usersWithEmail()

This function returns the **user names** of users matching a given **email address**.

Within Jira it is possible that multiple users have the same email address. In this case a text list with all related user names will be returned. If only one user is excepted though, the function first() can be used.

This function is case insensitive.

Syntax

```
usersWithEmail(emailAddress) #Output: Text list
```

Examples

Parser expression	Description
`usersWithEmail("b.smith@test.com")`	This example returns a list with all usernames for the email address b. smith@test.com, e.g.: **[b.smith]**
`usersWithEmail("B.SMITH@test.com")`	This example returns the same usernames like in the above example, since the function is case insensitive.
`usersWithEmail("j.doe@company.com")`	This example returns a list with all usernames for the email address j. doe@company.com, e.g.: **[j.doe, john-doe, john.doe]**
`first(usersWithEmail("j.doe@company.com"))`	This example returns only the first username with the given email address, e.g.: **j.doe**
`toString(textOnStringList(toStringList(%{issue. cf10400}, ", "), toString(usersWithEmail(%{seed. text}))))`	This example returns the user names for all email addresses, found in a custom text field (with ID 10400) that stores email addresses, e.g.: **b.smith, a.grant, d.conner** ## more info... Steps in this example: 1. Take all individual values found in the custom field (which are separated by ", ") and convert them into a text list with toStringList() 2. Perform an operation on each (seed) element of that list with textOnStringList() 3. The operation itself is to convert the email address for each element in the list (%{seed.text}) into a valid user names 4. Convert the output into a simple text with toString()

Additional information

Parameters used in this function

Parameter	Input (data type)	Description
emailAddress	TEXT LIST	Text containing a valid email address.

Output

This function returns a TEXT LIST

You might want to wrap the method with the toString() function so that the result can be parsed to a text field.

Use cases and examples

Use case

No content found.