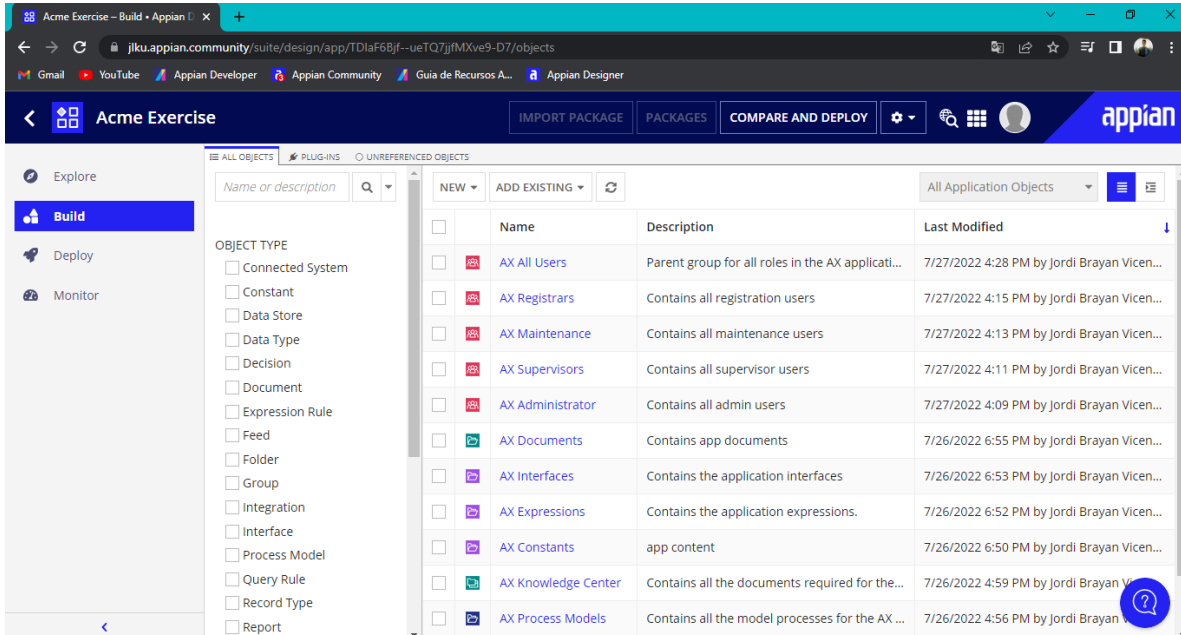


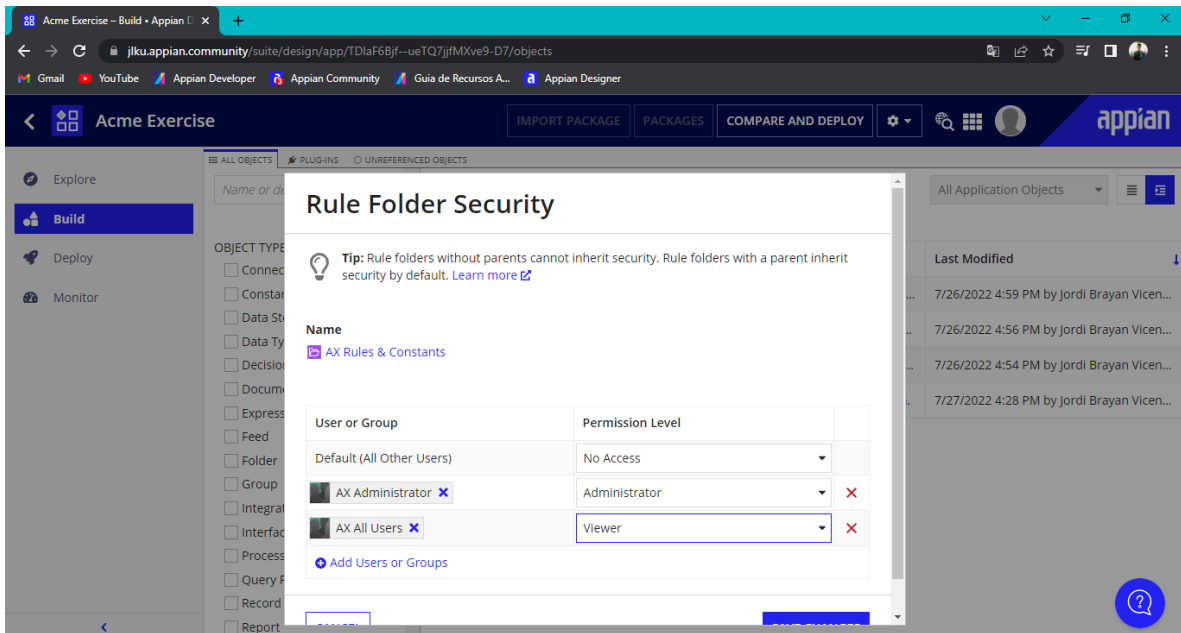
Creación de grupos



The screenshot shows the Appian Designer interface for 'Acme Exercise'. The 'Build' menu is active, and the 'All Application Objects' list is displayed. The list contains the following objects:

Name	Description	Last Modified
AX All Users	Parent group for all roles in the AX applicati...	7/27/2022 4:28 PM by Jordi Brayan Vicen...
AX Registrars	Contains all registration users	7/27/2022 4:15 PM by Jordi Brayan Vicen...
AX Maintenance	Contains all maintenance users	7/27/2022 4:13 PM by Jordi Brayan Vicen...
AX Supervisors	Contains all supervisor users	7/27/2022 4:11 PM by Jordi Brayan Vicen...
AX Administrator	Contains all admin users	7/27/2022 4:09 PM by Jordi Brayan Vicen...
AX Documents	Contains app documents	7/26/2022 6:55 PM by Jordi Brayan Vicen...
AX Interfaces	Contains the application interfaces	7/26/2022 6:53 PM by Jordi Brayan Vicen...
AX Expressions	Contains the application expressions.	7/26/2022 6:52 PM by Jordi Brayan Vicen...
AX Constants	app content	7/26/2022 6:50 PM by Jordi Brayan Vicen...
AX Knowledge Center	Contains all the documents required for the...	7/26/2022 4:59 PM by Jordi Brayan V...
AX Process Models	Contains all the model processes for the AX ...	7/26/2022 4:56 PM by Jordi Brayan V...

Seguridad para AX_Rules & Constants



The screenshot shows the 'Rule Folder Security' dialog box in Appian Designer. The dialog is titled 'Rule Folder Security' and contains the following information:

- Tip:** Rule folders without parents cannot inherit security. Rule folders with a parent inherit security by default. [Learn more](#)
