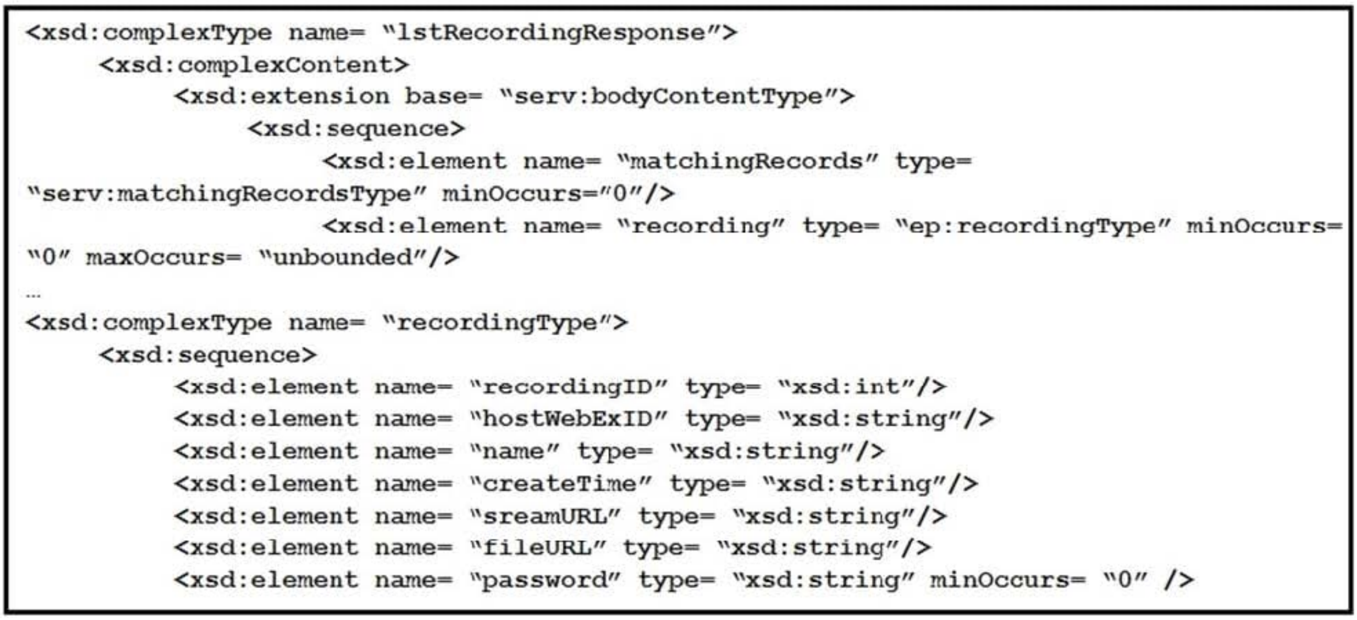
300-920$^{Q\&As}$

Developing Applications for Cisco Webex and Webex Devices (DEVWBX)

Pass Cisco 300-920 Exam with 100% Guarantee

Following Questions and Answers are all new published by Cisco Official Exam Center

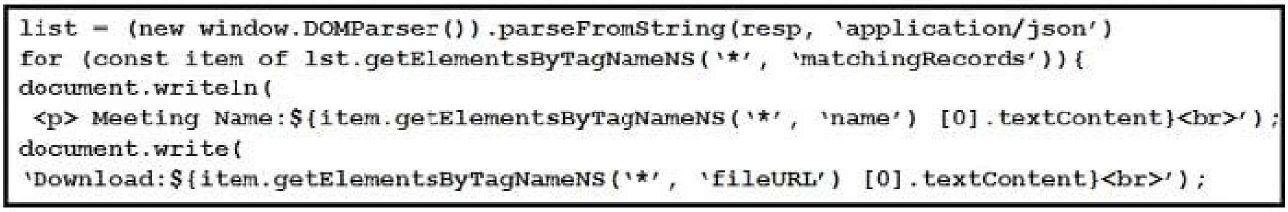
QUESTION 1



```
<xsd:complexType name= “lstRecordingResponse”>
     <xsd:complexContent>
          <xsd:extension base= “serv:bodyContentType”>
               <xsd:sequence>
                    <xsd:element name= “matchingRecords” type=
“serv:matchingRecordsType” minOccurs=”0”/>
                    <xsd:element name= “recording” type= “ep:recordingType” minOccurs=
“0” maxOccurs= “unbounded”/>
...
<xsd:complexType name= “recordingType”>
     <xsd:sequence>
          <xsd:element name= “recordingID” type= “xsd:int”/>
          <xsd:element name= “hostWebExID” type= “xsd:string”/>
          <xsd:element name= “name” type= “xsd:string”/>
          <xsd:element name= “createTime” type= “xsd:string”/>
          <xsd:element name= “sreamURL” type= “xsd:string”/>
          <xsd:element name= “fileURL” type= “xsd:string”/>
          <xsd:element name= “password” type= “xsd:string” minOccurs= “0” />
```

