 🕋 🖣	1 9 C		- T 🐉 🕨 🚯 - 🕦 -	
Projects 8 Files	Services 🔳 🙆	🔬 Account.	ava 🕫	
HelloWorld	S	Source H	story 🕼 🖩 - 🗐 - 🔽 🖓 🖓 📳 📪 🖓 😓 🖄 🗐 🗐 🕒 🔛 🏙 🚅	
B- WebBar	New	,	*	
- h Sour	D-214		* To change this template, choose Tools Templates	+
	Clean and Build		* and open the template in the editor.	E
	Clean		*/	
6	Verify			
	Generate Javadoc		ackage db;	
🖃 📇 S	D			
	Deploy		<pre>mport java.io.Serializable;</pre>	
🕀 🊹 Test	Debug		<pre>mport javax.persistence.Basic;</pre>	
🕀 🍙 Test	Profile		<pre>mport javax.persistence.Column;</pre>	
🕀 🐻 Enter	Test RESTful Web Services	5	<pre>mport javax.persistence.Entity;</pre>	
🕀 📐 Conf	Test	Alt+F6	<pre>mport javax.persistence.Id;</pre>	
tervi	Unset as Main Project		mport javax.persistence.NamedQueries;	
	Open Required Projects		mport javax.persistence.NamedQuery;	
	Close		mport javax.persistence.Table;	
	Panama			
	Move		**	
	Copy		*	
	Delete	Delete	* Cauthor Geni	-
	Find	Ctrl+F		
	Inspect and Transform		B Database Process x GlassFish Server 3+ x	
	Versioning	•	Automatic timeout occured	*
	Local History	•	Automatic timeout occured	
	Properties		Automatic timeout occured	
	Topenes	TNEO	Jauromatic timeout occured	
		INFO:	Automatic timeout occured	
		INFO:	Automatic timeout occured	
		201201	ANA ANNU ANA ANNU ANA AFANYA	-
		•	11. 11.	•
				🔞 1 1 INS
🚯 🜔		۷		▲ 🎼 🛱 🔐 🖣 2:15 PM

Click with right of the mouse on the Project. Select Properties

Select Libraries

Project Properties - WebBankingA	pp		×
<u>C</u> ategories:			
Sources	Java Platform: JDK 1.7 (Default)	•	Manage Platforms
Frameworks	L <u>i</u> braries Folder:		Browse
e ○ Build	Compile Processor Compile Tests Pup Tests		
···· Compiling			
Packaging Decimenting	Compile- <u>ti</u> me Libraries:		
····· · · · · · · · · · · · · · · · ·	Name	Package	Add Project
Formatting			Add Library
			Add JAR/ <u>F</u> older
			Edit
			Ear
			Remove
			Maya Lip
			Move <u>D</u> own
	Compile-time libraries are propagated to all library categories.		
	Build Required Projects (Libraries and additional WAR content)		
		ОК	Cancel <u>H</u> elp

Select on the right "Add JAR/Folder":

Add JAR/Folde	r	×
Look in:] Test	▼] 🔁 📸
Recent Items	HelloWorld WebBankingApOK WebBankingApp @ mysql-connector-java-5.1.15-bin.jar	Reference as <u>Relative Path:</u> /mysql-connector-java-5.1.15-bin.jar Path from <u>V</u>ariable:
Desktop		<pre><no suitable="" variable=""> </no></pre>
My Documents		
Computer		
	File <u>n</u> ame: mysql-connector-java-5.1.15-bin.jar	Open
Network	Files of type: Classpath Entry (folder, ZIP or JAR file)	Cancel

Select the file

mysql-connector-java-5.1.15-bin.jar

Now the Java program is properly set to connect to MySQL.

Project Properties - WebBankingApp	plate, choose Fools Templates	×
Categories:		
Sources	Java Platform: JDK 1.7 (Default)	Manage Platforms
Frameworks Libraries	Libraries Folder:	Browse
⊜… ○ Build	Compile Processor Compile Tests Run Tests	
O Packaging	Compile-time Libraries:	
Occumenting Run	Name Package	Add Project
Formatting	C:\Users\Geni\Desktop\DISTSYS\MYSLIDES\LESSON 9\Test\mys	Add Library
		Add JAR/Folder
		Edit
		Remove
		Mauralla
		Meus Dawa
		Move Down
	Compile-time libraries are propagated to all library categories.	
	Build Required Projects (Libraries and additional WAR content)	
	OK	Cancel Help

Create the project:

I I	letBe	ans ID	E 6.8		
File	Edit	View	Navigate	Source	Refactor
2	New	/ Projec	t	Ctrl+Shif	t+N
1	New	File		Ctrl+N	
	🖁 Ope	n Proje	ct	Ctrl+Shif	t+0
	Ope	n Recei	nt Project		•
	Ope	n Kenai	Project		
	Clos	e Proje	ct		
	Ope	n File			
	Ope	n Recei	nt File		•
	Proj	ect Gro	up		•
	Proj	ect Proj	perties		
	Imp	ort Proj	ect		•
	Sav	e		Ctrl+S	
	Sav	e As			
Ę	Sav	e All		Ctrl+Shif	it+S
	Pag	e Setup			
	Prin	t		Ctrl+Alt+	-Shift+P
	Prin	t to HTM	4L		
	Exit				