- Name:** AX Rules & Constants
- User or Group:** AX Administrator, AX All Users
- Permission Level:** No Access, Administrator, Viewer

User or Group	Permission Level
Default (All Other Users)	No Access
AX Administrator	Administrator
AX All Users	Viewer

Seguridad para AX_Process_Models

The screenshot shows the Appian Designer interface for configuring security on a Process Model Folder. The main window is titled "Process Model Folder Security". A tip indicates that security is not inherited from the parent and must be set individually. The configuration table shows the following settings:

User or Group	Permission Level
Default (All Other Users)	No Access
AX Administrator	Administrator
AX All Users	Viewer

The interface also includes a sidebar with navigation options (Explore, Build, Deploy, Monitor) and a right-hand pane showing a list of application objects with their last modified dates.

Seguridad para AX_Knowledge center

The screenshot shows the Appian Designer interface for configuring security on a Knowledge Center. The main window is titled "Knowledge Center Security". A tip indicates that security is automatically inherited by documents and folders within it. The configuration table shows the following settings:

User or Group	Permission Level
Default (All Other Users)	No Access
AX Administrator	Administrator
AX All Users	Editor

The interface also includes a sidebar with navigation options (Explore, Build, Deploy, Monitor) and a right-hand pane showing a list of application objects with their last modified dates.