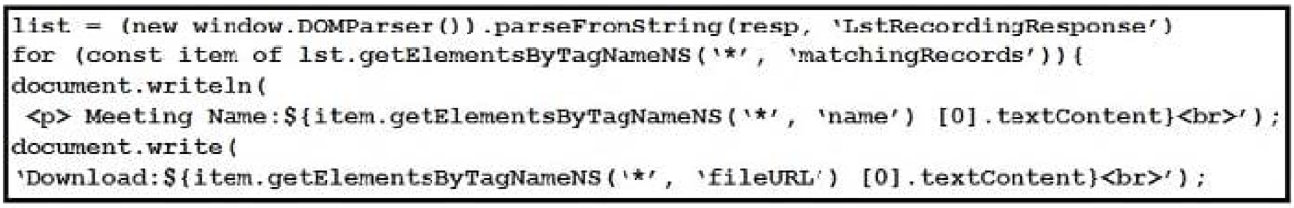
Refer to the exhibit. A snippet from the XSD schema of the Webex Meeting XML API `LstRecordingResponse\\' element is listed in the exhibit. Assuming that a variable named `resp\\' exists that contains the XML response from a successful `LstRecording\\' request, which code snippet correctly generates a simple report that lists meeting names and recording file download links?

A.



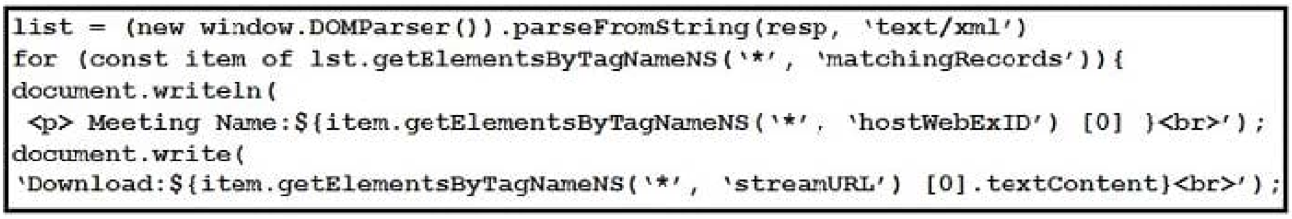
```
list = (new window.DOMParser()).parseFromString(resp, ‘application/json’)
for (const item of lst.getElementsByTagNameNS(‘*’, ‘matchingRecords’)){
document.writeln(
 <p> Meeting Name:${item.getElementsByTagNameNS(‘*’, ‘name’) [0].textContent}<br>’);
document.write(
‘Download:${item.getElementsByTagNameNS(‘*’, ‘fileURL’) [0].textContent}<br>’);
```

B.



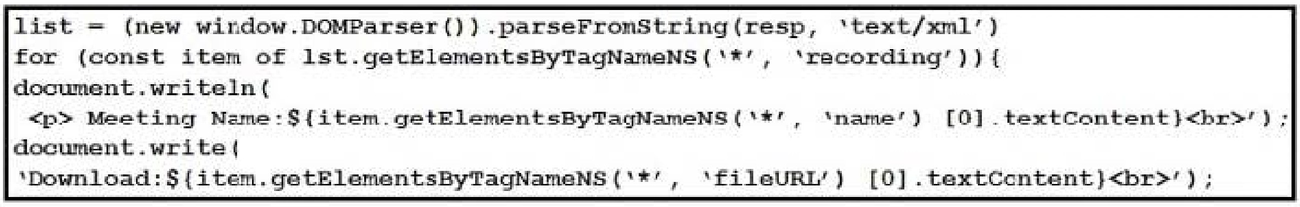
```
list = (new window.DOMParser()).parseFromString(resp, ‘LstRecordingResponse’)
for (const item of lst.getElementsByTagNameNS(‘*’, ‘matchingRecords’)){
document.writeln(
 <p> Meeting Name:${item.getElementsByTagNameNS(‘*’, ‘name’) [0].textContent}<br>’);
document.write(
‘Download:${item.getElementsByTagNameNS(‘*’, ‘fileURL’) [0].textContent}<br>’);
```

