Event Management API Reference

For Dynamics 365 Marketing

Release: March, Version 2.0

Contents

Introduction

This document provides reference information about the Event Management REST API. For information on how and where to use it, see [Build and Host Custom Event Portal](#)

Servers

URL: https://domain/api/v1.0

domain: domain

Version: v1.0

Authorize

apiSecuritySchemes.emApplicationToken

In order to control the access of web applications an event management web application token is required. This event management web application token needs to be generated in CRM. Each origin requires a separate token. More information: [Instructions on generating the token](#).

apiSecuritySchemes.optionalAuthorization

This authorization header to identify and authorize users. This authorization header is not necessarily required if anonymous registration is enabled.

ApiSecuritySchemes.requiredAuthorization

This authorization header to identify and authorize users. Endpoints with this security scheme require this authorization header.

Paths

GET/events/published

Gets all published events. A published event is any event with status **active** and publish status set to **live**

Note: Recurring event templates and templates will not be included.

Parameters

Name	Type	Description	Required

businessUnitId	Guid	The business unit ID of the event used for filtering.	No

Responses

Code: 200

Description: An array of **EventModel**. If there are no published events an empty array is returned.

Content: application/json

Example Value: `[`

```
  {
    "customFields": {},
    "eventId": "string",
    "allowCustomAgenda": true,
    "allowAnonymousRegistrations": true,
    "autoregisterWaitlistItems": true,
    "description": "string",
    "endDate": "string",
    "eventFormat": 0,
    "startDate": "string",
    "timeZone": 0,
    "timeZoneName": "string",
    "eventType": 0,
    "eventLanguage": 0,
    "maxCapacity": 0,
    "eventName": "string",
    "publicEventUrl": "string",
    "readableEventId": "string",
    "showAutomaticRegistrationCheckbox": true,
    "image": "string",
    "showWaitlist": true,
    "enableCaptcha": true
  }
]
```

Model: #/components/schemas/EventModel

GET/events/{readableEventId}

Gets all the information for a specific event ID.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes

Responses

Code: 200

Description:

Content: application/json

Example: {

```
  "customFields": {},
  "eventId": "string",
  "allowCustomAgenda": true,
  "allowAnonymousRegistrations": true,
  "autoregisterWaitlistItems": true,
  "description": "string",
  "endDate": "string",
  "eventFormat": 0,
  "startDate": "string",
  "timeZone": 0,
  "timeZoneName": "string",
  "eventType": 0,
  "eventLanguage": 0,
  "maxCapacity": 0,
  "eventName": "string",
  "publicEventUrl": "string",
  "readableEventId": "string",
  "showAutomaticRegistrationCheckbox": true,
  "image": "string",
  "showWaitlist": true,
  "enableCaptcha": true
}
```

Code: 404

Description: If the specified *readableEventId* does not exist.

Model: #/components/schemas/EventModel

GET/events/{readableEventId}/captcha

Gets all the information to display the captcha.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes

Responses

Code: 200

Description: If the readable event ID is valid an object containing the captcha data is returned. If captcha is disabled for an event, the object contains empty data.

Content: application/json

Example: {

```
  "FlowId": "string",
  "HipUrl": "string",
  "ObjectName": "string",
  "FunctionName": "string"
}
```

Code: 400

Description: If the event for the specified *readable* ID cannot be found.

Model: #/components/schemas/CaptchaData

GET/events/{readableEventId}custom-registration-fields

Gets all the custom registration fields of the specified event.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes

Responses

Code: 200

Description: An array of **CutomRegistrationFieldResponseModel**. If there are no associated custom registration fields for the specified event ID an empty array is returned.

Content: application/json

Example: [

```
  {
    "id": "string",
    "value": "string"
  }
]
```

Code: 404

Description: If the specified readable event ID is invalid or does not belong to a event.

Model: #/components/schemas/CustomRegistrationFieldResponseModel

GET/events/{readableEventId}/passes

Gets all the passes of the specified event.

Parameters:

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes

Responses:

Code: 200

Description: An array of **PassModel**. If there are no associated passes for the specified event ID an empty array is returned. If there is no existing event for the specified event ID an empty array is returned.