Seguridad para la aplicación Acme Exercise

The screenshot shows the Appian Designer interface with the 'Application Security' dialog box open. The dialog is titled 'Application Security' and contains a tip: 'Tip: Basic users must have at least viewer rights to view a published application's feeds and actions at runtime. Learn more'. Below the tip, the 'Name' is set to 'Acme Exercise'. A table lists the permissions for different users or groups:

User or Group	Permission Level	
Default (All Other Users)	No Access	
AX Administrator	Administrator	X
AX All Users	Viewer	X

There is an 'Add Users or Groups' button at the bottom of the table. The background shows the Appian Designer sidebar with 'Users' selected.

Regla AX_L2Ex1

The screenshot shows the Appian Designer interface for the 'AX_L2Ex1' rule. The rule is configured with the following expression:

```
1 user(loggedInUser(), "firstName")
2 & " created on " &
3 user(loggedInUser(), "created")
```

The 'Test Inputs' section is empty. The 'Test Output' section shows the following result:

Time 5 ms (View Performance) Type Text
Value Formatted Raw Expression
"Jordi Brayan created on 7/25/2022 6:53 PM GMT+00:00" (Text)

At the bottom, there is a note: 'Place cursor on function, rule, or constant to display help'.

Regla AX_L2Challenge

The screenshot displays the Appian Designer interface for the rule 'AX_L2Challenge'. The left pane shows the rule's configuration with the following code:

```
1 text(  
2 datevalue("12/12/21"),  
3 "dd mmmm, yyyy"  
4 )
```

The right pane shows the 'Test Inputs' section with a 'TEST RULE' button. Below it, the 'Test Output' section displays the following information:

Time 2 ms ([View Performance](#)) **Type** Text
Value Formatted Raw Expression
"12 December, 2021" (Text)

Below the code editor, the documentation for the `text(value, format)` function is visible:

text(value, format)
Converts number, date, time, or datetime timestamp value to a string.
Returns: Text
value (Any Type): The value to convert to a Text.
format (Text): The output format string, supporting "date/time" format, "positive" format, "positive:negative" format or "positive:negative:zero" format. "date/time" format can only be for the positive. For example, `text(10.25, "$00.00")` would return \$10.25, `text(-10.25, "$00.00:#00.00")` would return #10.25, `text(0, "$00.00:#00.00;#00.0")` would return #00.0 and `text(38353, "mmm/ddd/aaaa")` would return Jan/12/21. Please refer to the

At the bottom right, there is a notification: "Open the Learning Center to select a tutorial or video" with a question mark icon.

Regla AX_L4Ex1

The screenshot displays the Appian Designer interface for the rule 'AX_L4Ex1'. The left pane shows the rule's configuration with the following code:

```
1 if(  
2 400 > 200,  
3 "Sell",  
4 "Do not sell"  
5 )
```

The right pane shows the 'Test Inputs' section with a 'TEST RULE' button. Below it, the 'Test Output' section displays the following information:

