

Lesson 5 Gosu Templates

This exercise requires that you use **TrainingApp** and Guidewire Studio. Start **Guidewire Studio for TrainingApp**. Start the server as **Debug** 'Server'.

Exercise 1: Generate a String message payload



Exercise

For existing contacts with a modified tax ID and for newly created contacts of the type ABPerson, **TrainingApp** must determine if the given contact has been involved with a previous act of insurance fraud. The external system that checks for fraud requires that there is specific format for message payloads for various types of entities.

5.1.1 Requirements

Spec 1 The external system requires the following format for a String message payload for a contact of the type ABPerson:

- The names in each name/value pair must be taxID and fullName.
- A comma must separate the name and value.
- A semi-colon must delimit each name/value pair.
- For example: taxID,999-99-9999;fullName,John Doe

Spec 2 Create a gosutemplate package with the fully qualified name of: si.ta.gosutemplate

Spec 3 Create a gosutemplate called FraudTemplate in the gosutemplate package.

5.1.2 Tasks

1. Create a gosutemplates package.
2. Create a Gosu Template.
3. Deploy code changes.
4. Perform verification steps.

5.1.3 Verification steps

1. Generate debug console output using **Gosu Scratchpad**.
 - a) In **Studio**, open **Gosu Scratchpad** by clicking **Tools ➔ Gosu Scratchpad**.
 - b) Write code that will test the solution:
 - Create a query that references contact William Andy whose **PublicID** is **ab:5**.
 - Make sure only one result is retrieved, otherwise throw an exception.
 - Create a new payload variable the generates String output utilizing the Gosu Template.

- Output the payload to the debug console.
2. Verify that the output in the debug console has the two name/value pairs in the correct format.

```
taxID,123-45-6793;fullName,William Andy
```



Solution 1: Generate a String message payload

1. Create a gosutemplates package

- Right-click on the gsrc folder and select **New ➔ Package**.
- Enter **si.ta.gosutemplate** as the new package name.

2. Create a Gosu Template

- Right-click on the **gosutemplate** package and select **New ➔ Gosu Template**.
- Enter **FraudTemplate** as the new Gosu Template name.

```
<%@ params( anABPerson: ABPerson ) %>
taxID,${anABPerson.TaxID};fullName,${anABPerson.FullName}
```

3. Deploy code changes.

- From the **Studio** menu, **Restart** the server.

4. Perform verification steps.

```
uses gw.api.database.Query
uses gw.api.database.Relop

var queryObj = Query.make(ABPerson)
queryObj.compare(ABPerson#PublicID, Relop.Equals, "ab:5")
var targetPerson = queryObj.select().AtMostOneRow

var payload = si.ta.gosutemplate.FraudTemplate.renderToString(targetPerson)
print(payload)
```