Article
Yuri Marx · Feb 25, 2021 3m read

Day 3: Developing with InterSystems Objects and SQL

I'm participating 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 the training. Tips presented in the day 3:

1. You can see your class catalog using %Dictionary classes and see your sql objects into INFORMATIONSCHEMA table.
2. Is possible use SQL inside your ObjectScript methods using Dynamic SQL or Embedded SQL.
3. You can pass parameters into Dynamic SQL string using ? (eg.: where country = ?) and pass parameters to Embedded SQL using colon (ed.: where country = :variable).
4. Dynamic SQL Sample (from Intersystems documentation):

```
SET tStatement = ##class(%SQL.Statement).%New(,"Sample")
  SET myquery = 3
  SET myquery(1) = "SELECT TOP ? Name,DOB,Home_State"
  SET myquery(2) = "FROM Person"
  SET myquery(3) = "WHERE Age > 60 AND Age < 65"
  SET qStatus = tStatement.%Prepare(.myquery)
   IF qStatus'=1 {WRITE "%Prepare failed:" DO $System.Status.DisplayError(qStat
us) QUIT}
  DO tStatement.%Display()
  WRITE !,"End of %Prepare display"
```

5. Embedded SQL Sample (from Intersystems documentation):

```
#SQLCompile Select=Display
  &sql(SELECT DOB INTO :a FROM Sample.Person)
     IF SQLCODE<0 {WRITE "SQLCODE error ",SQLCODE," ",%msg  QUIT}
     ELSEIF SQLCODE=100 {WRITE "Query returns no results"  QUIT}
  WRITE "1st date of birth is ",a,!
  DO $SYSTEM.SQL.Util.SetOption("SelectMode",1)
  WRITE "changed select mode to: ",$SYSTEM.SQL.Util.GetOption("SelectMode"),!
  &sql(SELECT DOB INTO :b FROM Sample.Person)
  WRITE "2nd date of birth is ",b
```

6. Embedded SQL Sample - Insert:

```
&sql(INSERT INTO Sample.Person (Name, Age, Phone) VALUES (:name, :age, :phone)
```

7. If you need process data in batch use SQL, if you process a single record, use Persistent Object API.
8. You can create SQLQuery methods and if you use [SqlProc] in the method, will be created a SQL procedure in the SQL side.
9. From terminal is possible go to SQL Shell, a terminal to SQL commands, from terminal, execute do $system.SQL.Shell().
10. Persistent classes have a system generated ID, if you need ID controled by you, use IDKEY index with one or more properties. Eg: Index Key on SocialNumber [IdKey, PrimaryKey, Unique].
11. There two strategies to control concurrency when two or more process try process the same data at same time: Pessimistic and Optimistic.
12. To acquire a pessimistic control, lock the object with %OpenId(ID, 4), where 4 lock the table to exclusive access. After process ran the lock can be released.

13. To do optimistic control (indicated to web apps), create in your persistent class Parameter VERSIONPROPERTY = "Version"; Property Version as %Integer [InitialExpression = 1]. IRIS will increment property version in each instance update, allowing coordinate the order of updates, instead lock table.
14. When you have methods that update, insert or delete data, use transactions, to keep the data consistency. Example:

```
Transfer(from,to,amount)   // Transfer funds from one account to another
{
   TSTART
    &SQL(UPDATE A.Account
        SET A.Account.Balance = A.Account.Balance - :amount
        WHERE A.Account.AccountNum = :from)
    If SQLCODE TRollBack  Quit "Cannot withdraw, SQLCODE = "_SQLCODE
    &SQL(UPDATE A.Account
        SET A.Account.Balance = A.Account.Balance + :amount
        WHERE A.Account.AccountNum = :to)
    If SQLCODE TROLLBACK  QUIT "Cannot deposit, SQLCODE = "_SQLCODE
    TCOMMIT
    QUIT "Transfer succeeded"
}
```

15. InterSystems IRIS has an Architecture based in Namespaces (logical groups of databases) and Databases.
16. There two types of data to hold in the databases: for data (globals) and for code (source code - procedures).
17. You can do horizontal processing scaling to your databases using ECP - Enterprise Cache Protocol, allowing see different databases in several servers in the same namespace.
18. You can do horizontal data volume scaling (distributed database partitions) using Sharding (only IRIS), allowing partitioning data into distributed nodes (like MongoDB).
19. The maximum size to a database is 32TB.
20. To change from a namespace to another do zn "Namespace" or set $namespace = "Namespace".

PS 1: the course show in details how to do transactions control, this is very important.

Tomorrow I will post day 4 resume.

#Object Data Model #SQL #InterSystems IRIS