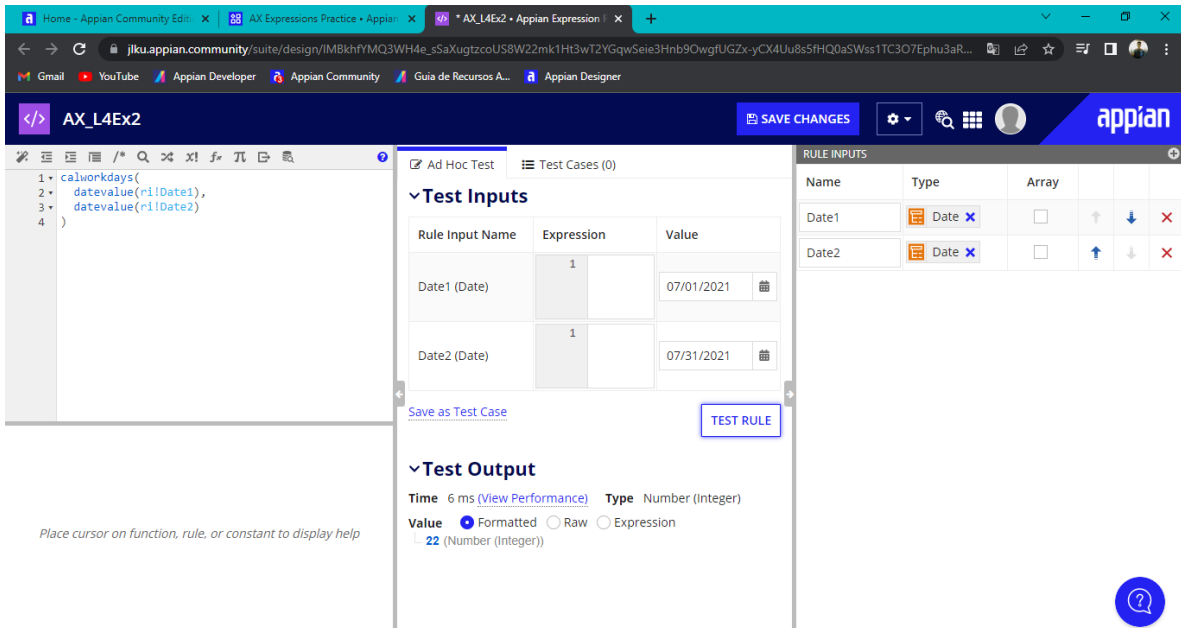
Time 1 ms ([View Performance](#)) **Type** Text
Value Formatted Raw Expression
"Sell" (Text)

Below the code editor, the documentation for the `if(condition, valueIfTrue, valueIfFalse)` function is visible:

if(condition, valueIfTrue, valueIfFalse)
Returns valueIfTrue if condition returns true; returns valueIfFalse otherwise.
Returns: Any Type
condition (Boolean): A test that determines whether valueIfTrue or valueIfFalse will be returned.
valueIfTrue (Any Type): The value to be returned if condition returns true.
valueIfFalse (Any Type): The value to be returned if condition returns false.

At the bottom right, there is a question mark icon.

Regla AX_L4Ex2



AX_L4Ex2

```
1 calworkdays(  
2   datevalue(r!!Date1),  
3   datevalue(r!!Date2)  
4 )
```

Ad Hoc Test | Test Cases (0)

Test Inputs

Rule Input Name	Expression	Value
Date1 (Date)	1	07/01/2021
Date2 (Date)	1	07/31/2021