🗊 New Project		Σ	<
Steps 1. Choose Project 2	Choose Project Categories: Dava DavaFX DavaFX Dava Web Dava Web Dava EE Dava ME Maven NetBeans Modules Samples	Projects: Web Application Web Application with Existing Sources Web Free-Form Application	-
	Description: Creates an empty Web application uses an IDE-generated build script	on in a standard IDE project. A standard project of to build, run, and debug your project. Next > Einish Cancel Help	

Rew Web Application		X
Steps	Name and Location	
 Choose Project Name and Location Server and Settings Frameworks 	Project Name: WebBankingApp Project Location: js\Geni\Desktop\ADV-OPSYS 2010-2011\MY SLIDES\LESSON 9\Test	Browse
	Project Folder: DIADV-OPSYS 2010-2011 (MY SLIDES(LESSON 9(Test(WebBankingApp Use Dedicated Folder for Storing Libraries]
	Libraries Folder: Different users and projects can share the same compilation libraries (see Help for details).	Browse
	✓ Set as Main Project	
	< <u>B</u> ack Next > Einish Cancel	<u>H</u> elp

New Web Application		
Steps	Server and Set	tings
 Choose Project Name and Location 	Add to Enterprise	Application: <pre> </pre> <pre> </pre>
Server and Settings Frameworks	Server:	GlassFish Server 3.1.2 Add
	Java EE Version:	Java EE 6 Web 👻
		Enable Contexts and Dependency Injection
		V Set Source Level to 6
		Recommendation: Source Level 6 should be used in Java EE 6 projects.
	Context Path:	/WebBankingApp
		< <u>Back</u> Next > Einish Cancel Help

New Web Application	
Steps 1. Choose Project 2. Name and Location 3. Server and Settings 4. Frameworks	Frameworks Select the frameworks you want to use in your web application. Spring Web MVC 2.5 JavaServer Faces Struts 1.3.8 Hibernate 3.2.5
	< <u>B</u> ack Next > <u>Finish</u> Cancel <u>H</u> elp

The application created is this one:



Create a Servlet:

W We	ebBa	nking	App - Ne	tBeans	IDE 6.8								
File E	Edit	View	Navigate	Source	Refactor	Run	Debug	Profile	Team	Tools	Window	v Help	
1	٢		5	9 C				¥ (77	9 🕨	3	• 🕚 •	
Proj.		4 0 ×	Files		Services		🗊 ind	ex.jsp 🏾 »	٢				
□ ∰) We	bBar	New			•	-	Entity C	lasses fro	om Data	base		
-	±		Build	d Duild				Servlet. Session	 Bean				
ļ ļ	- Ca	Sour	Clean ar				Ð	JSP					a.
	 A	Tark	Generat	e Javado	5		6	HTML					
Ē		Libra	Run					Java Cla	ass				
Þ		Test	Deploy					Fotity C	ckage lass				E1
Ξ	- <u>C</u>	Conf	Debug Profile					JSF Pag	es from E	Entity Cl	asses		31
			Test RES	5Tful Web	Services			- Web Ser	rvice	·			m
			Test			Alt+F6		Web Ser	rvice fror	n WSDL.			
		-	Set as M	lain Proje	ct		<u> </u>	Web Ser	rvice Clie	nt			
			Open Re	equired Pr	ojects		S	RESTful	Web Ser	vices fro	om Entity	/ Classes.	·· Þ:
			Close					RESTful	Web Ser	vices fro	om Patte	rns	E
			Rename					Message	e-Driven	Bean			
			Move				P	Other					1:
			Copy				18	-	<th>ly></th> <th></th> <th></th> <th></th>	ly>			
		_	Delete				19	- <th>ml></th> <th></th> <th></th> <th></th> <th></th>	ml>				
			Find Share or Versionir Local His	n Kenai ng itory		•		mpile- n-place	atabas jsps: deplo	e Proce	at D:	GlassFi	sh v3 ents
							Ir	hitiali	zing	-			

New Servlet			×
Steps	Name and L	ocation	
 Choose File Type Name and Location Configure Servlet Deployment 	Class <u>N</u> ame:	BankingServlet	
	<u>P</u> roject:	WebBankingApp	
	Location:	Source Packages	~
	Package:	server	~
	<u>C</u> reated File:	0-2011\MY SLIDES\LESSON 9\Test\WebBankingApp\src\java\server\BankingServlet.jav	'a
		< <u>B</u> ack Next > Einish Cancel Help	

New Servlet			
Steps	Configure Servlet D	eployment	
 Choose File Type Name and Location Configure Servlet Deployment 	Register the Servlet w specify patterns that is commas.	ith the application by giving the Servlet an internal dentify the URLs that invoke the Servlet. Separate o deployment descriptor (web.xml)	name (Servlet Name). Then multiple patterns with
	<u>⊂</u> lass Name:	server.BankingServlet	
	<u>S</u> ervlet Name:	BankingServlet	
	URL Pattern(s):	/BankingServlet	
	Initialization Param	neters:	
	Name	Value	New
			<u>E</u> dit
			Delete
	5		
		< Back Next > Einish	Cancel <u>H</u> elp