Content: application/json

Example:
```
[
  {
    "customFields": {},
    "passId": "string",
    "eventId": "string",
    "price": 0,
    "numberOfPassesSold": 0,
    "numberOfPassesLeft": 0,
    "passName": "string",
    "currencySymbol": "string",
    "sessions": [
      {
        "customFields": {},
        "id": "string",
        "camerasPermitted": 0,
        "detailedDescription": "string",
        "endTime": "string",
        "language": 0,
        "name": "string",
        "recordingsPermitted": 0,
        "sessionFormat": 0,
        "sessionObjectives": "string",
        "sessionPreRequisites": "string",
        "sessionSummary": "string",
        "sessionType": 0,
        "startTime": "string",
        "userEligibleToRegister": true,
        "alreadyRegistered": true,
        "speakers": [
          {
            "customFields": {},
            "id": "string",
            "about": "string",
            "accomplishments": "string",
            "blog": "string",
            "email": "string",
            "linkedin": "string",
            "name": "string",
            "publications": "string",
            "title": "string",
            "twitter": "string",
```

```
        "website": "string",
        "imageUrl": "string",
        "sessions": [
          null
        ]
      }
    ]
  }
 ]
 }
]
```

Code: 404

Description: If the specified readable event ID is invalid or does not belong to a event.

Model: #/components/schemas/PassModel

POST/events/{readableEventId}registrations

Registers the specified attendee(s) to the specified event. If the specified event is a paid event, the registration process is not completed at this point. You can find instructions on how to finalize a registration with payment in our documentation. If the authorization header is present, the registration is linked to the authorized user. Requests without authorization header are only allowed for events with anonymous registrations enabled.

In case, if the **walk-in** attribute is set to true in the request bidy, it will also create a check-in. Check-in can be created for Event and for Session as well.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes
Request body	application/json		Yes

Example:

```
{
  "attendees": [
    {
      "firstName": "John",
      "lastName": "Doe",
      "email": "johndoe@contoso.com",
      "passId": "null",
      "waitlisted": true,
      "autoRegister": null,
      "responses": [],
```

```
        "sessions": "null",
        "attendeeSessions": [
{
 "sessionId":"9b2c0a01-4c5d-ea11-b826-00155d3e6106",
  "waitlisted":false,
 "walkin": true
     }],
            "walkin": true
   }],
"hipObject": null
}
```

Model: #components/schemas/RegistrationRequestModel

Responses

Code: 200

Description: The **RegistrationResultModel** contains information on whether the registration was successful or not. Additionally, it contains the redirect URL to the payment provider and the purchase id (empty if the event is a free event).

Content: application/json

Example: {

```
  "status": "string",
  "errorMessage": "string",
  "redirectUrl": "string",
  "purchaseId": "string"
}
```

Model: #components/schemas/RegistrationRequestModel

GET/events/{readableEventId}/qrcode

Retrieves the event public url as a QR code.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes

Responses

Code: 200

Description: A QR code image representing the value of the ***msevtmgt_publiceventurl*** entity attribute.

Content: image/png

Example:
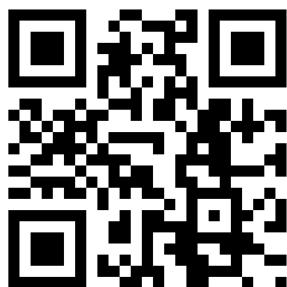
Code: 400

Description: Occurs when the parameter for **readableEventId** provided is empty.

Code: 404

Description: Occurs when the specified readable event ID does not belong to an event or the event does not contain a public event url.

GET/events/{readableEventId}/registrations/count

Gets the number of registrations for a specific event.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes

Responses

Code: 200

Description: The number of registrations for a specific event.

Content: application/json

Code: 404

Description: If the specified readable event ID is invalid or does not belong to a event.

GET/events/{readableEventId}/sessions

Gets all sessions that belong to the specified event ID. Requests without authorization header are only allowed for events with anonymous registrations enabled.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes

Responses

Code: 200

Description: An array of *SessionModel*. If there are no associated sessions for the specified event ID an empty list is returned.