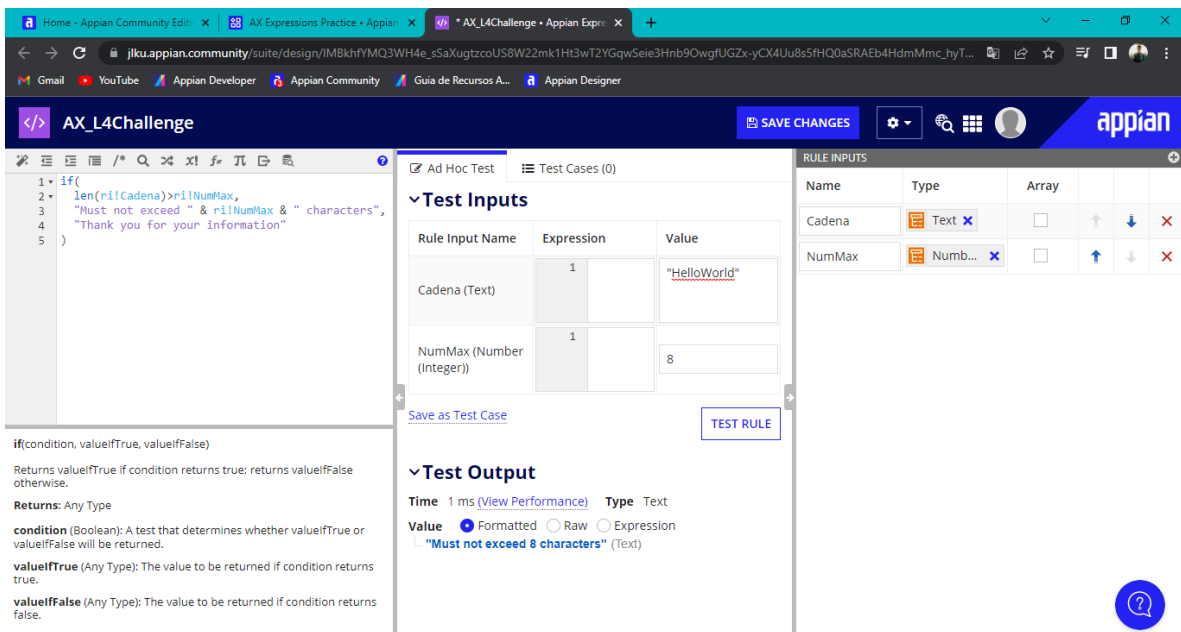
Save as Test Case TEST RULE

Test Output

Time 6 ms (View Performance) Type Number (Integer)
Value Formatted Raw Expression
22 (Number (Integer))

Place cursor on function, rule, or constant to display help

Regla AX_L4Challenge



AX_L4Challenge

```
1 if(  
2   len(r!!Cadena)>r!!NumMax,  
3   "Must not exceed " & r!!NumMax & " characters",  
4   "Thank you for your information"  
5 )
```

Ad Hoc Test | Test Cases (0)

Test Inputs

Rule Input Name	Expression	Value
Cadena (Text)	1	"HelloWorld"
NumMax (Number (Integer))	1	8

Save as Test Case TEST RULE

Test Output

Time 1 ms (View Performance) Type Text
Value Formatted Raw Expression
"Must not exceed 8 characters" (Text)

if(condition, valueIfTrue, valueIfFalse)
Returns valueIfTrue if condition returns true; returns valueIfFalse otherwise.
Returns: Any Type
condition (Boolean): A test that determines whether valueIfTrue or valueIfFalse will be returned.
valueIfTrue (Any Type): The value to be returned if condition returns true.
valueIfFalse (Any Type): The value to be returned if condition returns false.

Regla AX_L5Ex1

The screenshot shows the Appian Designer interface for rule AX_L5Ex1. The rule is configured as an Ad Hoc Test. The code in the left pane is:

```
1 if(  
2   r1|Vfinal>(r1|Vinicial+50000),  
3   "Run full diagnostics",  
4   "Return to Inventory"  
5 )  
6
```

The Test Inputs table is as follows:

Rule Input Name	Expression	Value
Vinicial (Number (Integer))	1	5000
Vfinal (Number (Integer))	1	70000

The Test Output section shows a time of 1 ms and a value of "Run full diagnostics" (Text).

Documentation for the `if` function is provided below the code editor:

if(condition, valueIfTrue, valueIfFalse)
Returns valueIfTrue if condition returns true; returns valueIfFalse otherwise.

Returns: Any Type

condition (Boolean): A test that determines whether valueIfTrue or valueIfFalse will be returned.

valueIfTrue (Any Type): The value to be returned if condition returns true.

valueIfFalse (Any Type): The value to be returned if condition returns false.

Regla AX_L5Ex2

The screenshot shows the Appian Designer interface for rule AX_L5Ex2. The rule is configured as an Ad Hoc Test. The code in the left pane is:

```
1 alocalVariables(  
2   local|suma: r1|first+r1|second,  
3   "The new total is " & local|suma  
4 )
```

The Test Inputs table is as follows:

Rule Input Name	Expression	Value
first (Number (Integer))	1	5
second (Number (Integer))	1	6

The Test Output section shows a time of 1 ms and a value of "The new total is 11" (Text).

