

# Outputting the Results

---



**Michael Hoffman**

DEVELOPER AND PLURALSIGHT AUTHOR

@mhi\_inc [github.com/michaelhoffmantech](https://github.com/michaelhoffmantech)



```
<changeSet id="010220180000001" author="system">
  <createTable tableName="patient">
    <column name="patient_id" type="bigint" autoIncrement="${autoIncrement}">
      <constraints primaryKey="true" nullable="false"/>
    </column>

    ...

  </createTable>
</changeSet>
```

010220180000000\_create\_patient\_objects.xml

New file in resource folder resources/config/liquibase/changelog



# Demo



**Demo 19 – Adding the patient database schema to the Liquibase change log**



JPItemWriter

**Leverages EntityManager  
Transactions**



```
@Bean
@StepScope
public JpaItemWriter<PatientEntity> writer() {

    JpaItemWriter<PatientEntity> writer = new JpaItemWriter<>();
    writer.setEntityManagerFactory(batchEntityManagerFactory);
    return writer;

}
```

## BatchJobConfiguration.java

Package `com.pluralsight.springbatch.patientbatchloader.config`



# Demo



**Demo 20 – Implementing the JPA item writer**



# BatchJobConfigurationTest

```
@Test
public void testWriter() throws Exception {

    PatientEntity entity = new PatientEntity( ... );

    StepExecution execution = MetadataInstanceFactory.createStepExecution();

    StepScopeTestUtils.doInStepScope(execution, () -> {

        writer.write(Arrays.asList(entity));

        return null;

    });

    assertTrue(patientRepository.findAll().size() > 0);

}
```



# Demo



## Demo 21 – Testing the item writer





# Demo



**Demo 22 - Executing the job with the item writer**



# Summary



Added a Liquibase change log

Implemented a JPA item writer

Tested the item writer



Thank you!