Create an EJB to connect to the database MySQL:

🗊 WebBanki	ngApp - NetBeans IDE 6.8	}					
File Edit View	v Navigate Source Refacto	r Run D)ebug	Profile Team To	ols Window Help		
1 1	- 5 C			🗹 T 😿	Image:		
Proj 🔍 🗸	Files Services	• E	ind 👌	ex.jsp 🗙 🚳 Banki	ngServlet.java 🗙		
🖃 💮 WebBa	New	•	8	Servlet			
	Build		8	Entity Classes from D)atabase		
T.	Clean and Build		ß	Session Bean			
🖕 🦳 🔂 Sou	II Clean		Þ	JSP			
ė 	Generate Javadoc		6	HTML	,		
	Rup		ß	Java Class	ſ		
🖽 📺 Tes			Ŧ	Java Package			
E Tes	, Debug		🚳 Entity Class				
E Cor	Profile		Þ	JSF Pages from Entit	y Classes		
	Test RESTful Web Services		ß	Web Service	e		
	Test	Alt+F6	ß	Web Service from W	SDL		
	Set as Main Project		Q	Web Service Client	•		
	Open Required Projects		S	RESTful Web Service	s from Entity Classes		
	Close		S	RESTful Web Service	s from Patterns (
	Rename		3	Message-Driven Bea	n		
	Move		÷	Other	ŧ		
	Copy	T	34	try	• {		
	Delete		35		/* TODO output :		
	Find		36		out.println(" <ht< th=""></ht<>		
	Share on Kenai		37		out.println(" <he< th=""></he<>		
	Versioning	•	30		out.println(" <t:< th=""></t:<>		

New Entity Classes from Data	base	×
Steps	Database Tables	
 Choose File Type Database Tables Entity Classes 	Data Source: Database Schema <no database="" in="" project="" schemas="" the=""></no>	✓
4. Mapping Options	Available Tables: Selected Tables:	
	Add > < Remove Add All >> << Remove All	
	Include Related Tables	
	Select the table source.	
	< <u>B</u> ack Next > Einish Cancel	<u>t</u> elp

New Entity Classes from Data	base	X
Steps 1. Choose File Type 2. Database Tables 3. Entity Classes 4. Mapping Options	Database Tables • Database Schema Available Tables: · Mew Data Source · Add > · Add All >> · Add All >> · Add All >>	
	Select the table source.	
	< <u>B</u> ack Next > Einish Cancel Help	<u>,</u>

🗊 Create Data Source 🛛 🔀				
JNDI Name:	jndi/bank			
Database Connection:				
A data source with the specified JNDI name already exists.				

If you do not have the connection to MySQL select "New Database Connection" as follows:

🗊 Create Data Sou	rce	X
<u>J</u> NDI Name:	jndi/bank	
Database Connection:		~
🚯 A data source with	jdbc:derby://localhost:1527/sample [app on APP] jdbc:mysql://localhost:3306/bank [root on Default schema] New Database Connection	
	OK Cancel <u>H</u> elp	

The following page appears:

🗊 New Database Conne	ction	×		
Basic setting Advanced		^		
Data Input <u>M</u> ode:				
Driver <u>N</u> ame:	Java DB (Network)	~		
Hos <u>t</u> :				
<u>P</u> ort:				
<u>D</u> atabase:				
User Name:				
Pass <u>w</u> ord:				
Displa <u>v</u> Name (Optional):	Database password			
	Remember password (see help for information on security risks)			
<u>A</u> dditional Props:				
Show <u>J</u> DBC URL				
Please specify a value for the required field Database:				
<				
	OK Cancel <u>H</u> elp			

New Connec	tion Wizard	×
Locate Driver		
Driver: MySC	QL (Connector/J driver)	•
Driver File(s):	C:\Program Files\WetBeans 7.2\ide\modules\ext\mysql-connector-java-5.1.18-bin.jar	Add
		Remove
	۰ الله الله الله الله الله الله الله الل	
	< Back Next > Finish Cancel	Help

Fill in the fields as follows;

New Connect	tion Wizard	x
Customize Co	nnection]
Driver <u>N</u> ame:	MySQL (Connector/J driver)	•
Hos <u>t</u> :	localhost Port: 3306	
<u>D</u> atabase:	bank	
<u>U</u> ser Name:	root	
Pass <u>w</u> ord:	••••	
	<u> <u> R</u>emember password </u>	
	Test <u>C</u> onnection	
JDBC URL:	jdbc:mysql://localhost:3306/bank?zeroDateTimeBehavior=convertToNull	
	< <u>B</u> ack Next > Einish Cancel H	elp

Press Test Connection to check your connection:

Driver Name	
Driver <u>N</u> ame;	MySQL (Connector/J driver)
Hos <u>t</u> :	localhost Port: 3306
Database:	bank
User Name:	root
Pass <u>w</u> ord:	
	Test Connection
JDBC URL:	jdbc:mysql://localhost:3306/bank?zeroDateTimeBehavior=convertToNull
(1 Connection	Succeeded.

