NiFi Custom Processor Full Student Guide

Step 1: Setup Maven in the PC

1. Download Apache Maven from: https://maven.apache.org/download.cgi
2. Extract the zip file to a directory like: C:\maven
3. Set Environment Variables:
   - MAVEN_HOME = C:\maven
   - Add %MAVEN_HOME%\bin to the PATH variable
4. Open Command Prompt and verify installation using:
   mvn -version

Expected output includes Maven version and Java version.

Step 2: Create Maven Project with Structure

1. Open terminal and run:
   mvn archetype:generate -DgroupId=com.example.nifi ^
   -DartifactId=nifi-custom-employee-processor ^
   -DarchetypeArtifactId=maven-archetype-quickstart ^
   -DinteractiveMode=false

2. Replace the generated pom.xml file with the provided one below.
3. Your final project structure should look like this:

```
nifi-custom-employee-processor/
 pom.xml
 src/
    main/
       java/
          com/
              example/
                  nifi/
                      UpdateEmployeeSalary.java
       resources/
           META-INF/
               services/
                   org.apache.nifi.processor.Processor
    test/
        java/
            (optional test classes)
```

Step 3: Add Java Code, XML File and Processor File

Create the following file with the given code:

File: src/main/java/com/example/nifi/UpdateEmployeeSalary.java

```
package com.example.nifi;

import com.fasterxml.jackson.databind.ObjectMapper;
```

NiFi Custom Processor Full Student Guide

```
import com.fasterxml.jackson.databind.node.ObjectNode;
import org.apache.nifi.annotation.documentation.CapabilityDescription;
import org.apache.nifi.annotation.documentation.Tags;
import org.apache.nifi.flowfile.FlowFile;
import org.apache.nifi.processor.*;
import org.apache.nifi.processor.exception.ProcessException;
import org.apache.nifi.processor.io.StreamCallback;

import java.io.*;
import java.util.Collections;
import java.util.HashSet;
import java.util.Set;

@Tags({"employee", "salary", "json", "update"})
@CapabilityDescription("Updates salary based on employee age.")
public class UpdateEmployeeSalary extends AbstractProcessor {

    public static final Relationship SUCCESS = new Relationship.Builder()
            .name("success")
            .description("All successful FlowFiles")
            .build();

    private Set<Relationship> relationships;

    @Override
    protected void init(final ProcessorInitializationContext context) {
        final Set<Relationship> rels = new HashSet<>();
        rels.add(SUCCESS);
        relationships = Collections.unmodifiableSet(rels);
    }

    @Override
    public Set<Relationship> getRelationships() {
        return relationships;
    }

    @Override
    public void onTrigger(ProcessContext context, ProcessSession session) throws ProcessException {
        FlowFile flowFile = session.get();
        if (flowFile == null) return;

        FlowFile updatedFlowFile = session.write(flowFile, new StreamCallback() {
            @Override
            public void process(InputStream in, OutputStream out) throws IOException {
                BufferedReader reader = new BufferedReader(new InputStreamReader(in));
                BufferedWriter writer = new BufferedWriter(new OutputStreamWriter(out));
                ObjectMapper mapper = new ObjectMapper();
                String line;
                while ((line = reader.readLine()) != null) {
                    ObjectNode node = (ObjectNode) mapper.readTree(line);
                    int age = node.get("age").asInt();
```

```
                    double salary = node.get("salary").asDouble();
                    double multiplier = (age < 25) ? 1.05 : (age <= 35 ? 1.10 : 1.15);
                    node.put("salary", salary * multiplier);
                    writer.write(mapper.writeValueAsString(node));
                    writer.newLine();
                }
                writer.flush();
            }
        });

        session.transfer(updatedFlowFile, SUCCESS);
    }
}
```

File: pom.xml (in project root)

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <groupId>com.example.nifi</groupId>
    <artifactId>nifi-custom-employee-processor</artifactId>
    <version>1.0.6</version>
    <packaging>nar</packaging>

    <properties>
        <nifi.version>1.24.0</nifi.version>
        <maven.compiler.source>11</maven.compiler.source>
        <maven.compiler.target>11</maven.compiler.target>
    </properties>

    <dependencies>
        <dependency>
            <groupId>org.apache.nifi</groupId>
            <artifactId>nifi-api</artifactId>
            <version>${nifi.version}</version>
            <scope>provided</scope>
        </dependency>
        <dependency>
            <groupId>org.apache.nifi</groupId>
            <artifactId>nifi-utils</artifactId>
            <version>${nifi.version}</version>
            <scope>provided</scope>
        </dependency>
        <dependency>
            <groupId>com.fasterxml.jackson.core</groupId>
            <artifactId>jackson-databind</artifactId>
            <version>2.16.1</version>
        </dependency>
    </dependencies>
```

```
    <build>
        <plugins>
            <plugin>
                <groupId>org.apache.nifi</groupId>
                <artifactId>nifi-nar-maven-plugin</artifactId>
                <version>1.5.1</version>
                <extensions>true</extensions>
                <configuration>
                    <group>com.example.nifi</group>
                </configuration>
            </plugin>
        </plugins>
    </build>
</project>
```

File: src/main/resources/META-INF/services/org.apache.nifi.processor.Processor

Content:

com.example.nifi.UpdateEmployeeSalary

Step 4: Build the Project and Check for NAR

1. Run the following command from the project root:
   mvn clean package

2. After a successful build, the following file should be generated:
   target/nifi-custom-employee-processor-1.0.6.nar

3. If the file doesn't appear, check the build logs for errors and ensure all directory/file paths are correct.

Step 5: Add NAR File to NiFi 2.3.0

1. Stop NiFi if it is currently running:
   nifi.bat stop

2. Copy the generated .nar file to the extensions directory of NiFi:
   C:\nifi-2.3.0\extensions\

3. Restart NiFi:
   nifi.bat start

Step 6: Start NiFi and Check for UpdateEmployeeSalary Processor

1. Open browser and go to: http://localhost:8080/nifi
2. Click the "+" button to add a new processor.
3. In the search box, type: UpdateEmployeeSalary
4. You should see the custom processor if the NAR was loaded correctly.

NiFi Custom Processor Full Student Guide

Step 7: Create NiFi Flow and Test the Processor

1. Drag and drop the following processors in sequence:
   - GenerateFlowFile
   - UpdateEmployeeSalary
   - PutFile

2. Configure GenerateFlowFile:
   - Custom Text:
     {"name":"John","age":30,"salary":50000}
   - Set 'Run Schedule' to 5 sec or manual trigger

3. Configure UpdateEmployeeSalary (no config needed)

4. Configure PutFile:
   - Directory: C:\nifi-output (create manually)

5. Start the flow and observe the output in the C:\nifi-output directory.
   - You should see the updated salary based on age logic.