C.



```
list = (new window.DOMParser()).parseFromString(resp, ‘text/xml’)
for (const item of lst.getElementsByTagNameNS(‘*’, ‘matchingRecords’)){
document.writeln(
 <p> Meeting Name:${item.getElementsByTagNameNS(‘*’, ‘hostWebExID’) [0] }<br>’);
document.write(
‘Download:${item.getElementsByTagNameNS(‘*’, ‘streamURL’) [0].textContent}<br>’);
```

D.



```
list = (new window.DOMParser()).parseFromString(resp, ‘text/xml’)
for (const item of lst.getElementsByTagNameNS(‘*’, ‘recording’)){
document.writeln(
 <p> Meeting Name:${item.getElementsByTagNameNS(‘*’, ‘name’) [0].textContent}<br>’);
document.write(
‘Download:${item.getElementsByTagNameNS(‘*’, ‘fileURL’) [0].textContent}<br>’);
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: A

QUESTION 2

A Webex Teams bot is deployed but soon it stops responding. Which two explanations are the cause of the issue? (Choose two.)

A. A new webhook was created, which marks the old webhook as inactive.

B. The web server that is set to receive webhooks is not configured to return a 200 message. And the webhook is disabled.

C. The webhook secret is expired and must be refreshed.

D. The refresh token is not being used.

E. The bot owner regenerated the access token on developer.webex.com.

Correct Answer: AB

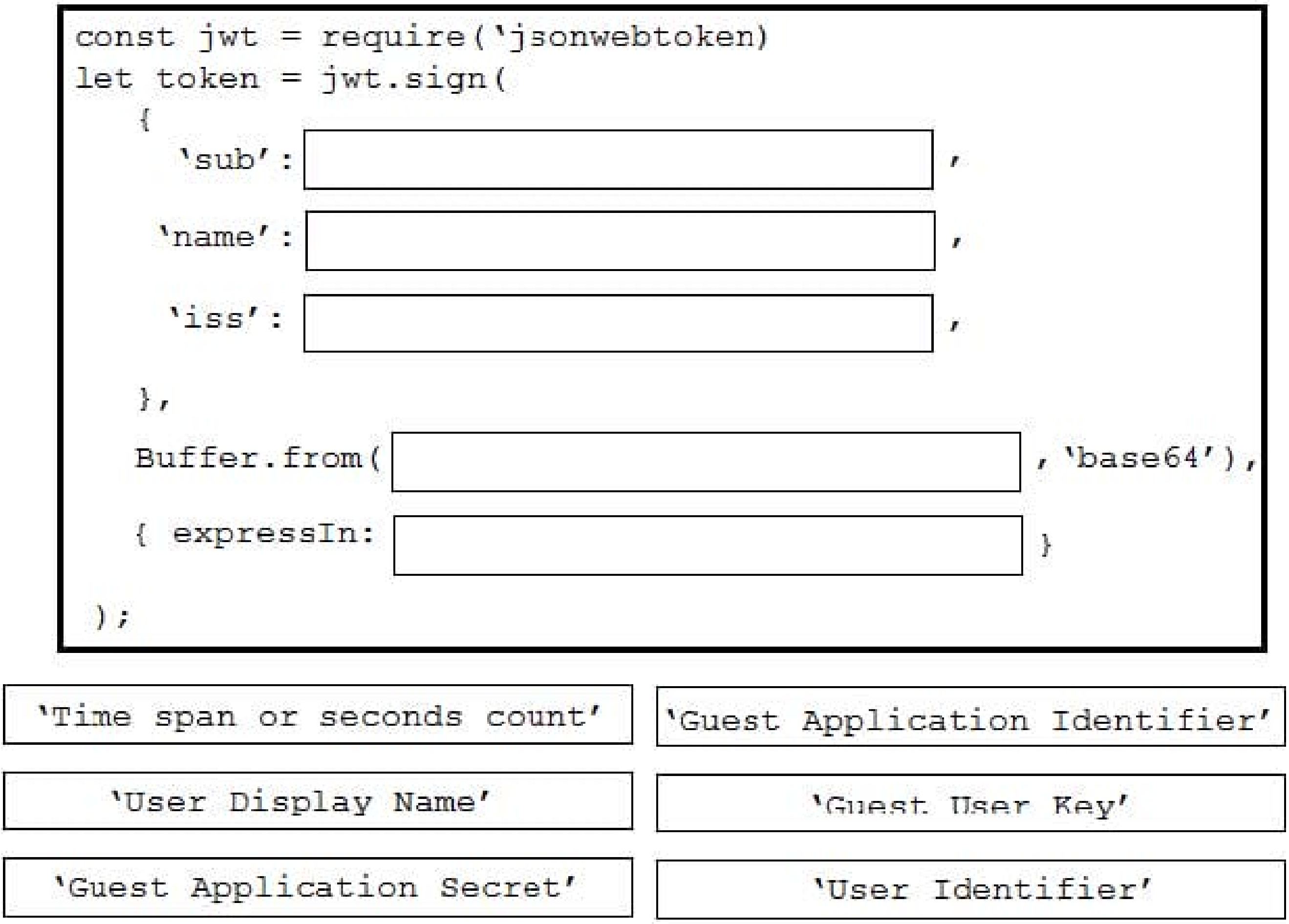
Reference: https://developer.authorize.net/api/reference/features/webhooks.html

QUESTION 3

DRAG DROP

Drag and drop the code to create a valid Guest Issuer JWT token to use with the Webex Teams Browser SDK. Not all options are used.

Select and Place:



```
const jwt = require('jsonwebtoken)
let token = jwt.sign(
   {
     'sub':                              ,

    'name':                              ,

     'iss':                              ,

   },
   Buffer.from(                              , 'base64'),
   { expressIn:                              }
 );
```

'Time span or seconds count'

'Guest Application Identifier'

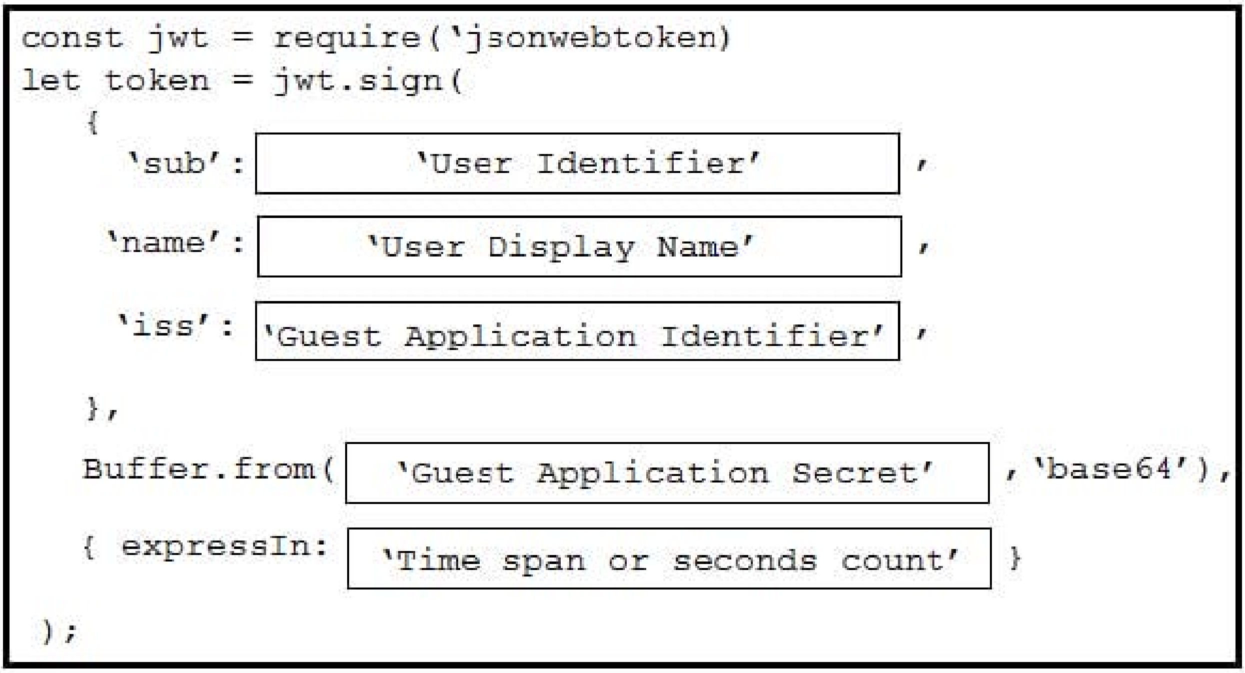
'User Display Name'

'Guest User Key'

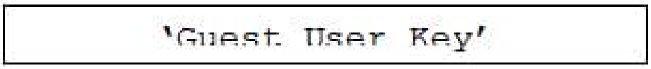
'Guest Application Secret'

'User Identifier'

Correct Answer:



```
const jwt = require('jsonwebtoken)
let token = jwt.sign(
   {
     'sub': 'User Identifier',
    'name': 'User Display Name',
     'iss': 'Guest Application Identifier',
   },
   Buffer.from( 'Guest Application Secret' ,'base64'),
   { expressIn: 'Time span or seconds count' }
 );
```



```
'Guest User Key'
```

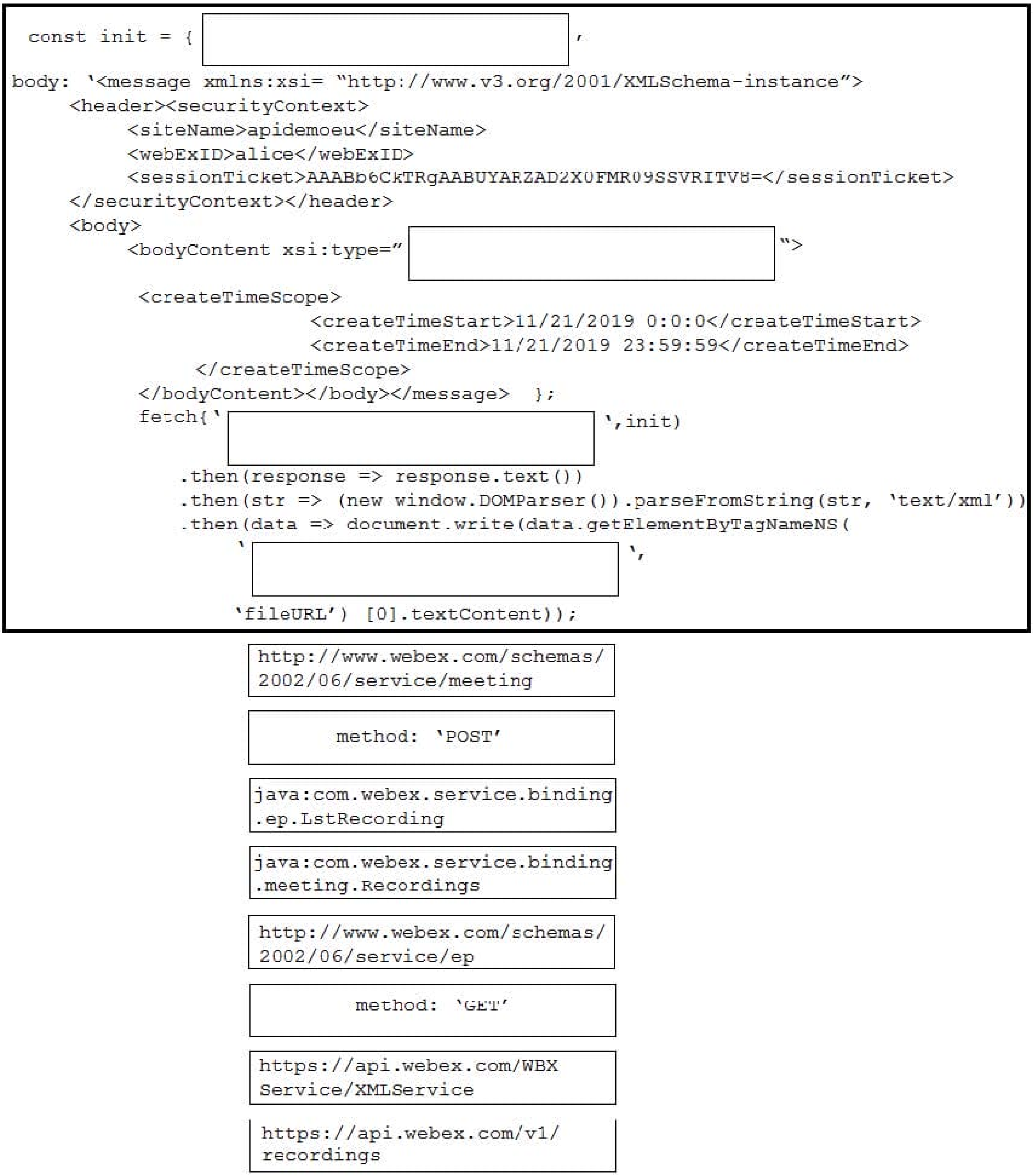
Reference: https://developer.webex.com/docs/guest-issuer

QUESTION 4

DRAG DROP

Drag and drop the code to complete the JavaScript to display the first URL from a user\\'s list of Webex Meetings recordings. Not all options are used.

Select and Place:



```
  const init = {                              ,

body: '<message xmlns:xsi= "http://www.v3.org/2001/XMLSchema-instance">
     <header><securityContext>
          <siteName>apidemoeu</siteName>
          <webExID>alice</webExID>
          <sessionTicket>AAABb6CkTRgAABUYARZAD2X0FMR09SSVRITV8=</sessionTicket>
     </securityContext></header>
     <body>
          <bodyContent xsi:type="                              ">

           <createTimeScope>
                          <createTimeStart>11/21/2019 0:0:0</createTimeStart>
                          <createTimeEnd>11/21/2019 23:59:59</createTimeEnd>
                <createTimeScope>
           </bodyContent></body></message>  };
           fetch('                              ',init)

               .then(response => response.text())
               .then(str => (new window.DOMParser()).parseFromString(str, 'text/xml'))
               .then(data => document.write(data.getElementByTagNameNS(
                    '                              ',

                    'fileURL') [0].textContent));
```

```
http://www.webex.com/schemas/2002/06/service/meeting
```

```
method: 'POST'
```

```
java:com.webex.service.binding.ep.LstRecording
```

```
java:com.webex.service.binding.meeting.Recordings
```

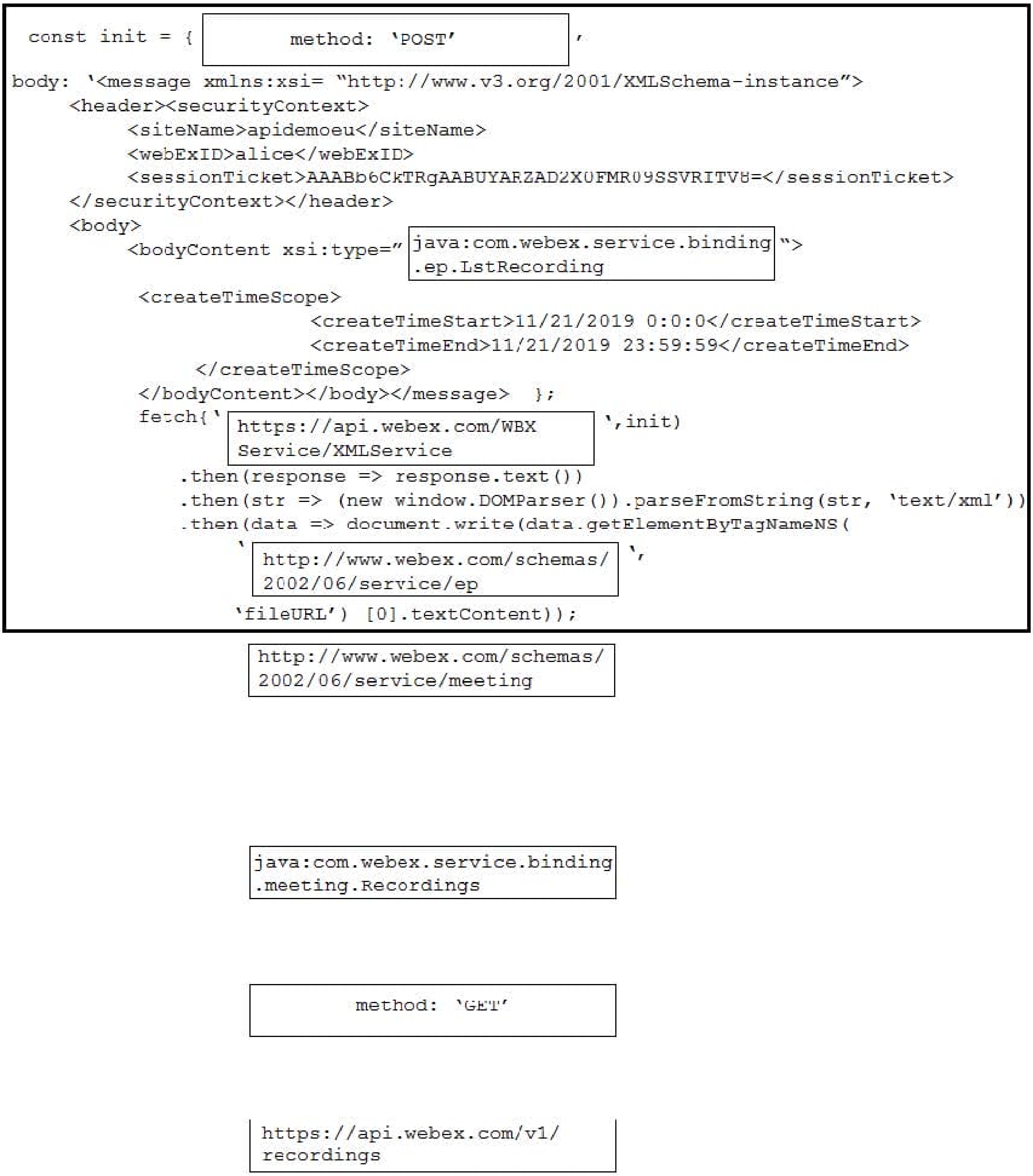
```
http://www.webex.com/schemas/2002/06/service/ep
```

```
method: 'GET'
```

```
https://api.webex.com/WBXService/XMLService
```

```
https://api.webex.com/v1/recordings
```

Correct Answer:



