**API**

**API- Application Programming Interface.**

* a software intermediary that set of rules or protocols enables two applications to communicate to each other.

In NetSuite API’s ensure the application data is accessible to other programs.

**Example:** nlapiGetFieldValue(fieldtype)

***How to get to know the type***

* on existing Sales order click edit
* on URL TYPE “&xml=t” (& - ampersand) 

In this section, we can find the parameters. Copy and insert them into the syntax. The machine section represents sublists. If there are n tabs in sublists, then there will be n machine sections.

In **1.0** we use Api level structure (i.e. all the API’s are used) due to this reason it’s slow.

**2.0** is a module level architecture. Whatever module we want those modules only we can insert.

Declare a **variable** example

***“******var recordObj = nlapiCreateRecord(‘salesorder’);”***

To get the value from sales order

***“******var currencyValue = recordObj.getFieldValue(‘currency name’);”***

**Note:**

“recordObj , currencyValue” are the variable names we can write test or whatever we want to but it's better to follow camelCase where the first letter is lowercase, and each subsequent word starts with an uppercase letter.

**Record API:**

**nlapiCreateRecord (type)**

Syntax of 1.0 - nlapiCreateRecord(type);

* Var recordObj = nlapiCreateRecord(type);

Syntax 2.0 - record.create(options)

* var scriptDeployment = record.create({  
   type: record.Type.SCRIPT\_DEPLOYMENT,  
   defaultValues: {  
   script: 205 // Internal id of existing script record  
   }  
  });

**nlapiDeleteRecord(type, id)**

Syntax of 1.0 - nlapiDeleteRecord(type, id);

* Var recordObj = nlapiDeleteRecord(type, id)

Syntax of 2.0 - record.delete(options)

* // Delete a sales order  
  var salesOrderRecord = record.delete({  
   type: record.Type.SALES\_ORDER,  
   id: 88,  
  });  
    
  // Delete an instance of a custom record type with the ID customrecord\_feature  
  var featureRecord = record.delete({  
   type: 'customrecord\_feature',  
   id: 3,  
  });

**nlapiTransformRecord(type, id, transformType, transformValues)**

Syntax of 1.0 - nlapiTransformRecord

* Var recordObj = nlapiTransformRecord(type, id, transformType, transformValues)

Syntax of 2.0 **-** [record.transform(options)](https://tstdrv2268550.app.netsuite.com/app/help/helpcenter.nl?fid=section_4267258715.html)

* var objRecord = record.transform({  
   fromType: record.Type.CUSTOMER,  
   fromId: 107,  
   toType: record.Type.SALES\_ORDER,  
   isDynamic: true,

});

**nlapiSubmitRecord**

Syntax of 1.0 - nlapiSubmitRecord(record, doSourcing, ignoreMandatoryFields)

* var objRecord = nlapiSubmitRecord(record, doSourcing, ignoreMandatoryFields);

Syntax of 2.0 - Record.save(options)

* var recordId = objRecord.save({

enableSourcing: true,

ignoreMandatoryFields: true

});

**nlapiLoadRecord**

Syntax of 1.0 - nlapiLoadRecord(type, id, initializeValues)

* var objRecord = nlapiLoadRecord(type, id, initializeValues);

Syntax of 2.0 - record.load(options)

* var objRecord = record.load({

type: record.Type.SALES\_ORDER,

id: 157,

isDynamic: true,

});

**Field API’s**

**nlapiGetFieldValue(fieldtype)**

Syntax of 1.0 - nlapiGetFieldValue(fldnam)

* var objRecord = nlapiGetFieldValue(fldnam);

Syntax of 2.0 - Record.getValue(options)

- CurrentRecord.getValue(options)

* var value = objRecord.getValue({  
   fieldId: 'item'  
  });

**nlapiSetFieldValue(fieldtype,Value)**

1.0 - nlapiSetFieldValue(fldnam, value, firefieldchanged, synchronous)

* var objRecord = nlapiSetFieldValue(fldnam, value, firefieldchanged);

2.0 - Record.setValue(options)

- CurrentRecord.setValue(options)

* objRecord.setValue({  
   fieldId: 'item',  
   value: true,  
   ignoreFieldChange: true  
  });

**nlapiLookupField(type,id,field)**

1.0 - nlapiLookupField(type, id, fields, text)

* var objRecord = nlapiLookupField(type, id, fields, text);

2.0 - search.lookupFields(options)

* var fieldLookUp = search.lookupFields({  
   type: search.Type.SALES\_ORDER,  
   id: '87',  
   columns: ['entity', 'subsidiary', 'name', 'currency']  
  });

**nlapiSubmitField(type,id,values)**

1.0 - nlapiSubmitField(type, id, fields, values, doSourcing)

* var objRecord = nlapiSubmitField(type, id, fields, values, doSourcing);

2.0 - record.submitFields(options)

* var otherId = record.submitFields({  
   type: 'customrecord\_book',  
   id: '4',  
   values: {  
   'custrecord\_rating': '2'  
   }  
  });