Documentation for the `alocalVariables` function is provided below the code editor:

alocalVariables localVar1, localVarN, expression
Lets you define one or more local variables for use within an expression. When used within an interface, the value of each variable can be refreshed under a variety of conditions, configured using `alrefreshVariable()`. When used outside of an interface, all refresh properties configured using `alrefreshVariable()` are ignored.

localVar1 (Any Type): The local variable to use when evaluating the given expression. Use the 'local' domain to define and reference individual variables. By default, a local variable will automatically update when any variables it references are changed. To change the way variables are updated, use the `alrefreshVariable()` function. Variables can be refreshed under the following conditions: after

Regla AX_L5Challenge

The screenshot shows the Appian Designer interface for the rule 'AX_L5Challenge'. The left pane contains the rule's code:

```
1 if(  
2   isnull(r1|valor),  
3   "Null value",  
4   "The value is " & r1|valor  
5 )
```

The right pane shows the 'Test Inputs' section with a table:

Rule Input Name	Expression	Value
valor (Number (Integer))	1	

Below the table are buttons for 'Save as Test Case' and 'TEST RULE'. The 'Test Output' section shows:

Time 1 ms (View Performance) Type Text
Value Formatted Raw Expression
└─ "Null value" (Text)

At the bottom right, there is a notification: 'Open the Learning Center to select a tutorial or video' with a question mark icon.

Regla AX_L6Ex1

The screenshot shows the Appian Designer interface for the rule 'AX_L6Ex1'. The left pane contains the rule's code:

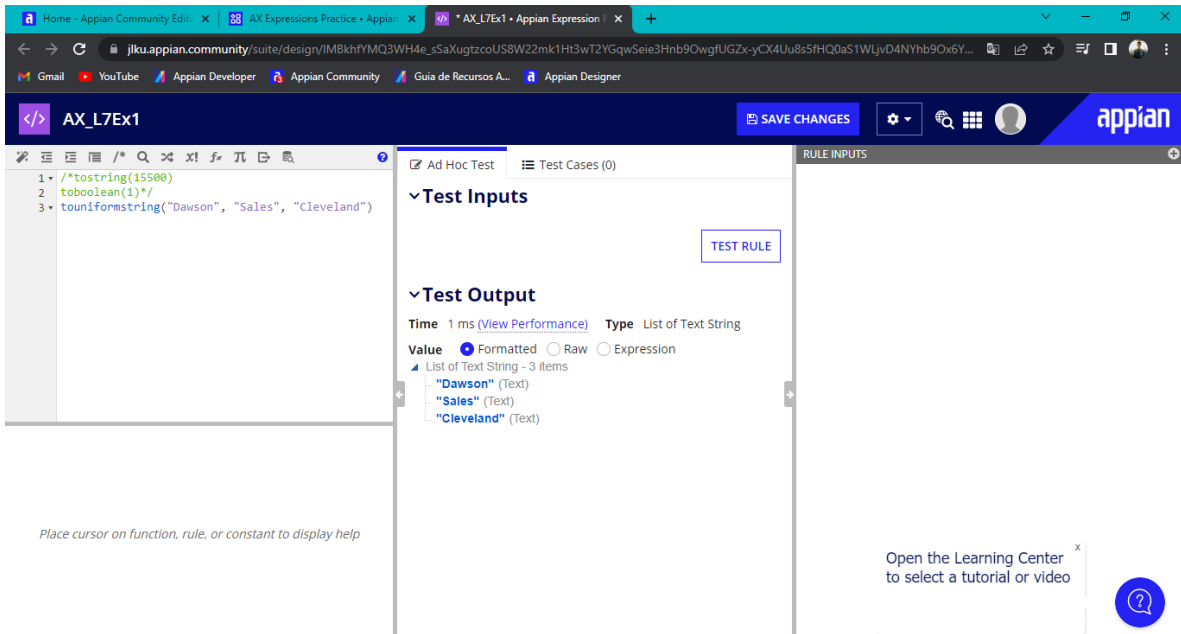
```
1 index(  
2 {  
3   color: "Red",  
4   make: "Toyota",  
5   model: "Prius",  
6   mileage: 15000  
7 },  
8 "model"  
9 )
```

The right pane shows the 'Test Inputs' section, which is currently empty. Below it, the 'Test Output' section shows:

Time 1 ms (View Performance) Type Any Type (Text)
Value Formatted Raw Expression
└─ "Prius" (Text)

At the bottom left, there is a note: 'Place cursor on function, rule, or constant to display help'. At the bottom right, there is a question mark icon.