Content: application/json

Example: [

```
{
  "customFields": {},
  "id": "string",
  "camerasPermitted": 0,
  "detailedDescription": "string",
  "endTime": "string",
  "language": 0,
  "name": "string",
  "recordingsPermitted": 0,
  "sessionFormat": 0,
  "sessionObjectives": "string",
  "sessionPreRequisites": "string",
  "sessionSummary": "string",
  "sessionType": 0,
  "startTime": "string",
  "userEligibleToRegister": true,
  "alreadyRegistered": true,
  "speakers": [
    {
      "customFields": {},
      "id": "string",
      "about": "string",
      "accomplishments": "string",
```

```
            "blog": "string",
            "email": "string",
            "linkedin": "string",
            "name": "string",
            "publications": "string",
            "title": "string",
            "twitter": "string",
            "website": "string",
            "imageUrl": "string",
            "sessions": [
                null
            ]
        }
    ]
  }
]
```

Code: 404

Description: If the specified readable event ID is invalid or does not belong to an event.

Model: #components/schemas/SessionModel

POST/events/{readableEventId}/sessions/{sessionId}/registrations

Registers the authenticated user to the specified session of the specified event. The authorization header must be present in order to register for an event. This requires both, the *emApplicationtoken* and the authorization token.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes
sessionId	String	The ID of the session to which the user should be registered	Yes

Responses

Code: 200

Description: True if the registration was successful, false otherwise.

Content: application/json

GET/events/{readableEventId}/speakers

Gets all speakers of the specified event.

Parameters

Name	Type	Description	Required

readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes

Responses

Code: 200

Description: An array of **SpeakerModel**. If there are no associated speakers for the specified event ID an empty array is returned.

Content: application/json

Example:
```json
[
  {
    "customFields": {},
    "id": "string",
    "about": "string",
    "accomplishments": "string",
    "blog": "string",
    "email": "string",
    "linkedin": "string",
    "name": "string",
    "publications": "string",
    "title": "string",
    "twitter": "string",
    "website": "string",
    "imageUrl": "string",
    "sessions": [
      {
        "customFields": {},
        "id": "string",
        "camerasPermitted": 0,
        "detailedDescription": "string",
        "endTime": "string",
        "language": 0,
        "name": "string",
        "recordingsPermitted": 0,
        "sessionFormat": 0,
        "sessionObjectives": "string",
        "sessionPreRequisites": "string",
        "sessionSummary": "string",
        "sessionType": 0,
        "startTime": "string",
        "userEligibleToRegister": true,
        "alreadyRegistered": true,
        "speakers": [
          null
        ]
      }
    ]
  }
]
```

Code: 404

Description: If the specified readable event ID is invalid or does not belong to a event.

Model: #components/schemas/SpeakersModel

GET/events/{readableEventId}/sponsorship

Gets all sponsors of the specified event.

Note: The logo of the sponsors is not included. Please use the **sponbsorship/{sponsorshipId}/logo** endpoint to retrieve the logo of a sponsor.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes

Responses

Code: 200

Description: An array of **SponsorshipModel** excluding the logo of the sponsorship. If there are no associated sponsors for the specified event ID an empty array is returned.

Content: application/json

Example: {

```
  "customFields": {},
  "id": "string",
  "description": "string",
  "sponsorshipTopic": "string",
  "sponsorId": "string",
  "sponsorName": "string"
}
```

Code: 404

Description: If the specified readable event ID is invalid or does not belong to a event.

Model: #components/schemas/SponsorshipModel

GET/events/{readableEventId}/tracks

Gets all session tracks of the specified event.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the	Yes

		information should be retrieved.	

Responses

Code: 200

Description: An array of **SessionTrackModel**. If there are no associated sponsors for the specified event ID an empty array is returned.

Content: application/json

Example: `{`

```
  "customFields": {},
  "id": "string",
  "description": "string",
  "name": "string"
}
```

Code: 404

Description: If the specified readable event ID is invalid or does not belong to a event.