```
 const init = {        method:  'POST'          ,
body: '<message xmlns:xsi= "http://www.v3.org/2001/XMLSchema-instance">
     <header><securityContext>
          <siteName>apidemoeu</siteName>
          <webExID>alice</webExID>
          <sessionTicket>AAABb6CkTRgAABUYARZAD2X0FMR09SSVRITV8=</sessionTicket>
     </securityContext></header>
     <body>
          <bodyContent xsi:type=" java:com.webex.service.binding
                                  .ep.LstRecording             ">
           <createTimeScope>
                          <createTimeStart>11/21/2019 0:0:0</createTimeStart>
                          <createTimeEnd>11/21/2019 23:59:59</createTimeEnd>
                <createTimeScope>
           </bodyContent></body></message>  };
           fetch(' https://api.webex.com/WBX
                   Service/XMLService         ',init)
               .then(response => response.text())
               .then(str => (new window.DOMParser()).parseFromString(str, 'text/xml'))
               .then(data => document.write(data.getElementByTagNameNS(
                      ' http://www.webex.com/schemas/
                        2002/06/service/ep           ',
                      'fileURL') [0].textContent));
```

http://www.webex.com/schemas/2002/06/service/meeting

java:com.webex.service.binding.meeting.Recordings

method: 'GET'

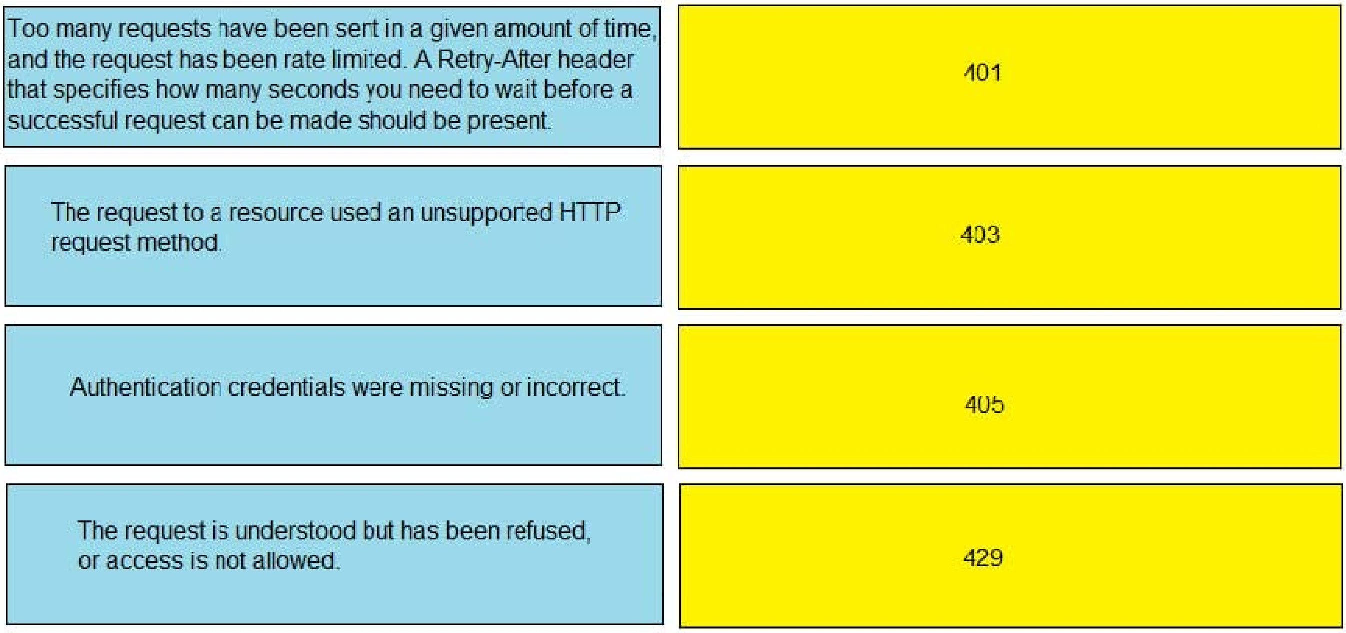
https://api.webex.com/v1/recordings

QUESTION 5

DRAG DROP

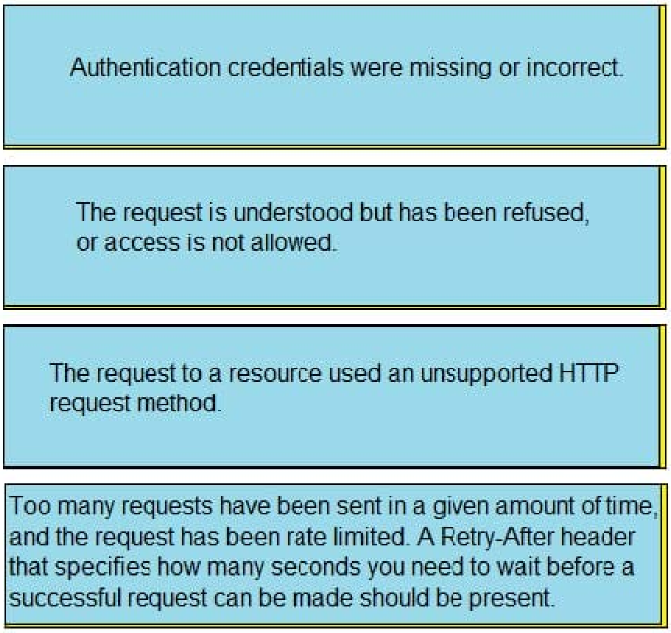
Drag and drop the definitions from the left onto the correct Webex Teams REST API HTTP response status codes on the right.

Select and Place:



Too many requests have been sent in a given amount of time, and the request has been rate limited. A Retry-After header that specifies how many seconds you need to wait before a successful request can be made should be present.	401
The request to a resource used an unsupported HTTP request method.	403
Authentication credentials were missing or incorrect.	405
The request is understood but has been refused, or access is not allowed.	429

Correct Answer:



Authentication credentials were missing or incorrect.

The request is understood but has been refused, or access is not allowed.

The request to a resource used an unsupported HTTP request method.

Too many requests have been sent in a given amount of time, and the request has been rate limited. A Retry-After header that specifies how many seconds you need to wait before a successful request can be made should be present.

Reference: https://developer.webex.com/docs/api/basics