1. Create RunbaseBatch Class :

**class** TEL\_ClsBillerServiceOrder **extends** RunBaseBatch

{

}

**boolean** canGoBatchJournal()

{

*//true if the class is shown in the list of Journal types; otherwise, false.*

**return** **true**;

}

**public** **container** pack()

{

*//Serializes the current instance of the RunBase class.*

*//return [#CurrentVersion];*

**return** **conNull**();

}

**public** **void** run()

{

*// The purpose of your job.*

info(**strFmt**("Hello from Batch4DemoClass .run at %1"

,DateTimeUtil ::toStr(

DateTimeUtil ::utcNow())

));

}

**public** **boolean** unpack(**container** packedClass)

{

**return** **true**;

}

**public** **static** **void** main(Args args)

{

*// instantiate and object of our class and run it*

*/\* TEL\_ClsBillerServiceOrder self = new TEL\_ClsBillerServiceOrder();*

*;*

*self.run();\*/*

}

1. Run the Incremental CIL
2. Create new job and call the below code :

Batch batch,batch\_del;

BatchJob batchJob;

TEL\_ClsBillerServiceOrder \_ProcessIncrement;

BatchInfo processBatchInfo;

BatchRetries noOfRetriesOnFailure = **4**;

**str** sParmCaption = "Biller Grid Specification";

header = BatchHeader::construct();

\_ProcessIncrement = **new** TEL\_ClsBillerServiceOrder();

processBatchInfo = \_ProcessIncrement.batchInfo();

processBatchInfo.parmRetriesOnFailure(noOfRetriesOnFailure);

processBatchInfo.parmCaption("Biller Grid Specification");

header.addTask(\_ProcessIncrement);

header.parmAlerts(NoYes::No, NoYes::Yes, NoYes::No, NoYes::Yes, NoYes::Yes);

header.save();

info(**strFmt**("'%1' batch has been scheduled.", sParmCaption));