

JAVA DEBUG

<< 201912 > UTP-MB_201912 > Temenos > R20 > Env > Common > Scripts > Slot01					Search Slot
	Name	Date modified	Type	Size	
	JMSReceiver	12/12/2019 1:29 AM	Windows Batch File	1 KB	
	JMSSender	12/12/2019 1:29 AM	Windows Batch File	1 KB	
	SetupSlot	12/12/2019 1:29 AM	Windows Batch File	1 KB	
	StartUp	12/11/2019 12:39 PM	Windows Batch File	2 KB	
	StartUpDES	12/4/2019 12:23 AM	Windows Batch File	3 KB	
	StartUpH2	12/12/2019 1:29 AM	Windows Batch File	1 KB	
	StartUpJboss	2/6/2020 11:00 AM	Windows Batch File	2 KB	
	Stop	12/12/2019 1:29 AM	Windows Batch File	2 KB	

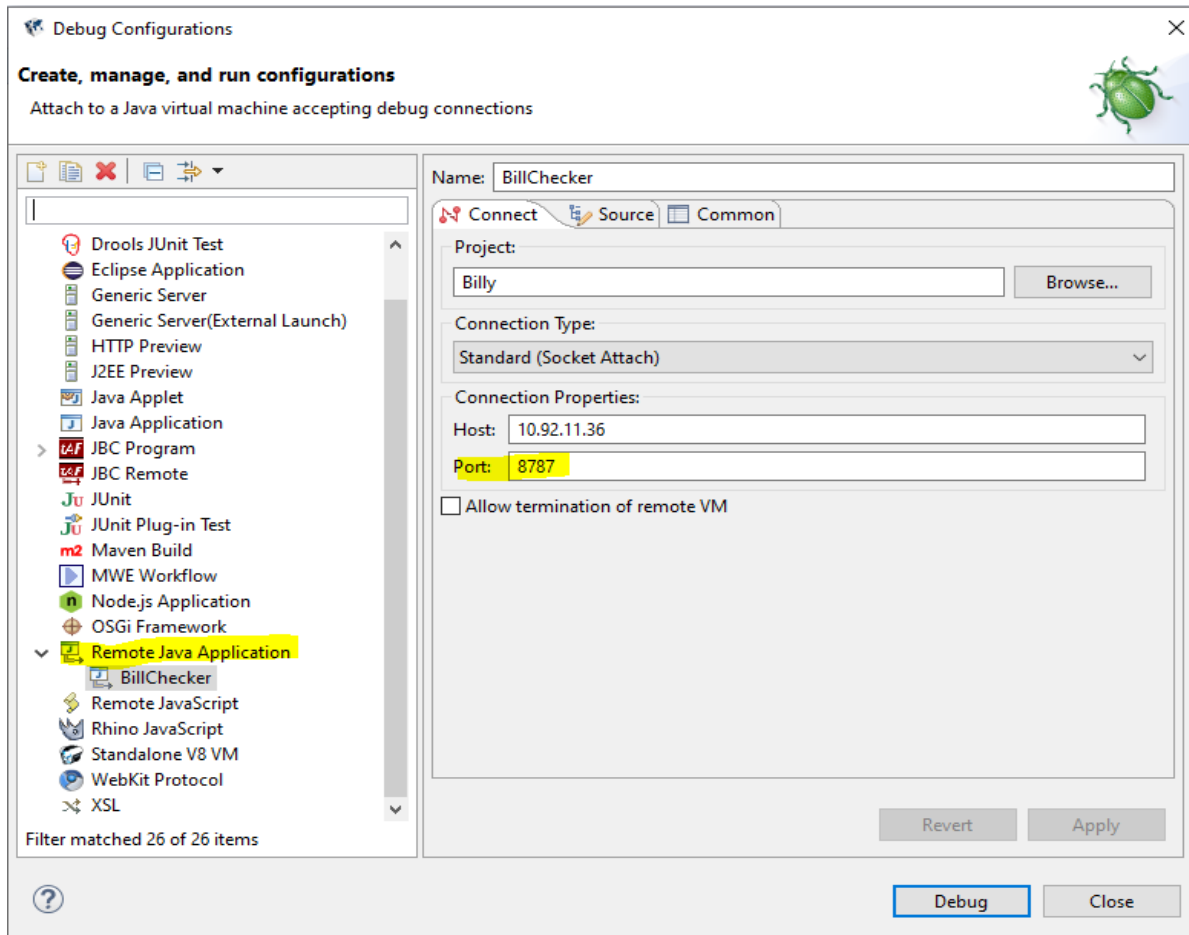
Open the StartUpJboss script for editing purpose.

```
TITLE Application server start for Slot01
@net file l>nml 2>nml && goto :run || powershell -ex unrestricted -Command "Start-Process -Verb RunAs -FilePath '%comspec%' -ArgumentList '/X %~fnx0 %*'
@goto :eof

:run
@CD /D %~dp0
@CALL SetupSlot.bat
REM Start JBoss
@ECHO "Starting JBoss"
SET JAVA_OPTS=%JBOSS_JAVA_OPTS% -Xms8G -Xmx18G -Dfile.encoding=UTF-8 -Xdebug -Xnoagent -Xrunjdpw:transport=dt_socket,address=8787,server=y,suspend=n
CD %JBOSS_HOME%\bin
ECHO CALL standalone.bat --server-config=standalone-%SLOT_NUMBER%.xml -Djboss.server.base.dir=%JBOSS_HOME%\standalone-%SLOT_NUMBER% -DtestDeployFolder=%SLOT_NUMBER%
CALL standalone.bat --server-config=standalone-%SLOT_NUMBER%.xml -Djboss.server.base.dir=%JBOSS_HOME%\standalone-%SLOT_NUMBER% -DtestDeployFolder=%SLOT_NUMBER%
```