New Connectio	on Wizard	×	
Choose Databas	se Schema		
For each database connection, the Services window only displays objects from one database schema. Select the schema of the tables to be displayed.			
Select schema:	<no schema=""></no>	T	
	< <u>Back</u> Next > <u>Finish</u> Cancel <u>H</u>	<u>t</u> elp	

Press Finish:

Create Data Source	2		×
JNDI Name:	jndi/bank		
Database Connection:	jdbc:mysql://localhost:3306/	/bank?zeroDateTimeBehavior	=c ▼
	_		
		OK Cancel	Help

Press OK and you should have the following page:

Choose

New Entity Classes from Data	base		×
Steps	Database Tables		
Choose File Type 2. Database Tables S. Entity Classes Monoing Options	● Data Source: ○ Database Schema	indi/bank <no database="" in="" project="" schemas="" the=""></no>	>
4. Mapping Options	Available <u>T</u> ables:	Selected Tables:	
	account accountcustomer customer	Add > < Remove Add All >> << Remove All	
		Include Related Tables	
	🚯 Select at least one l	able.	
		< <u>B</u> ack Next > Einish Cancel <u>H</u> elp	

Now select the tables with "Add All":

New Entity Classes from Data	base		\mathbf{X}
Steps	Database Tables		
 Choose File Type Database Tables Entity Classes 	• Data Source:	jndi/bank	~
Mapping Options	Available <u>T</u> ables:	S <u>e</u> lected Tables:	
		Add > account Add > customer Add All >> Add All >> << Remove All	
		✓ Include Related Tables	
		< <u>B</u> ack Next > Einish Cancel Help	

New Entity Classes from Database				
Steps	Entity Classes			
1. Choose File Type 2. Database Tables	Specify the names and the location of the entity classes.			
3. Entity Classes 4. Mapping Options	<u>C</u> lass Names:	Database Table	Class Name	Generation Type
		account	Account	New
		accountcustomer	Accountcustomer	New
		customer	Customer	New
	Project: WebBankingApp Location: Source Packages Package: db			
	Generate Named Ouery Annotations for Persistent Fields			
	Create Persistence Unit			
		_		
			< <u>Back</u> Next > F	inish Cancel <u>H</u> elp
New Entity Classes from Database				
--	--			
Steps	Mapping Options			
 Choose File Type Database Tables Entity Classes Mapping Options 	Specify the default mapping options. Association Eetch: default Collection Type: java.util.Collection Fully Qualified Database Table Names			
	Use Defaults if Possible			
	Generate Fields for Unresolved Relationships			
	< <u>B</u> ack Next > <u>Finish</u> Cancel <u>H</u> elp			

The following will be generated:



In BankingServlet code the following:

rvices	🝺 inde>	ajsp 🗴 🙆 BankingServlet.java 🗴
	🔀 🔀	·
	16 🖵	/**
	17	*
	18	* @author Geni
	19	*/
	20	<pre>@WebServlet(name="BankingServlet", urlPatterns={"/BankingServlet"})</pre>
	21	<pre>public class BankingServlet extends HttpServlet {</pre>
njava IDK isus	22	
PNJAVA	S	0 <u>PersistenceUnit</u>
	S	EntityManagerFactory emf;
	25	
DVI	26 🖵	/ * *
	27	* Processes requests for both HTTP <code>GET</code> and <code>POST<!--</th--></code>
	28	* @ param request servlet request

Arrange the imports. Click on the red point on the left shown by NetBeans.





You will see that in the imports there are two more lines:

8 5	-	import	java.io.IOException;
9		import	java.io.PrintWriter;
10		import	javax.persistence.EntityManagerFactory;
11		import	javax.persistence.PersistenceUnit;
12		import	javax.servlet.ServletException;
13		import	javax.servlet.annotation.WebServlet;
14		import	javax.servlet.http.HttpServlet;
15		import	javax.servlet.http.HttpServletRequest;
16	L	import	<pre>javax.servlet.http.HttpServletResponse;</pre>
17			

Now we go the point of generating the webpage.



Uncomment the commented part as follows:

```
38
                    PrintWriter out = response.getWriter();
      39
                    try (
      40
                        // TODO output your page here
      41
                        out.println("<html>");
      42
                        out.println("<head>");
/a
      43
                        out.println("<title>Servlet BankingServlet</title>");
      44
                        out.println("</head>");
      45
                        out.println("<body>");
                        out.println("<h1>Servlet BankingServlet at " + request.getContextPath () + "</h1>");
      46
      47
                        out.println("</body>");
                        out.println("</html>");
      48
      49
      50
                    } finally {
      51
                        out.close();
      52
                    }
      53
                }
```

