Today’s session for upgrade service so why we need to implement the upgrade service.

normally what happens when we add one column in service.xml. When we do build service it automatically update the table by that column

why does it happens because we added include-and-override=portal-developer.properties into potal-ext.prperties file

but if I remove that line and add one more column and try to build the service and check wheather check the table updates or not.

What happens in production server there is no file like this “potal-ext.prperties” to there must be not field like include-and-override to resolve this issue liferay provide the upgrade process.

We add dependancy of compileOnly group: "com.liferay", name: "com.liferay.portal.upgrade.api"

Now create a package of upgrade in service moduile adding two files

1file content the class that implements UpgradeStepRegistrator this class also we implement the register method

registry.register(<current-version>"1.0.0",<new-version> "1.0.1",<class which extends the upgrade process>\_addColumnUpgradeProcess);

|  |
| --- |
| @Component(immediate = true, service = UpgradeStepRegistrator.class)  public class ServiceUpgradeStepRegistrator implements UpgradeStepRegistrator{  @Override  public void register(Registry registry) {  \_log.info(">>>>>> UPGRADE PROCESS FOR SERVICE BUILDER : START >>>>>>>");  registry.register("1.0.0", "1.0.1",\_addColumnUpgradeProcess);  // registry.register("1.0.1", "1.0.2",\_addColumnUpgradeProcess2);  // registry.register("1.0.2", "1.0.3",\_addColumnUpgradeProcess3);  \_log.info(">>>>> UPGRADE PROCESS FOR SERVICE BUILDER : END >>>>>>>");  }  @Reference(target = "(component.name=com.stpl.training.saturday.demo.upgrade.v1\_0\_1.AddColumnUpgradeProcess)")  private UpgradeProcess \_addColumnUpgradeProcess;  //  // @Reference(target = "(component.name=com.stpl.training.saturday.demo.upgrade.v1\_0\_2.AddColumnUpgradeProcess)",unbind = "-")  // private UpgradeProcess \_addColumnUpgradeProcess2;  //  // @Reference(target = "(component.name=com.stpl.training.saturday.demo.upgrade.v1\_0\_3.AddColumnUpgradeProcess)",unbind = "-")  // private UpgradeProcess \_addColumnUpgradeProcess3;    protected static final Log \_log = LogFactoryUtil.getLog(ServiceUpgradeStepRegistrator.class);    } |

|  |
| --- |
| @Component(immediate = true, service = UpgradeProcess.class)  public class AddColumnUpgradeProcess extends BaseUpgradeProcess {    @Override  protected void doUpgrade() throws Exception {  \_log.info("----------- Registering Employee Upgrade start -----------");  if (hasTable(EmployeesImpl.TABLE\_NAME)) {  \_addColumn(EmployeesImpl.TABLE\_NAME, "email", "VARCHAR(255) null");  }else {  \_log.info(EmployeesImpl.TABLE\_NAME+" table does not exist in entity");  }  \_log.info("----------- Registering Employee Upgrade end -----------");  }    protected static final Log \_log = LogFactoryUtil.getLog(AddColumnUpgradeProcess.class);    } |

|  |
| --- |
| package com.stpl.training.saturday.demo.upgrade;    import com.liferay.portal.kernel.log.Log;  import com.liferay.portal.kernel.log.LogFactoryUtil;  import com.liferay.portal.kernel.upgrade.UpgradeProcess;    public class BaseUpgradeProcess extends UpgradeProcess {    protected static final Log \_log = LogFactoryUtil.getLog(BaseUpgradeProcess.class);    @Override  protected void doUpgrade() throws Exception {  }    protected void \_addColumn(String tableName, String columnName, String columnType) throws Exception {  \_log.info("Adding column " + columnName + " to the table " + tableName);    if (!hasColumn(tableName, columnName)) {  alterTableAddColumn(tableName, columnName, columnType);  } else {  \_log.info("Column " + columnName + " already exists on table " + tableName);  }  }    protected void \_dropColumn(String tableName, String columnName) throws Exception {  \_log.info(String.format("Dropping column %s from table %s", columnName, tableName));    if (hasColumn(tableName, columnName)) {  alterTableDropColumn(tableName,columnName);  } else {  \_log.info(String.format("Column %s already does not exist on table %s", columnName, tableName));  }  }    } |

|  |
| --- |
| Bundle-Name: employees-service  Bundle-SymbolicName: com.stpl.training.saturday.demo.service  Bundle-Version: 1.0.0  Liferay-Require-SchemaVersion: 1.0.1  Liferay-Service: true  -dsannotations-options: inherit |

|  |
| --- |
| dependencies {  compileOnly group: "com.liferay", name: "com.liferay.portal.upgrade.api"  } |

Now deploying jar or check the table update or notif not you can also use upgrade:check in gogo shell.

it shows your version updated but still not updated table use

upgrade:execute com.stpl.training.saturday.demo.service

You got your changes.