For the variable JAVA_OPTS add the following line to debug the L3 API's.

-Xms8G -Xmx18G -Dfile.encoding=UTF-8 -Xdebug -Xnoagent -
Xrunjdpw:transport=dt_socket,address=8787,server=y,suspend=n



In the Design Studio, setup the Debug configuration for the Remote Java Application and set the following HOST can be the localhost or you can specify the ip address and the port should be the same as the “socket-address” as specified in the StartUpJboss script.

```

Select Administrator: Setting up the Slot01 parameters

JBoss Bootstrap Environment

JBOSSE_HOME: "C:\Temenos\201912\UTP-MB_201912\Temenos\R20\Env\Common\Scripts\Slot01\...\Slot01\...\infra\AppServer\JBoss\Default"

JAVA: "C:\Temenos\201912\UTP-MB_201912\Temenos\R20\Env\Common\Scripts\Slot01\...\Slot01\...\infra\Java\Default\bin\java"

JAVA_OPTS: "-Dprogram.name=standalone.bat -Xms8G -Xmx18G -Dfile.encoding=UTF-8 -Xms8G -Xmx18G -Dfile.encoding=UTF-8 -Xdebug -Xnoagent -Xrunjdwp:transport=dt_socket,address=8787,server=y,suspend=n -verbose:gc -XX:+PrintGCDetails -XX:+PrintGCDateStamps -XX:+UseGCLogFileRotation -XX:NumberOfGCLogFiles=5 -Xloggc:"C:\Temenos\201912\UTP-MB_201912\Temenos\R20\Env\Common\Scripts\Slot01\...\Slot01\logs\JBoss\gc.log" -XX:GCLogFileSize=3M -XX:-TraceClassUnloading"

=====
Listening for transport dt_socket at address: 8787
15:26:38,280 INFO [org.jboss.modules] (main) JBoss Modules version 1.6.0.Final-redhat-1
15:26:38,841 WARN [org.jboss.as.server] (main) WFLYSRV0266: Server home is set to 'C:\Temenos\201912\UTP-MB_201912\Temenos\R20\Env\Common\Scripts\Slot01\...\Slot01\...\infra\AppServer\JBoss\Default\standalone-Slot01', but server real home is 'C:\Temenos\201912\UTP-MB_201912\Temenos\R20\Infra\AppServer\JBoss\Default\standalone-Slot01' - unpredictable results may occur.
15:26:38,884 INFO [org.jboss.msc] (main) JBoss MSC version 1.2.7.SP1-redhat-1
15:26:39,006 INFO [org.jboss.as] (MSC service thread 1-8) WFLYSRV0049: JBoss EAP 7.1.0.GA (WildFly Core 3.0.10.Final-redhat-1) starting

```

On starting the Jboss, we can see that the port 8787 is started listening and also check whether the axis2.war is deployed.