Now let us use the EJB we created. Perform the following operations:

		javax.persistence.EntityManagerFactory
Services	🗊 index.jsp 🗙 🚳 BankingServlet.java 🗙	
	📴 💀 - 🔍 🔍 🖓 🖓 🖶 🔗 😓 🗠	public <u>EntityManager</u> createEntityManager()
	36 🗐 🛛 throws ServletException,	Create a new application-managed EntityManager. This method returns a new
	37 response.setContentTy	EntityManager instance each time it is invoked. The isOpen method will return true
20	38 PrintWriter out = res	on the returned instance.
	39 try (
ut iava	40 // TODO output yo	
itcustomer java	41 out.println(" <htm< th=""><th>Returns:</th></htm<>	Returns:
structomerPK java	42 out.println(" <hea< td=""><td>entity manager instance</td></hea<>	entity manager instance
her java	43 out.println(" <tit< td=""><td>Throws:</td></tit<>	Throws:
	44 out.println(" <td>IllegalStateException - if the entity manager factory has been closed</td>	IllegalStateException - if the entity manager factory has been closed
-Sarulat java	45 out.println(" <bod< th=""><th></th></bod<>	
goer viec, java	46 out.println(" <hi></hi>	
	47	
	48 Customer cust = e	mf.
iles	49	😑 close () void 📥
100	50 out.println(" <th>\ominus createEntityManager() EntityManager</th>	\ominus createEntityManager() EntityManager
	51 out.println(" <td>🕘 createEntityManager(Map map) EntityManager</td>	🕘 createEntityManager(Map map) EntityManager
	52	🥥 equals (Object obj) boolean
	53) finally (\ominus getCache() Cache
	54 out.close();	⊖ getClass() Class
	55 }	<pre> getCriteriaBuilder() CriteriaBuilder </pre>
	56 L }	<pre> getMetamodel() Metamodel </pre>
	57	\ominus getPersistenceUnitUtil() PersistenceUnitUtil
	58 + HttpServlet methods. Clic	<pre> getProperties() Map<string, object=""> </string,></pre>
	93	\ominus hashCode() int
		🥥 isOpen() boolean
ator 🗦 🛪	Output	<pre>ontify() void</pre>
~	🕪 🛛 Java DB Database Process 🗴 🛛 GlassFish v3 🛛	😑 notifyAll() void
		😑 toString() String
rvletRequest reque 🔽	mpile-jsps:	😑 wait() void
>	itializing	🕒 wait(long timeout) void 🔽
_		

Then:

: L					jauay parsistanca EntituManagar
es	🗊 in	dex.jsp 🗙 🐼 BankingServlet.	java x		
		🗟 • 🗐 • 尾 • 🖓 🖓 😜	중 ॡ 중 월 일 ●		public <u>Query</u> createNamedQuery(<u>String</u> name)
	36	throws ServletE	xception, IOExcept	ion {	Create an instance of Query for executing a named query (in the Java Persistence
	37	response.se	tContentType("text,	/html;charset=	query language or in native SQL).
	38	PrintWriter	out = response.ge	tWriter();	
	39	try (
	40	// TODO) output your page :	here	Parameters:
	41	out.pri	<pre>ntln("<html>");</html></pre>		name - the name of a query defined in metadata
5VE	42	out.pri	<pre>ntln("<head>");</head></pre>		Returns:
java	43	out.pri	ntln(" <title>Servl</title>	et BankingServ	, the new query instance
	44	out.pri	$\operatorname{ntln}("$		Throws:
	45	out.pri	ntln(" <body>");</body>		java.lang.IllegalArgumentException - if a query has not been defined
1	46	out.pri	ntln(" <h1>Servlet</h1>	BankingServlet	with the given name or if the query string is found to be invalid
	47				
	48	Custome	r cust = emf.creat	eEntityManager	:().
	49				⊖ clear()
	50	out.pri	<pre>ntln("");</pre>		⊖ close()
	51	out.pri	<pre>ntln("");</pre>		<pre>contains(Object entity)</pre>
	52				<pre>oreateNamedQuery(String name)</pre>
	53) finally (<pre>oreateNamedQuery(String name, Class<t> resultClass) TypedQ</t></pre>
	54	out.clo	se();		<pre>oreateNativeQuery(String sqlString)</pre>
	55	}			<pre>oreateNativeQuery(String sqlString, Class resultClass)</pre>
	56	L }			oreateNativeQuery(String sqlString, String resultSetMapp.
	57				<pre>oreateQuery(CriteriaQuery<t> criteriaQuery) TypedQ</t></pre>
	58	+ HttpServlet met	thods. Click on the	e + sign on the	🛛 🔵 createQuery(String qlString)
	93				<pre>oreateQuery(String qlString, Class<t> resultClass) TypedQ</t></pre>
	04	<u>،</u>			🔵 🕘 detach (Object entity)
₽ x	Outp	ut			😑 equals(Object obj)
		Java DB Database Process 🛛 🗙	GlassFish v3 Domain ×	WebBankingApp (run)) find(Class <t> entityClass, Object primaryKey)</t>
		ompire.	11	1	<pre>find(Class<t> entityClass, Object primaryKey, LockModeType</t></pre>
		compile-jsps:			

You have now:

44	<pre>out.println("");</pre>
45	<pre>out.println("<body>");</body></pre>
46 47	out.println(" <h1>Servlet BankingServlet at " + request.getContextPatH</h1>
9 49	Customer cust = emf.createEntityManager().createNamedQuery(null)

Go to the EJB Customer.java and copy a query name as follows:

```
14
                               import javax.persistence.iu;
Source Packages
                         13
                               import javax.persistence.NamedQueries;
🕮 db
                         14
                               import javax.persistence.NamedQuery;
  🚳 Account.java
                            import javax.persistence.Table;
                         15

    Accountcustomer.java

                         16
  AccountcustomerPK.java
                         17 - /**
  🚳 Customer.java
                         18
📴 server
                                * @author Geni
                         19
  👩 BankingServlet.java
                         20
                               */
Test Packages
                         21
                               @Entity
Libraries
                               @Table(name = "customer")
                         22
Test Libraries
                         23
                               @NamedQueries({
Configuration Files
                                   @NamedQuery(name = "Customer.findAll", query = "SELECT c FROM Customer c").
                         24
                                   @NamedQuery(name = "Customer.findByIdCustomer", query = "SELECT c FROM Cust
                         25
                                   @NamedQuery(name = "Customer.findByName", query = "SELECT c FROM Customer (
                         26
                                   @NamedQuery(name = "Customer.findBySurname", query = "SELECT c FROM Customet")
                         27
                         28
                               public class Customer implements Serializable {
                         29
                                   private static final long serialVersionUID = 1L;
                         20
                                   а ты
```

```
Copy "Customer.findAll"
```

Paste what you copied from Customer, in the following window in the servlet. Now get the results from th database as follows:

Returns:
a list of the results
Throws:
IllegalStateException - if called for a Java Persistence query language
UPDATE or DELETE statement
QueryTimeoutException - if the query execution exceeds the query
timeout value set and only the statement is rolled back
in no transaction reduception - II a lock mode has been set and there
IS NO U di ISdauton Recentratente de la Ruccusti en la fragoniziatio la duipa foilo pad the
<pre>JamedQuery("Customer.findAll").</pre>
🥥 equals(Object obj) boolean
<pre>executeUpdate() int</pre>
<pre> getClass() Class<?> </pre>
<pre>getFirstResult() int</pre>
<pre> getFlushMode() FlushModeType </pre>
<pre> getHints() Map<string, object=""> </string,></pre>
<pre> getLockMode() LockModeType </pre>
<pre>getMaxResults() int</pre>
<pre> getParameter(String name) Parameter<?> </pre>
edit 🔵 getParameter(int position) Parameter
getParameter(String name, Class <t> type) Parameter<t></t></t>
getParameter(int position, Class <t> t Parameter<t></t></t>
sks getParameterValue(Parameter <t> param) T</t>
<pre> getParameterValue(String name) Object </pre>
getParameterValue(int position) Object
OlO_2 GetParameters () Set <parameter<?>></parameter<?>
● getResultList() List

Take the first record with get(0) as follows:



```
Now the code is:
```

```
out.println("<h1>Servlet BankingServlet at " + request.getContextPath () + "</h1>");
Customer cust = emf.createEntityManager().createNamedQuery("Customer.findAll").getResultList().get(0);
out.println("</body>");
out.println("</html>");
```

You still need to cast as follows by adding (Customer) before the statement::



Now arrange the imports as follows:



Now we need to print the data on the web page that is going to be generated by the servlet. Write the following code as shown below to get the name of the customer:

<pre>public String getName() public String getName() p</pre>	ou
<pre>public string getName() public string getName() p</pre>	οu
<pre>s response.setContentType("text/html;charset=UTF-8"); PrintWriter out = response.getWriter(); try { // TODO output your page here out.println("<html>"); out.println("</html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></pre>	rou
<pre>PrintWriter out = response.getWriter(); try { // TODO output your page here out.println("<html>"); out.println("</html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></pre>	
<pre>b try { // TODO output your page here out.println("<html>"); out.println("<html>"); out.println("<html>"); out.println("<html>"); </html></html></html></html></pre>	
<pre>1 // TODO output your page here 2 out.println("<html>"); 3 out.println("<html>"); 4 out.println("<itle>Servlet_BankingGervlet");</itle></html></html></pre>	
<pre>2 out.println("<html>"); 3 out.println("<head>"); 4 out.println("<title>Servlet_BankingGervlet</title>");</head></html></pre>	
<pre>3 out.println("<head>"); aut.println("<title>Servlet BankingGervlet</title>");</head></pre>	
1 out mrintln/" <title>Servlet BankingServlet</title> ").	
The second	
<pre>5 out.println("");</pre>	
<pre>5 out.println("<body>");</body></pre>	
7 out.println(" <h1>Servlet BankingServlet at " + reques</h1>	
3	
Customer cust = (Customer)emf.createEntityManager().c	
) out.println(" <h2> The name of the customer is: " + cust. + "</h2> ");	
• getName() String	
<pre>2 out.println("");</pre>	=
<pre>3 out.println("");</pre>	
1 	
5) finally (Olass	
5 out.close(); OgetIdCustomer() Integer	
7)) int	
3 L) Ontify() void	
ontifyAll() void	~
HttpServlet methods Click on the + sign on the left to edit estimate (Integer idCustomer) void	A
SetName (String name) void	-
void void	
Java DB Database Process × GlassFish v3 Domain × WebBankingApp (run) × Ovid void	
Browsing: http://localhost:8080/WebBankingApp/	^
run-display-browser:	