Model: #components/schemas/SessionTrackModel

POST/events/{readableEventId}/registrations/finalize

Creates event registration for the paid event that has been initialized in the previous step by calling **/events/{readableEventId}/registrations** endpoint. Result of the previous call was purchaseId that is now an input parameter (as a part of FinalizeRegistrationRequestModel) for this endpoint. Endpoint must be called after the after the payment process was successful.

 Note: This endpoint is disabled by default. It is enabled by going to Event Administration and checking Enable demo payment confirmation checkbox.

Parameters

Name	Type	Description	Required
readableEventId	string	The readable ID of the event for which the information should be retrieved.	Yes
Request body	Application/json		Yes

Example: {

"purchaseId": "string",

"payload": "string"

}

Responses

Code: 200

Description: The *RegistrationResultModel* contains information on whether the registration was successful or not.

Content: application/json

Example:
```
{
  "status": "string",
  "errorMessage": "string",
  "redirectUrl": "string",
  "purchaseId": "string"
}
```

Model: #components/schemas/FinalizeRegistrationRequestModel

GET/purchases/{purchaseId}

Gets all information for a specified event ID.

Parameters

Name	Type	Description	Required
purchaseId	string	The ID of the purchase which is returned by the */register* endpoint when initiating a paid event.	Yes

Responses

Code: 200

Description:

Content: application/json

Example:
```
{
  "eventName": "string",
  "amount": 0,
  "currencyName": "string",
  "isoCurrencyCode": "string",
  "currencySymbol": "string"
}
```

Code: 400

Description: If the specified purchase ID is invalid.

GET/purchases/{purchaseId}/details

Gets all information for a specified event ID.

Parameters

Name	Type	Description	Required
purchaseId	string	The ID of the purchase which is returned by the */register* endpoint when initiating a paid event.	Yes

Responses

Code: 200

Description:

Content: application/json

Example:

```json
{
"eventName": "string",
"amount": 0,
"currencyName": "string",
"isoCurrencyCode": "string",
"currencySymbol": "string",
"attendees": [
  {
    "firstName": "string",
    "lastName": "string",
    "email": "string"
  },
  {
    "firstName": "string",
    "lastName": "string",
    "email": "string"
  }
]
}
```

Code: 400

Description: If the specified purchase ID is invalid.

Code 401:

Description: If the user has not been authenticated.

GET/sessions/{sessionId}/registrations/count

Gets the number of registrations for a specified session.

Parameters

Name	Type	Description	Required
sessionId	string	The session ID for which the information should be retrieved.	Yes

Responses

Code: 200

Description: The number of registrations for a specified session.

Content: application/json

Example:

Code: 404

Description: If the specified session ID is invalid or does not belong to a session.

GET/speakers/{speakerId}/image

The image of a specified speaker.

Parameters

Name	Type	Description	Required
speakerId	string	The ID of the speaker for which the image should be retrieved.	Yes

Responses

Code: 200

Description: The base64 encoded image.

Content: image/gif

Example: string

Code: 404

Description: If the speaker or speaker image for the specified speaker ID does not exist.

GET/sponsorship/{sponsorshipId}/logo

Gets the sponsorship logo base64 encoded for a specified sponsorship ID.

Parameters

Name	Type	Description	Required
sponsorshipId	string	The ID of the sponsorship for which the logo should be retrieved.	Yes

Responses

Code: 200

Description: The sponsorship logo base64 encoded.

Content: image/gif

Example: `string`

Code: 404

Description: If the sponsorship or sponsorship logo for the specified sponsorship ID does not exist.

GET/tracks/{trackId}/sessions

Gets all sessions for a specified track.

Parameters

Name	Type	Description	Required
trackId	string	The ID of the track	Yes

Responses

Code: 200

Description: An array of *SessionTrackModel*. If there are no associated sponsors for the specified event ID an empty array is returned.

Content: application/json

Example: `[`

```
  {
    "customFields": {},
    "id": "string",
    "description": "string",
    "name": "string"
  }
]
```

Code: 404

Description: If the specified track ID does not match to a track.

