

# REST - Update

---

The EWUpdate REST operation updates the specified record and returns the final, updated contents of the record as encoded data.

- **Returns:** The final, updated contents of the record as encoded data.
- **Supported Content-Type:** application/x-www-form-urlencoded
- Compatible with `EWAsyncStatus`. If you use this operation asynchronously, you can use `EWAsyncStatus` to check its status.
- Accepts a URL with URL-encoded parameters and record data. For more information about general URL conventions, see [REST Interface](#).

In place of the usual ID parameter, you can use `$searchSQL` to identify the record to update, provided that the search returns exactly one record and no more. For example, if you have a unique `ext_id` field in Agiloft, you can update a single record that has `ext_id='a0B2c345'` using `&$searchSQL=ext_id='a0B2c345'` in the URL.

If a record is locked, you can force a record update in two different ways using the `&$operationHints=NOLOCK` parameter:

- Adding it to the end of the URL
- Adding it to the POST body below login, password, and language.

## Example

Let's build upon the record created in the [REST - Create](#) example, on an instance of Agiloft that is available on localhost, port 8080 with a KB called Demo. This example code updates that record by linking that employee to an opportunity record for company Agiloft.

This operation was separated from the Create example for illustrative purposes only. The same criteria could have been supplied in the create request.

The following needs to be considered:

- The Employee table is a subtable of the People table.
- The link to Opportunities is created via a Linked Field that imports multi-value fields `company_name` and `opportunity_name`.
- We will use the Query By Example capability and supply the name of the company as an example.

The following request is issued:

```
https://localhost:8080/ewws/EWUpdate?$KB=Demo&$table=contacts.
employees&$login=admin&$password=qwerty&$lang=en&id=358&_1576_company_name0=:IBM
```

The following result is returned, when the record is successfully updated:

```
EWREST_full_name='John Doe';
EWREST__1576_company_name0='IBM';
EWREST_f_group_0='Service Manager';
EWREST_id='358';
EWREST__106_sw_description='Service Management Team';
EWREST_login='jdoe';
EWREST_date_updated='Dec 27 2017 04:40:24';
EWREST_type='employees';
EWREST_date_created='Dec 27 2017 04:34:38';
EWREST_rep_email='example23@example.com';
EWREST_default_approval_title='Document Approval';
EWREST_last_name='Doe';
```

## JavaScript Example

Here is an example of a JavaScript-based client that invokes the REST interface via AJAX:

```
function xmlhttpGet (strURL) {
    var xmlhttpReq=false;
    var self=this;
    // Mozilla/Safari
    if (window.XMLHttpRequest) {
        try {
            netscape.security.PrivilegeManager.
enablePrivilege("UniversalBrowserRead");
        } catch (e) {
            alert("Permission UniversalBrowserRead denied.");
        }
        self.xmlHttpReq=new XMLHttpRequest();
    } // IE
    else if (window.ActiveXObject) {
        self.xmlHttpReq=new ActiveXObject("Microsoft.XMLHTTP");
    }
    self.xmlHttpReq.open('GET', strURL, true);
    self.xmlHttpReq.onreadystatechange=requestComplete;
    self.xmlHttpReq.send(null);
}
function requestComplete() {
    if (xmlHttpReq.readyState==4||xmlHttpReq.readyState=="complete") {
        alert ("Update completed");
    }
}
function main() {
    xmlhttpGet('https://localhost:8080/ewws/EWUpdate?$KB=Demo&$table=contacts.
employees&$login=admin&$password=qwerty&$lang=en&id=358&_1576_company_name0=:IBM');
}
```