Now the code looks as follows:

Customer cust = (Customer)emf.createEntityManager().createNamedQuery("Customer.findAll").getResultList().get(0); out.println("<h2> The name of the customer is: " + cust.getName() + "</h2>");

If you want to take also the surname add as follows:

```
Customer cust = (Customer)emf.createEntityManager().createNamedQuery("Customer.findAll").getResultList().get(0);

out.println("<h2> The name of the customer is: " + cust.getName() + "</h2>");

put.println("<h2> The surname of the customer is: " + cust.getSurname() + "</h2>");
```

Save the project and run it.

sr 🌏	🕹 JSP Page - Mozilla Firefox								
Eile	<u>E</u> dit	⊻iew	Hi <u>s</u> tory	<u>B</u> ook	marks	<u>T</u> ools	Help		
<)>	- C	×		÷.		http://localhost:8080/WebBankingApp/		
ዾ м	ost Visit	ed 🗋	Getting S	tarted	🚵 La	test Hea	dlines		
	JSP Pa	ge					÷		

Hello World!

Add the servlet name in the browser:

http://localhost:8080/WebBankingApp/BankingServlet

The following will appear:



Servlet BankingServlet at /WebBankingApp

The name of the customer is: Leo

The surname of the customer is: Messi

If we want to take some data about the accounts:

Go to the EJB Account.java and copy a query name as follows:

		12	11	immort javay nersistence.Id:	
- 🖨 - 🔐	Source Packages	10			
	🕮 db	13		import javax.persistence.NameaQueries;	
	Account java	14	1	<pre>import javax.persistence.NamedQuery;</pre>	
		15	L	<pre>import javax.persistence.Table;</pre>	
	Accountcustomer.java	16			
	AccountcustomerPK.java	17		/**	
Customer.java					
- i	- 💭 server	18		*	FCT a FROM Account a").
	BankingSerulet java	19		* @author Geni	
1~~	Test Packages 20 */ Libraries 21 @Entity	*/			
		21		ØFntite	
		22			
	Test Libraries	44		<pre>[lable(name = "account")</pre>	
÷	Configuration Files	23	1	@NamedQueries({	
±[3	coningaracion nico	24	1	<pre>@NamedQuery(name = "Account.findAll", query = "SELECT a FROM Account a"),</pre>	
		25		<pre>@NamedQuery(name = "Account.findByIdAccount", query = "SELECT a FROM Account</pre>	a WHERE a.idA
		26		<pre>@NamedQuery(name = "Account.findByBalance", query = "SELECT a FROM Account a</pre>	WHERE a.balar
		27		public class Account implements Serializable (
		28		<pre>private static final long serialVersionUID = 1L;</pre>	
		29		ØId	

```
Copy "Account
```

```
@Entity
@Table(name = "account")
@NamedQueries({
    @NamedQuery(name = "Account.findAll", query = "SELECT a FROM Account a"),
    @NamedQuery(name = "Account.findByIdAccount", query = "SELECT a FROM Account a
    @NamedQuery(name = "Account.findByBalance", query = "SELECT a FROM Account a
    @NamedQuery(name = "Account.findByBalance", query = "SELECT a FROM Account a
    @Public class Account implements Serializable {
    private static final long serialVersionUID = 1L;
    @Id
```

Add the following code:

Customer cust = (Customer)emf.createEntityManager().createNamedQuery("Customer.findAll").getResultList().get(0);
out.println(" <h2> The name of the customer is: " + cust.getName() + "</h2> ");
out.println("< <u>h2> The surname of</u> the customer is: " + cust.getSurname() + ""); [Variable account is not used]
Account account = (Account) emf.createEntityManager().createNamedQuery("Account.findAll").getResultList().get(0);

Arrange the imports by click the red point on the left shown by NetBeans:



Now add the code to the id of the account and the balance as follows:

```
Account account = (Account)emf.createEntityManager().createNamedQuery("Account.findAll").getResultList().get(0);
out.println("<h2> The ID of the account is: " + account.getIdAccount () + "</h2>");
out.println("<h2> The balance of the account is: " + account.getBalance() + "</h2>");
```

Save the project. If you refresh the browser you will get:

🕹 Se	rvlet	Bankiı	ngServle	et - Mo	ozilla I	Firefox
<u>F</u> ile	<u>E</u> dit	⊻iew	Hi <u>s</u> tory	<u>B</u> ookr	narks	Tools Help
<)>	- C	×		eg .	http://localhost:8080/WebBankingApp/BankingServlet
🔎 Me	🖻 Most Visited 📄 Getting Started 🔝 Latest Headlines					
	5ervle	t Banki	ingServl	et		*