Model: #components/schemas/SessionTrackModel

GET/users/authenticated

Gets the user that is associated to the provided authorization token. We try to associate the user with a contact in Dynamics 365 by using a contact matching strategy (first name + last name + email is used by default). It is possible to change the contact matching strategy in the event administration settings in Dynamics 365. More information Event Management Settings. A new contact is created if it's not possible to find a matching contact. This requires the *emApplicationtoken* and optionally the authorization token.

Parameters

No parameters

Responses

Code: 200

Description: The user that is associated with the authorization token provided in the request. If there is no authorization token provided, an anonymous user is returned.

Content: application/json

Example: {

```
"id": "string",

"firstName": "string",

"lastName": "string",
"email": "string",
"isAnonymous": true

}
```

Model: #components/schemas/UserModel

GET/users/authenticated/registrations

Gets all event registrations for the user that is associated with the authorization token provided in the request. This requires both, the *emApplicationtoken* and the authorization token.

Parameters

No parameters

Responses

Code: 200

Description: An array of *MyRegistrationModel*. If there are no registrations, an empty list is returned.

Content: application/json

Example: [

```
    {
```

```
    "id": "string",
    "contact": "string",
    "email": "string",
    "event": "string",
    "pass": "string",
    "sessionRegistrations": [
      {
        "id": "string",
        "sessionName": "string",
        "createdon": "string"
      }
    ],
    "createdon": "string"
  }
]
```

Model: #components/schemas/MyRegistrationModel

POST/users/authenticated/registrations/{eventRegistrationId}/cancel

Cancels the specified event registration. The authorization header must be present. This requires both, the *emApplicationtoken* and the authorization token.

Parameters

Name	Type	Description	Required
eventRegistrationId	string	The ID of the event registration.	Yes

Responses

Code: 200

Description: True if successful, false otherwise.

Content: application/json

GET/users/authenticated/waitlist-items

Gets all event waitlist registrations for the user that is associated with the authorization token provided in the request. A waitlist item can either be related to an event or a session. If a user registers for the waitlist of two sessions within an event, this endpoint will return two waitlist items.

Parameters

No parameters

Responses

Code: 200

Description: An array of *waitlistItemModel*. If there are no waitlist registrations an empty list is returned.

Content: application/json

Example: [
  {
    "id": "string" "invited": true,
    "autoRegister": true,
    "contact": "string",
    "email": "string",
    "event": "string",
    "session": "string",
    "createdon": "string"
  }
]

Model: #components/schemas/WaitlistItemModel

POST/checkins

Creates check-in based on input in request body.

Parameters

Name	Type	Description	Required
Request body	Application/json		Yes

Responses

Code: 200

Description: If the check-in is successful.

Example:
/api/v2.0/checkins

Body example:

1. Create checkin for Session based on readable event registration id and session id
   ```
   {

   "eventRegistrationId":"ER Y7XQZQ53E8JBETL74TEN7U9TWD",
   "sessionId":"b66bcb13-4f5d-ea11-b826-00155d3e6106"
   }
   ```

2. Create checkin for Event based on readable event registration id
   ```
   {
     "eventRegistrationId":"ER Y7XQZQ53E8JBETL74TEN7U9TWD"
    }
   ```

3. Create checkin for Session based on readable session registration id
   ```
   {

   "sessionRegistrationId":"SR 2L46XRZUEQUUEB8MTCQUF7FMFH"
    }
   ```

Code: 400

Description: Occurs on a validation error.

Models

Custom Registration Field Response

Properties	Value
id	string
value	string

Custom Registration Field

Properties	Value
customFields	
customRegistrationFieldId	string
text	string
type	integer
isRequired	boolean
choices	string
x-abstract	true

Event

Properties	Value
eventId	string
businessUnitId	guid
allowCustomAgenda	boolean
allowAnonymousRegistrations	boolean
autoregisterWaitlistItems	boolean
description	string
endDate	string
eventFormat	integer
startDate	string
timeZone	integer
timezoneName	string

eventType	integer
eventLanguage	integer
maxCapacity	integer
isCapacityRestricted	integer
eventName	string
publicEventUrl	integer
readableEventId	string
showAutomaticRegistrationCheckbox	boolean
image	string
showWaitlist	boolean
building	Building
room	Room
enableCaptcha	boolean
x-abstract	true