Regla AX_L7Ex1



Home - Appian Community Edition | AX Expressions Practice - Appian | AX_L7Ex1 - Appian Expression

AX_L7Ex1

```
1 /*tostring(15500)
2 toboolean(1)*/
3 touniformstring("Dawson", "Sales", "Cleveland")
```

Ad Hoc Test | Test Cases (0)

TEST INPUTS

TEST OUTPUT

Time 1 ms (View Performance) Type List of Text String

Value Formatted Raw Expression

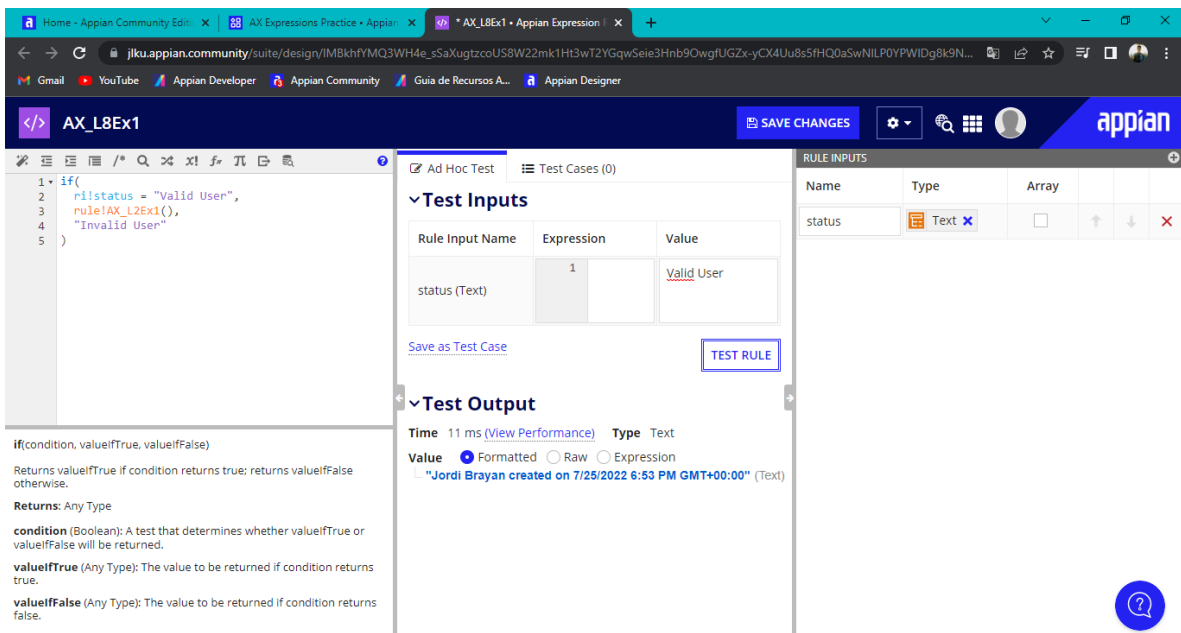
List of Text String - 3 Items

- "Dawson" (Text)
- "Sales" (Text)
- "Cleveland" (Text)

Place cursor on function, rule, or constant to display help

Open the Learning Center to select a tutorial or video

Regla AX_L8Ex1



Home - Appian Community Edition | AX Expressions Practice - Appian | AX_L8Ex1 - Appian Expression

AX_L8Ex1

```
1 if(
2   r1!status = "Valid User",
3   rule!AX_L2Ex1(),
4   "Invalid User"
5 )
```

Ad Hoc Test | Test Cases (0)

TEST INPUTS

Rule Input Name	Expression	Value
status (Text)	1	Valid User

Save as Test Case

TEST OUTPUT

Time 11 ms (View Performance) Type Text

Value Formatted Raw Expression

- "Jordi Brayan created on 7/25/2022 6:53 PM GMT+00:00" (Text)

if(condition, valueIfTrue, valueIfFalse)

Returns valueIfTrue if condition returns true; returns valueIfFalse otherwise.

