37

#bcopendiscussion

How to Use Txt2AL.exe

10 April 2021

Saurav Dhyani
https://www.sauravdhyani.com/

#bcopendiscussion

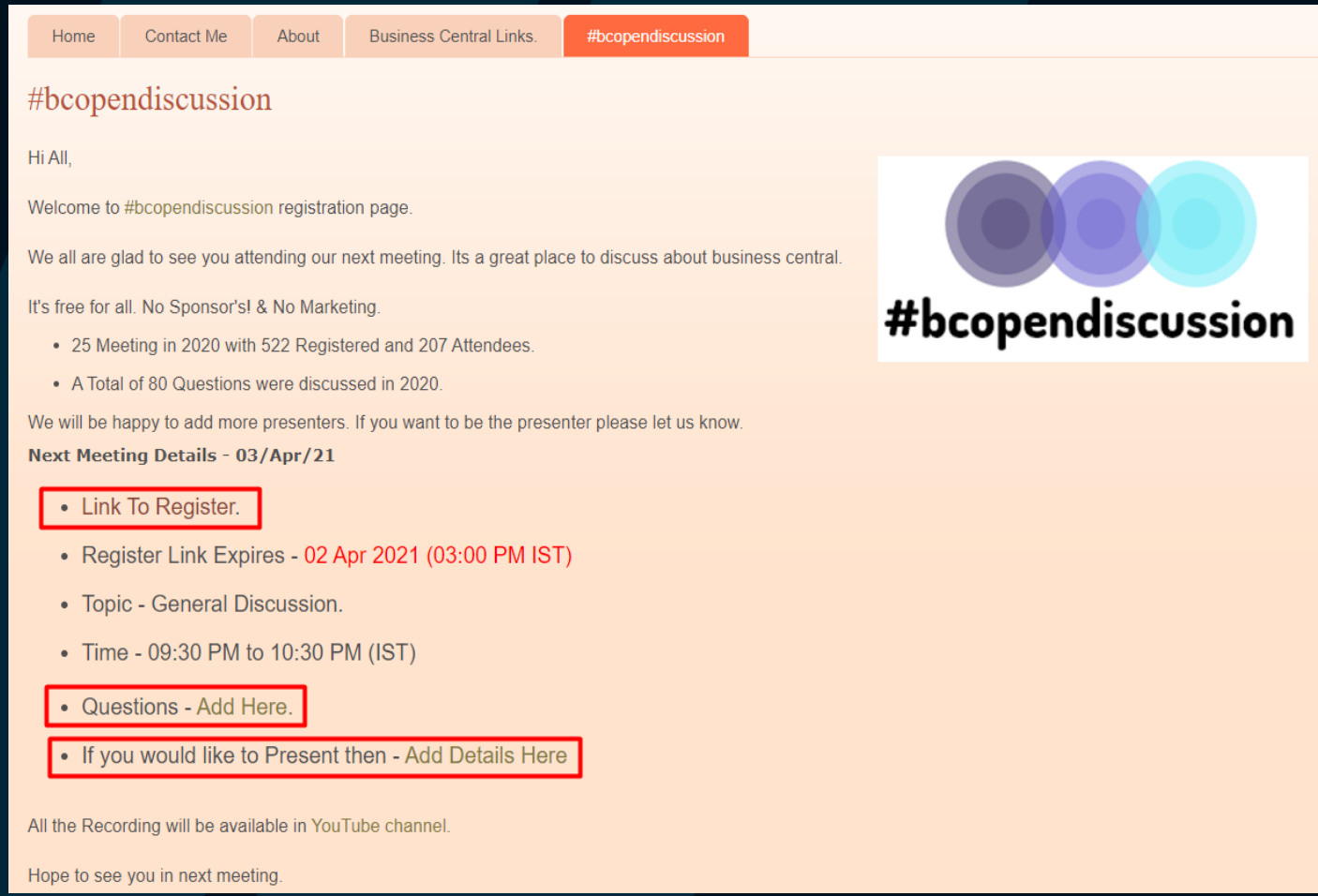
Where Do I Register?

- Register Here –

http://bit.ly/msdyn365bcopendiscussion

- Meeting Attendees –

If you would like to be default attendee for future meeting, please drop your name & Email address in chat during meeting.



#bcopendiscussion

Upcoming Events.

Where can I register?

https://aka.ms/BCLE

What is the date and time?

April 13, 2021

Pacific Time (UTC -7)

#bcopendiscussion

Upcoming Events.

Power Community – Business Central Bootcamp.

When – 16’th & 17’th April 2021.

Cost – Free.

Tracks –

- Round Table.
- Functional Track.
- Developer Track.

Register Here –

https://events.powercommunity.com/business-central-bootcamp/

#bcopendiscussion

Question 104

Mohamad Alkhateb

I have question about how to consolidate two or more entities in BC and how we can post entries to eliminate transactions between these entities. My project is to know how I can as a financial controller to determine the real value of a group with two or more entities.

Txt2AL Exe.

- Poll.
- What is allowed in C/AL and What not?
- How to Execute Txt2AL exe?

Txt2AL.exe

Minimum Parameters –

-- source **Required**. The path of the directory containing the TXT files.

-- target **Required**. The path of the directory into which the converted AL files will be placed.

Txt2AL.exe

Additional Parameters –

-- rename – Rename the output files to prevent clashes with the source .txt files.

-- format – Format the converted AL Code using standard formatter.

-- type – The type of object to convert. Allowed values: Codeunit, Table, Page, Report, Query, Xmlport.

Txt2AL.exe

Should Use –

-- extensionStartId
-- runtime
-- objectFileNamePattern
-- extensionObjectFileNamePattern
-- dataclassificationDefaulting

latest/major.minor
{name}.{type} / {name}/{type}/{id}
{name}.{type}-{targetName} / {targetid}

Txt2AL.exe

Parameters for Dotnet–

-- injectDotNetAddin - Inject the definition of standard .NET Add-Ins in the resulting .NET package.

-- dotNetAddInsPackage - Inject the definition of .NET Add-Ins from the given .NET package in the resulting .NET package.

-- dotNetTypePrefix - Specify a prefix to be used for all .NET type aliases.

#bcopendiscussion

Txt2AL.exe - Special Parameters

-- stacktrace - Display the stack trace of exceptions raised during the conversion.

-- multithreaded - Run using multiple threads. This improves performance but causes non-repeatable extension numbers and .xlf content.

--fallbackTypeNameResolution - Performs a best effort conversion of code that was exported to the new syntax but that might have unresolved references.

-- transalationFormat - Specify the format to use when generating translation files. The allowed values are: Xliff, Lcg

-- addLegacyTranslationInfo - Add information to the translation file that can be used to migrate existing translations.

Txt2AL.exe Parameter Summary

-- Source
-- target
-- rename
-- type
-- extensionStartId
-- stacktrace
--fallbackTypeNameResolution
-- injectDotNetAddin
-- dotNetAddInsPackage
-- multithreaded
-- dotNetTypePrefix
-- transalationFormat
-- addLegacyTranslationInfo
-- runtime
-- objectFileNamePattern
-- extensionObjectFileNamePattern
-- format
-- dataclassificationDefaulting

#bcopendiscussion

THANK YOU!

Saurav Dhyani

Website - www.sauravdhyani.com
Email – postsaurav@gmail.com
Facebook - @SauravMSDYN365BC
Twitter – @SauravDhyani
YouTube – SauravDhyani