

Article

[Yuri Marx](#) · Mar 2, 2021 4m read

Day 5: Developing with InterSystems Objects and SQL

I finished my participation in the Developing with InterSystems Objects and SQL with Joel Solon. The course is very nice and I will share with you some tips I got during this the training final day. Tips presented in the day 5:

1. %SYSTEM package has general utility classes. Use \$system.class.Help() to list the utilities. The main %SYSTEM classes are Encryption (industry encryption methods), OBJ to manage classes and objects, Security for security methods and SQL to use SQL language.
2. IRIS allows validation to the properties and arguments using MINVAL and MAXVAL to integer, FORMAT to date and time, SCALE to decimal, MINLEN, MAXLEN, TRUNCATE, VALUelist (restrict the input to a list of values) and PATTERN to string. Example: Method Print(name As %String(MAXLEN=50)).
3. You can create new data type validations and conversions extending a datatype like %String and using DisplayToLogical() and LogicalToDisplay() to implement new behaviors. It is possible to change the datatype parameter like the PATTERN parameter for change the validation of a string.
4. IRIS uses -> operator to do implicit joins, example: select Country->States->County from Simple.Places. This joins 3 related tables.
5. To create a reference between classes/tables use ForeignKey. Example: ForeignKey CountryFK(Country) References Simple.Country() [OnDelete = setnull]. There are the options setnull to set null to the other side table/class, cascade to propagate the deletion, noaction to no action and setdefault to configure a default value to the reference.
6. Use %ConstructClone() to do a clone of a object and its references.
7. Use SqlComputeCode to the calculated fields. Example: (from IRIS documentation)

```
Property FullName As %String [ SqlComputeCode = {set {*}={FirstName}_ " "_{LastName}}, SqlComputed ];
```

8. It is possible to automate tests using %UnitTest framework. Use %UnitTest.TestCase for defining test cases and %UnitTest.Manager for running a set of Test Cases. To see the results go to System Explorer > Tools > UnitTest Portal. See <https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=TUNT>.
9. IRIS supports 2 type collections:
 1. List: key based on position starting in 1.
 1. Declaration: Property States as list of Simple.State.
 2. Get: country.States.GetAt(1)
 3. Insert: country.States.Insert(florida)
 4. Remove: country.States.RemoveAt(1)
 5. Count: country.States.Count()
 6. Clear: country.States.Clear()
 7. Next: country.States.GetNext(1) - to get item 2
 8. Loop: for i=1:1:country.States.Count() { write !, country.States.GetAt(i)
 2. Array: key based on associated value.
 1. Declaration: Property States as array of Simple.State.
 2. Get: country.States.GetAt("FL")
 3. Insert: country.States.SetAt(florida, "FL")
 4. Remove: country.States.RemoveAt("FL")
 5. Count: country.States.Count()
 6. Clear: country.States.Clear()
 7. Next: country.States.GetNext("FL") - to get item GA
 8. Loop: set state = "" for { set val = country.States.GetNext(.state) quit: (state = "") write !, val }
10. To create relationships between classes, we have:

1. Parent/Children: Relationship States as Simple.State [cardinality = children, Inverse = Country].
The otherside: Relationship Country as Simple.Country [cardinality = parent, Inverse = Country].
2. One/Many: Relationship States as Simple.State [cardinality = many, Inverse = Country]. The
otherside: Relationship Country as Simple.Country [cardinality = one, Inverse = Country].
11. One-To-One and Many-to-Many are not supported. One-To-One may be simulated and Many-To-Many can
be made using 2 Many-to-One to a relationship table.
12. Stream is object that can contain amount of data larger than limit of regular properties, may store characters
(%Stream.GlobalCharacter) or binary (%Stream.GlobalBinary) data. Use the following methods to
read/write data:
 1. Write(text): write in the end of line;
 2. WriteLine(text): write to a new line;
 3. Clear(): clear the content of the stream;
 4. CopyFrom(stream): copy from a stream and overwrite the current content;
 5. Read(length): read characters limited to the length;
 6. ReadLine(length): read characters in the current line limited to the length;
 7. Rewind(): retorna para o início do stream;
 8. MoveToEnd(): move to end of stream.
13. You can use triggers to do something before or after a CRUD event. Example: Trigger T1 [Event = INSERT,
Time = AFTER, Order = 1, Foreach = row/object] { write your code }.
14. The IRIS has some functions frequently used:
 1. \$EXTRACT: get a substring -
<https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=RCOSfextract>
 2. \$FIND: get the position of a value passed as argument -
<https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=RCOSffind>
 3. \$LENGTH: get the size of a string or substring
- <https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=RCOSflength>
 4. \$LIST: return/create a list of elements -
<https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=RCOSflist>
 5. \$LISTDATA: return the element in the list by position -
<https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=RCOSflistdata>
 6. \$LISTFIND: return the index of a element in the list -
<https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=RCOSflistfind>
 7. \$LISTGET - get an element in the list -
<https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=RCOSflistget>
 8. \$LISTLENGTH - get total of elements in the list -
<https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=RCOSflistlength>
 9. \$PIECE - return or replace a substring using a delimiter -
<https://docs.intersystems.com/irislatest/csp/docbook/DocBook.UI.Page.cls?KEY=RCOSfpiece>

The course was very nice and has a rich set of practical exercises. The course is important to preparation to IRIS developer certification too. See the course agenda in <https://www.intersystems.com/support-learning/learning-services/classroom-learning/?course-view=course-schedule>.

[#Tips & Tricks](#) [#InterSystems IRIS](#)

Source URL: <https://community.intersystems.com/post/day-5-developing-intersystems-objects-and-sql>