Returns: Any Type

condition (Boolean): A test that determines whether valueIfTrue or valueIfFalse will be returned.

valueIfTrue (Any Type): The value to be returned if condition returns true.

valueIfFalse (Any Type): The value to be returned if condition returns false.

Regla AX_L8Ex2

The screenshot shows the Appian Designer interface for rule AX_L8Ex2. The rule is an 'if' statement with the following code:

```
1 if(  
2   rlmilestone = "On Schedule",  
3   cons!AX_Exercise8,  
4   "Revise Deadline"  
5 )
```

The 'Test Inputs' section shows a table with one input:

Rule Input Name	Expression	Value
milestone (Text)	1	On Schedule

The 'Test Output' section shows the following results:

Time: 1 ms (View Performance) Type: Date
Value: Formatted (selected) Raw Expression
12/1/2021 (Date)

Below the code, there is a help section for the 'if' function:

if(condition, valueIfTrue, valueIfFalse)
Returns valueIfTrue if condition returns true; returns valueIfFalse otherwise.
Returns: Any Type
condition (Boolean): A test that determines whether valueIfTrue or valueIfFalse will be returned.
valueIfTrue (Any Type): The value to be returned if condition returns true.
valueIfFalse (Any Type): The value to be returned if condition returns false.

Regla AX_L9Ex1

The screenshot shows the Appian Designer interface for rule AX_L9Ex1. The rule is a concatenation of strings and a rule call:

```
1 "Ticket #" & rlticketID &  
2 " created by " & user(  
3   loggedInUser(),  
4   "firstName"  
5 )  
6
```

The 'Test Inputs' section shows a table with one input:

Rule Input Name	Expression	Value
ticketID (Number (Integer))	1	7

The 'Test Output' section shows the following results:

Time: 5 ms (View Performance) Type: Text
Value: Formatted (selected) Raw Expression
"Ticket #7 created by Jordi Brayán" (Text)

Below the code, there is a help section:

Place cursor on function, rule, or constant to display help

Creación de las constantes para todos los grupos

All Application Objects > AX Rules & Constants >
AX Constants

Name	Description	Last Modified
AX_DOCUMENTS_FOLDER_POINTER	Points to the AX Documents folder	7/28/2022 1:16 AM by Jordi Brayan Vi
AX_MAINTENANCE_POINTER	A pointer to the AX application Maintena...	7/28/2022 1:12 AM by Jordi Brayan Vi
AX_REGISTRARS_POINTER	A pointer to the AX application Registrars...	7/28/2022 1:11 AM by Jordi Brayan Vi
AX_SUPERVISORS_POINTER	A pointer to the AX application Superviso...	7/28/2022 1:09 AM by Jordi Brayan Vi
AX_ALL_USERS_POINTER	A pointer to the AX application All Users ...	7/28/2022 1:08 AM by Jordi Brayan Vi
AX_ADMINISTRATORS_POINTER	A pointer to the AX application administr...	7/28/2022 1:02 AM by Jordi Brayan Vi
AX_Exercise8	Constant for use in Exercise 8 of the Expr...	7/28/2022 12:33 AM by Jordi Brayan Vi

AX_DisplayUser

AX_DisplayUser

Rule Input Name	Expression	Value
user (Text)	1 "jbvicente@minsait.com"	"jbvicente@minsait.com"

TEST RULE

Time 7 ms (View Performance) Type Text
Value Formatted Raw Expression
"Jordi Brayan Vicente Moreno" (Text)