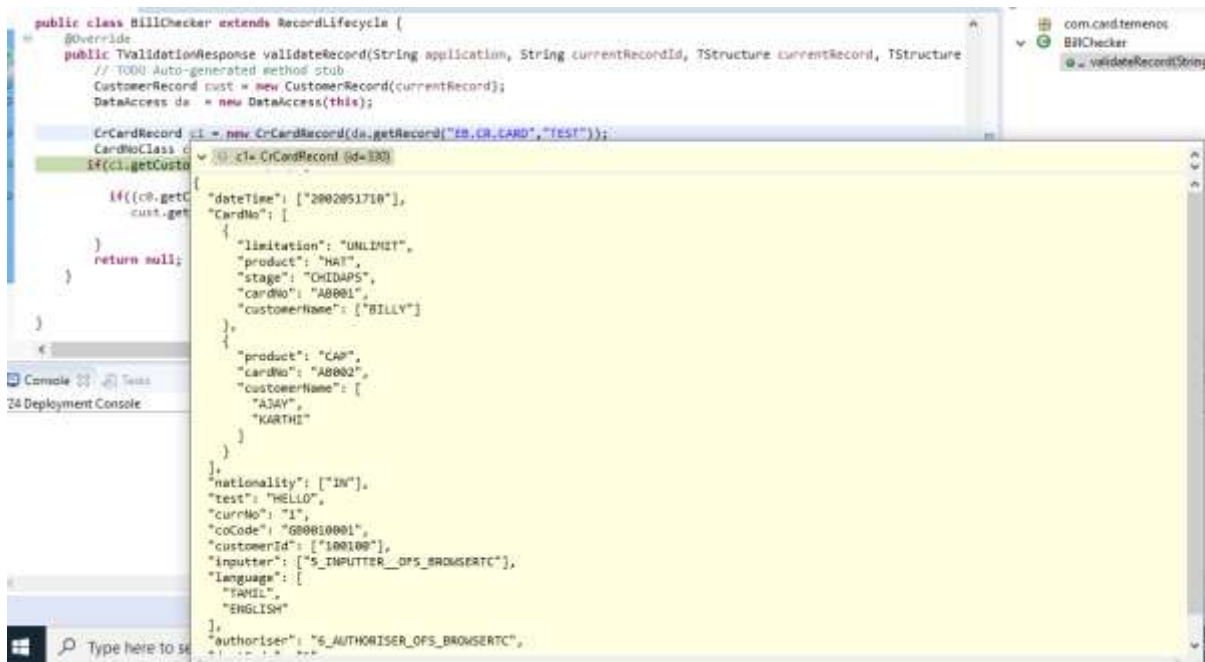
For a Local Application created using the EB.TABLE.DEFENITION, and using the table in the L3 java code and debugging them, a json file representing the table will be popped up when we hover around the object that holds the table record.



The screenshot shows a table viewer window titled 'EB.CR.CARD TEST (Model Bank)'. It contains a list of fields and their corresponding values. The fields are: Customer Id.1, Card No.1, Customer Name.1.1, Product.1, Stage.1, Limitation.1, Card No.2, Customer Name.2.1, Customer Name.2.2, Product.2, Nationality.1, Language.1, Language.2, Test, Curr No, Inputter.1, Date Time.1, Authoriser, Co Code, and Dept Code.

Field	Value
Customer Id.1	100100
Card No.1	AB001
Customer Name.1.1	BILLY
Product.1	HAT
Stage.1	CHIDAPS
Limitation.1	UNLIMIT
Card No.2	AB002
Customer Name.2.1	AJAY
Customer Name.2.2	KARTHI
Product.2	CAP
Nationality.1	IN
Language.1	TAMIL
Language.2	ENGLISH
Test	HELLO
Curr No	1
Inputter.1	S_INPUTTER_OPS_BROWSERTC
Date Time.1	05 FEB 20 17:10 05 FEB 20 17:10
Authoriser	S_AUTHORISER_OPS_BROWSERTC
Co Code	GB-001-0001 Model Bank
Dept Code	1 Implementation

Is the record of the local application.



The screenshot shows an IDE with a Java class named BiliChecker. The class has a method validateRecord that takes application, currentRecordId, TStructure currentRecord, and TStructure. The method is annotated with @Override and has a comment // TODO Auto-generated method stub. The method body contains logic to create a CustomerRecord, get a DataAccess object, and then create a CCardRecord object. The CCardRecord object is created with the following fields: dateTime, cardNo, limitation, product, stage, cardNo, customerName, nationality, test, currNo, coCode, customerId, inputter, language, and authoriser. The JSON representation of the record is displayed as a tooltip when the mouse hovers over the CCardRecord object.

```
public class BiliChecker extends RecordLifecycle {
    @Override
    public TValidationResponse validateRecord(String application, String currentRecordId, TStructure currentRecord, TStructure
    // TODO Auto-generated method stub
    CustomerRecord cust = new CustomerRecord(currentRecord);
    DataAccess da = new DataAccess(this);
    CCardRecord c1 = new CCardRecord(da.getRecord("EB.CR.CARD", "TEST"));
    CardNoClass c
    if(c1.getCusto
        {
            "dateTime": ["2002051710"],
            "cardNo": [
                {
                    "limitation": "UNLIMIT",
                    "product": "HAT",
                    "stage": "CHIDAPS",
                    "cardNo": "AB001",
                    "customerName": ["BILLY"]
                },
                {
                    "product": "CAP",
                    "cardNo": "AB002",
                    "customerName": [
                        "AJAY",
                        "KARTHI"
                    ]
                }
            ],
            "nationality": ["IN"],
            "test": "HELLO",
            "currNo": "1",
            "coCode": "GB0010001",
            "customerId": ["100100"],
            "inputter": ["S_INPUTTER_OPS_BROWSERTC"],
            "language": [
                "TAMIL",
                "ENGLISH"
            ],
            "authoriser": "S_AUTHORISER_OPS_BROWSERTC"
        }
    }
}
```

'c1' is the object and hovering the mouse around the object will display the json file representation of the record.