

Table of Contents

Function library example 2

[Programmers Reference](#), [Example](#), [IFunctionLibrary Interface](#)

Function library example

See the code snippet below for an example of a function library. For illustration purposes this code snippet contains only a handful of functions supported by the standard Date Function Library.

```
public class DateFunctionLibrary implements IFunctionLibrary
{
    supported date functions
    public static final String DAY_OF_WEEK_FUNCTION="DAY_OF_WEEK";
    public static final String CURRENT_YEAR_FUNCTION="CURRENT_YEAR";
    public DateFunctionLibrary()
    {
    }
    /**
     * @see IFunctionLibrary#getName()
     */
    @Override
    public String getName()
    {
        return "DateFunctionLibrary";
    }
    /**
     * @see IFunctionLibrary#hasFunction(String,int)
     */
    @Override
    public boolean hasFunction(String functionName, int paramNmb)
    {
        if (paramNmb==0)
        {
            return functionName.equalsIgnoreCase (CURRENT_YEAR_FUNCTION);
        }
        else if (paramNmb==1)
        {
            return functionName.equalsIgnoreCase (DAY_OF_WEEK_FUNCTION);
            return false ;
        }
    }
    /**
     * @see IFunctionLibrary#getAvailableFunctions()
     */
    @Override
    public FunctionDescription[] getAvailableFunctions()
    {
        FunctionDescription[] results= new FunctionDescription[2];
        results[0]= new FunctionDescription(CURRENT_YEAR_FUNCTION,
            "Parameters: none\nReturn: current year as 4 digit Number");
        results[1]= new FunctionDescription(DAY_OF_WEEK_FUNCTION,
            "Parameters: attribute of Date type or expression producing Date"+" \nReturn:
            day of week of the provided date as Plain Text ('Monday', 'Tuesday' etc)");
    }
}
```

```

        return results;
    }
    /**
     ** @see IFunctionLibrary#calculate(String,Object[],INodeHelper)
     **/
    public Object calculate(String functionName, Object[] params,
        INodeHelper helper)
    {
        if (params == null || params.length == 0)
        {
            if (functionName.equalsIgnoreCase(CURRENT_YEAR_FUNCTION))
            {
                Calendar cal = Calendar.getInstance();
                cal.setTime(new Date ());
                return new Integer(cal.get(Calendar.YEAR));
            }
        }
        else if (params != null && params.length == 1)
        {
            // a single parameter must be of DateHolder type
            if (functionName.equalsIgnoreCase(DAY_OF_WEEK_FUNCTION))
            {
                if (!(params[0] instanceof DateHolder))
                    return null ;
                Date date = TypeUtils.toDate((DateHolder)params[0]);
                Calendar cal = Calendar.getInstance();
                cal.setTime(date);
                SimpleDateFormat sdf = new SimpleDateFormat("E");
                return sdf.format(date);
            }
        }
        return null ;
    }
    /**
     ** @see IFunctionLibrary#toSQL(String,Collection,ISQLBuilderHelper)
     **/
    @Override
    public String toSQL(String functionName, Collection parameters,
        ISQLBuilderHelper sqlHelper)
        throws Exception
    {
        IDatabaseClientInterface dbi = sqlHelper.getDatabaseInterface();
        if (parameters == null || parameters.size() == 0)
        {
            if (dbi instanceof MySQLClientInterface)
            {
                if (functionName.equalsIgnoreCase
(CURRENT_YEAR_FUNCTION))
                {
                    return "YEAR(CURDATE())";
                }
            }
        }
    }

```

```

    }
    else if (dbi instanceof MSSQLServerClientInterface)
    {
        if (functionName.equalsIgnoreCase
(CURRENT_YEAR_FUNCTION))
        {
            return "DATEPART (yy, GETDATE())";
        }
    }
}
else if (parameters != null && parameters.size() == 1)
{
    Iterator iter = parameters.iterator();
    IArithmeticNode arNode = (IArithmeticNode) iter.next();
    // representation of the argument
    String dateArg = arNode.toSQL(sqlHelper);
    if (dateArg == null || dateArg.length() == 0)
        return ""; // the expression won't be calculated
    if (functionName.equalsIgnoreCase(DAY_OF_WEEK_FUNCTION))
    {
        if (sqlFlavour == CommonConstants.SQL_FLAVOUR_TSQL)
            return "DATENAME(dw, "+dateArg+")";
        else if (sqlFlavour == CommonConstants.SQL_FLAVOUR_MYSQL)
            return "DAYNAME("+dateArg+")";
    }
}
}
throw new UnsupportedOperationException(functionName);
}
/** @see IFunctionLibrary#getTypeClass(String, Collection) */
@Override
public Class getTypeClass(String functionName, Collection params)
{
    return Integer.class;
}
/** @see IFunctionLibrary#getRequiredFactPatterns(String, Collection) */
@Override
public HashSet getRequiredFactPatterns(String functionName, Collection params)
{
    return null;
}
}

```

From:

<http://www.awareim.com/dokuwiki/> - Documentation

Permanent link:

<http://www.awareim.com/dokuwiki/docs/3500/0500/0530>

Last update: 2023/04/06 04:45



