

Explore GlideQuery

Lab 7.4

10-15 minutes

Lab objectives

You will perform the following in this lab:

- Create a Development department recall list.
- Retrieve a user's mobile phone number.
- Uppdate their mobile phone number.
- Ddelete a user record.

A. Create a Development department recall list

1. Navigate to **System Definition > Fix Scripts**.

Note: *Fix Scripts is a good place to store and reuse scripts that fix data issues. For more information go to the Product Documentation website and enter "Fix scripts" in Search documents.*

2. Locate and open the following **Fix Script**.

Name: **Lab 7.4 GlideQuery create**

Description (add the Description, if not populated): **Test GlideQuery create**

3. Verify the script matches the script shown here.

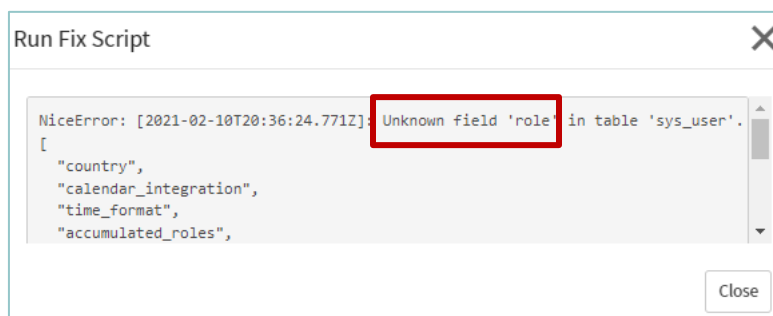
```
1  new GlideQuery('sys_user')
2  .insert({
3      user_name: 'walt.disney',
4      first_name: 'Walt',
5      last_name: 'Disney',
6      department: '221f3db5c6112284009f4becd3039cc9',
7      roles: 'admin',
8      email: 'walt.disney@example.com',
9      phone: '555-1212',
10     mobile_phone: '815-1945'
11 });
```

```

12
13 new GlideQuery('sys_user')
14 .insert({
15     user_name: 'mickey.mouse',
16     first_name: 'Mickey',
17     last_name: 'Mouse',
18     department: '221f3db5c6112284009f4becd3039cc9',
19     roles: 'itil',
20     email: 'mickey.mouse@example.com',
21     phone: '555-2112',
22     mobile_phone: '606-1944'
23 });
24
25 new GlideQuery('sys_user')
26 .insert({
27     user_name: 'minnie.mouse',
28     first_name: 'Minnie',
29     last_name: 'Mouse',
30     department: '221f3db5c6112284009f4becd3039cc9',
31     roles: 'itil',
32     email: 'minnie.mouse@example.com',
33     phone: '555-1234',
34     mobile_phone: '127-1941'
35 });

```

4. Change "roles" to "role" on line 7.
5. Select **Save**.
6. Select **Run Fix Script**.
7. Select **Proceed**.



Note: Congratulations! You've experienced your first NiceError. This is a perfect example of field checking since it points out that the unknown field 'role' in the table 'sys_user' caused the NiceError. Also, in case you're wondering, even though the error was for the first user, the other users were **not** inserted into the system.

8. Change "role" back to "roles" on line 7 and repeat steps 5-7.

9. Navigate to **User Administration > Users**. Filter by '**Created on Today**' and verify your results are as those shown below. You may need to configure your Default List view to include all the columns.

User ID	Name	First name	Last name	Department	Roles	Email	Business phone	Mobile phone
<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>
walt.disney	Walt Disney	Walt	Disney	Development	admin	walt.disney@example.com	555-1212	815-1945
mickey.mouse	Mickey Mouse	Mickey	Mouse	Development	Itil	mickey.mouse@example.com	555-2112	606-1944
minnie.mouse	Minnie Mouse	Minnie	Mouse	Development	Itil	minnie.mouse@example.com	555-1234	127-1941

B. Retrieve Walt's Phone Number

1. Create a new Fix Script.

Name: **Lab 7.4 GlideQuery retrieve**
Description: **Test GlideQuery retrieve**

2. Type the following script.

```
1  new GlideQuery('sys_user')
2    .where('active', true)
3    .where('user_name', 'walt.disney')
4    .select('user_name', 'sys_id', 'mobile_phone')
5  ▾ .forEach(function(u) {
6      gs.info('\n'+user name = ' + u.user_name + '\n'+sys_id = ' + u.sys_id +
7      '\n'+ 'mobile phone = ' + u.mobile_phone);
      });
```

3. Select **Save**.
4. **Run Fix Script**.
5. Select **Proceed**.
6. Verify your results resemble those shown below. Please note that your **sys_id** will vary from this example.

```
*** Script:
user name = walt.disney
sys_id = 578585aa994da0107f4401b700f144b4
mobile phone = 815-1945
[0:00:00.003] Total Time
```

7. Copy Walt's **sys_id** to your clipboard. You will need it in the next two sections.

C. Update Walt's Phone Number

1. Create a new Fix Script.

Name: **Lab 7.4 GlideQuery update**

Description: **Test GlideQuery update**

2. Type the following script. Enter Walt's **sys_id** from the previous script into the where statement.

```
1 new GlideQuery('sys_user')
2   .where('sys_id', '578585aa994da0107f4401b700f144b4')
3   .update({
4     mobile_phone: '800-Cof-fee1'
5   });
```

3. Select **Save**.
4. Select **Run Fix Script**.
5. Select **Proceed**.
6. Navigate to **User Administration > Users** and verify if your results resemble those shown here.

Email	Business phone	Mobile phone
walt.disney@example.com	555-1212	800-Cof-fee1

D. Delete Walt's user record

1. Create a new Fix Script.

Name: **Lab 7.4 GlideQuery delete**

Description: **Test GlideQuery delete**

2. Type the following script and enter Walt's **sys_id** in the where statement.

```
1 new GlideQuery('sys_user')
2   .where('active', true)
3   .where('sys_id', '578585aa994da0107f4401b700f144b4')
4   .deleteMultiple();
```

3. Select **Save**.
4. **Run Fix Script**.

5. Select **Proceed**.
6. Navigate to **User Administration > Users**.
7. Filter by **Created on today** and verify your results resemble those shown here.

All										
<input type="checkbox"/>		User ID	Name	First name	Last name	Department	Roles	Email	Business phone	Mobile phone
		minnie.mouse	Minnie Mouse	Minnie	Mouse	Development	itil	minnie.mouse@example.com	555-1234	127-1941
		mickey.mouse	Mickey Mouse	Mickey	Mouse	Development	itil	mickey.mouse@example.com	555-2112	606-1944

Lab Completion

Great job! You are now familiar with GlideQuery basics. With a little practice you will become a GlideQuery API guru. However, given the shameless way you treated Walt, you are no longer welcome at Disney Parks. 😞