The name of the customer is: Leo

The surname of the customer is: Messi

The ID of the account is: 1

The balance of the account is: 580436.0

We can also add the ID of the customer as follows:

	db.Customer		
🕼 BankingServlet.java 🗴 🗟 Account.java 🗴 🗟 Customer.java 🗴	<pre>public Integer getIdCustomer()</pre>		
<pre>// TODO output your page here out.println("<html>"); out.println("<html>"); out.println("<title>Servlet BankingServlet</title>" out.println("<title>Servlet BankingServlet</title>" out.println("<html>"); out.println("<html>");</html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></html></pre>	Javadoc not found. Either Javadoc documentation for this item does not exist or you have not added specified Javadoc in the Java Platform Manager or the Library Manager.		
out.println(" <h2> The name of the customer is: " +)</h2>			
out.println(" <h2> The ID of the customer is: " + cu</h2>	st. + "");		
Account account = (Account)emf.createEntityManager(out.println(" <h2> The ID of the account is: " + acc out.println("<h2> The balance of the account is: " -</h2></h2>	o getName() String o getSurname() String ResultList().get(0); o toString() String o equals(Object object) boolean o getClass() Class		
<pre>out.println(""); out.println("");</pre>	getIdCustomer() Integer hashCode() int		
<pre>> finally (out.close(); } }</pre>	<pre>void void void void setIdCustomer(Integer idCustomer) void setName(String name) void setSurname(String surname) void</pre>		
	wait() void wait(long timeout) void wait(long timeout, int nanos) void		

The code is:

```
Customer cust = (Customer)emf.createEntityManager().createNamedQuery("Customer.findAll").getResultList().get(0);
out.println("<h2> The name of the customer is: " + cust.getName() + "</h2>");
out.println("<h2> The surname of the customer is: " + cust.getSurname() + "</h2>");
out.println("<h2> The ID of the customer is: " + cust.getIdCustomer() + "</h2>");
Account account = (Account)emf.createEntityManager().createNamedQuery("Account.findAll").getResultList().get(0);
out.println("<h2> The ID of the account is: " + account.getIdAccount() + "</h2>");
out.println("<h2> The ID of the account is: " + account.getIdAccount() + "</h2>");
```

If you refresh the browser you will get:

🕹 Servlet BankingServlet - Mozilla Firefox							
Eile	<u>E</u> dit	⊻iew	Hi <u>s</u> tory	<u>B</u> ookn	narks	Tools Help	
<)>	- C	×		8	http://localhost:8080/WebBankingApp/BankingServlet	
🖻 Most Visited 📄 Getting Started 流 Latest Headlines							
Servlet BankingServlet +				et		÷	

The name of the customer is: Leo

The surname of the customer is: Messi

The ID of the customer is: 1

The ID of the account is: 1

The balance of the account is: 580436.0

If we want to add the relationship between the customer and the account add the following:

```
Customer cust = (Customer)emf.createEntityManager().createNamedQuery("Customer.findÅll").getResultList().get(0);
out.println("<h2> The name of the customer is: " + cust.getName() + "</h2>");
out.println("<h2> The surname of the customer is: " + cust.getSurname() + "</h2>");
out.println("<h2> The ID of the customer is: " + cust.getIdCustomer() + "</h2>");
Accountcustomer accCust = (Accountcustomer)emf.createEntityManager().createNamedQuery("Accountcustomer.findÅll").getResultList().get(0)
out.println("<h2> The customer with account ID: " + accCust.getAccountcustomerPK().getIdAccount() + " has the ID: " +
accCust.getAccountcustomerPK().getIdCustomer() + "</h2>");
Account account = (Account)emf.createEntityManager().createNamedQuery("Account.findÅll").getResultList().get(0);
out.println("<h2> The ID of the account is: " + account.getIdAccount() + "</h2>");
out.println("<h2> The ID of the account is: " + account.getIdAccount() + "</h2>");
account account = (Account)emf.createEntityManager().createNamedQuery("Account.findÅll").getResultList().get(0);
out.println("<h2> The ID of the account is: " + account.getIdAccount() + "</h2>");
out.println("<h2> The balance of the account is: " + account.getBalance() + "</h2>");
```

Refresh the browser and you will get the following:



The name of the customer is: Leo

The surname of the customer is: Messi

The ID of the customer is: 1

The customer with account ID: 1 has the ID: 1

The ID of the account is: 1

The balance of the account is: 580436.0

If you need to retrieve an account directly by ID you can use the named queries by setting the appropriate parameters:

Add the following code:

Check the setParameter method above which sets the parameter idAccount to the value 2.

The result of the query will be:

The name of the customer is: Leo

The surname of the customer is: Messi

The ID of the customer is: 1

The customer with account ID: 1 has the ID: 1

The ID of the account is: 1

The balance of the account is: 580436.0

Retreiving account by ID

The ID of the account is: 2

The balance of the account is: 3000.0

For more details on Creating Queries Using the Java Persistence Query Language you can refer the following Oracle resource:

http://docs.oracle.com/javaee/6/tutorial/doc/bnbrg.html