

AppWorks Platform Example Project

Send Email To Many

Home Page ▾

Home > 00008

Print

Delete

Send email

Description

Concat related emails into To address

Interested Party Emails

ggreen@innovate.com;bbrown@innovate.com;bbblue@innovate.com

Interested Parties

☒ Gary Green

☒ Ben Brown

☐ AppWorks Developer

☒ Barbara Blue

New email

To


ggreen@innovate.com;bbrown@innovate.com;bbblue@innovate.com

Cc

Subject

Sending email to multiple recipients

Attachments




↩ ↪

B *I* U ~~S~~ *I*_x

Font ▾

Size ▾



This email is being sent to multiple people

Introduction	3
Business Use Case	3
The Design Challenge	3
The Solution	3
About OpenText	4

Introduction

This guide outlines the configuration of the Example AppWorks Platform Project.

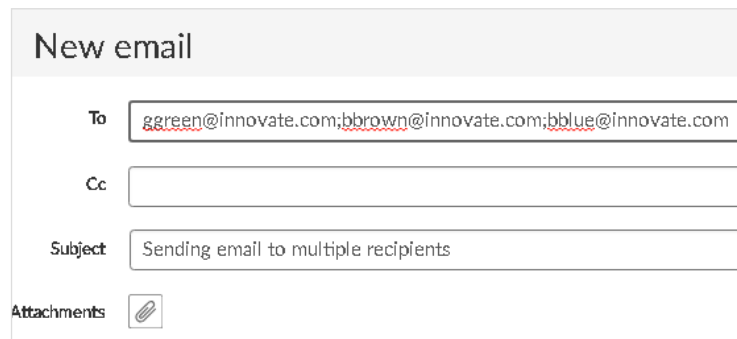
Business Use Case

The use case in this example project is the sending of an email to multiple 'interested parties' of a case, where a one-to-many relationship to an entity is used to maintain the list of 'interested parties'. It also applies to any situation where you want to use repeating entity values but use a concatenated list of those values in the primary entity user interface, such as in a list or on a form.

The Design Challenge

When using a one-to-many relationship between entities, the expression language does not allow you to access the values of the multiple related entities, apart from some arithmetic calculations such as totalling numeric values.

In this use case there is a need to build the list of addresses from the related entities so that it can be used to pre-populate the To field of an Email Template as shown here.



The screenshot shows a 'New email' form with the following fields:

- To:** A text field containing the concatenated email addresses: `ggreen@innovate.com;bbrown@innovate.com;bblue@innovate.com`. The addresses are separated by semicolons and underlined.
- Cc:** An empty text field.
- Subject:** A text field containing the text: 'Sending email to multiple recipients'.
- Attachments:** A section with a paperclip icon and an empty space for adding attachments.

The Solution

The solution is to use an additional attribute on the primary entity that holds the concatenated list of values, and a business process that concatenates the appropriate value(s) from the related entities and updates the new property of the primary entity each time a relationship changes. The process is triggered with the use of a couple of Rules, when the primary entity is created, and when a related entity is added or removed. An additional Rule is used for a specific situation where the last relationship is removed.

About OpenText

OpenText enables the digital world, creating a better way for organizations to work with information, on premises or in the cloud. For more information about OpenText (NASDAQ: OTEX, TSX: OTC) visit opentext.com.

Connect with us:

[OpenText CEO Mark Barrenechea's blog](#)

[Twitter](#) | [LinkedIn](#) | [Facebook](#)