Registration Request Model

Properties	Value
attendees	Attendee
hipObject	Verify Captcha Input Request

Registration Result Model

Properties	Value
status	string
errorMessage	string
redirectUrl	string
purchaseId	string

Captcha Data

Properties	Value
FlowId	string
HipUrl	string
objectName	string
FunctionName	string

Pass

Properties	Value
customFields	
passId	string
eventId	string
price	number
numberOfPassesSold	integer
numberOfPassesLeft	integer
passName	string
currencySymbol	string
Session	Session
x-abstract	true

Session

Properties	Value
id	string
camerasPermitted	integer
detailedDescription	string
endTime	string
language	integer
name	string
recordingsPermitted	integer
sessionFormat	integer
sessionObjectives	string
sessionPreRequisites	string
sessionsummary	string
sessionType	integer
startType	string
userEligibleToRegister	boolean
alreadyRegistered	boolean
building	Building
room	Room
speakers	Speaker
x-abstract	true

Session Track

Properties	Value
id	string

description	string
name	string
x-abstract	true

Speaker

Properties	Value
customFields	
id	string
about	integer
accomplishments	string
blog	string
email	string
linkedin	string
name	string
publications	string
title	string
twitter	string
website	string
imageUrl	string
sessions	Session
x-abstract	true

Sponsorship

Properties	Value
customFields	
id	string
description	string
sponsorshipTopic	string
sponsorId	string
sponsorName	string
x-abstract	true

Sponsorship with Image

Properties	Value
customFields	
id	string
description	string
sponsorshipTopic	string

sponsorId	string
sponsorName	string
Image	string
x-abstract	true

User

Properties	Value
id	string
fullName	string
lastName	string
isAnonymous	boolean
email	string

Waitlist Item

Properties	Value
id	string
invited	boolean
autoRegister	boolean
contact	string
email	string
event	string
session	string
createdon	string

Building

Properties	Value
id	string
accessibleToilets	boolean
additionalFacilities	string
addressComposite	string
addressLine1	string
addressLine2	string
addressLine3	string
city	string
country	string
description	string
disabledAccess	boolean
disabledParking	boolean

email	string
name	string
postalCode	string
publicTelephoneAvailable	boolean
stateProvince	string
telephone1	string
telephone2	string
telephone3	string
website	string
wifiAvailable	boolean
wifiPassword	string
wifiSSID	string

Room

Properties	Value
id	string
description	string
disabledAccess	boolean
name	string

Attendee

Properties	Value
firstName	string
lastName	string
email	string
passId	string
waitlisted	boolean
walkin	boolean
autoRegister	boolean
responses	Custom Registration Field Response
attendeeSessions	Attendee Session

Attendee Session

Properties	Value
sessionId	string
waitlisted	boolean

Verify Captcha Input Request

Properties	Value
FlowId	string
Solution	string
Token	string
ClientIp	string
Type	string

Base Website

Properties	Value
customFields	string
x-abstract	string

My Registration

Properties	Value
Id	string
contact	string
email	string
event	string
pass	string
sessionRegistrations	Session Registration
createdon	string

Session Registration

Properties	Value
id	string
sessionName	string
createdon	string

Purchase Details

Properties	Value
eventName	string
amount	number
currencyName	string
isoCurrencyCode	string
currencySymbol	string
attendees	Purchase Attendee

Purchase

Properties	Value
eventName	string
amount	number
currencyName	string
isoCurrencyCode	string
currencySymbol	string

Purchase Attendee

Properties	Value
firstName	string
lastName	string
email	string

Copyright

This document is provided "as-is". Information and views expressed in this document, including URL and other internet web site references, may change without notice.

Some examples depicted herein are provided for illustration only and are fictitious. No real association or connection is intended or should be inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes. This document is confidential and proprietary to Microsoft. It is disclosed and can be used only pursuant to a non-disclosure agreement.

© 2020 Microsoft. All rights reserved.

Microsoft is a trademark of the Microsoft group of companies. All other trademarks are property of their respective owners.