

# PrepAwayExam

PrepAwayExam

> Contact Us

Login / Register

Search...



HOME

ALL VENDORS

★ GUARANTEE

? FAQ

TESTIMONIALS

CART (0)

## Pass Your Next Certification Exam Fast!

Everything you need to prepare, learn & pass your certification exam easily.

365 days free updates. First attempt guaranteed success.

Try **Online Engine** before you buy

### Instant Download



After Payment, our system will send you the products you purchase in mailbox in a minute after payment. If not received within 2 hours, please contact us.

### 365 Days Free Updates



Free update is available within 365 days after your purchase. After 365 days, you will get 50% discounts for updating.



### Money Back Guarantee

Full refund if you fail the corresponding exam in 60 days after purchasing. And Free get any another product.



### Security & Privacy

We respect customer privacy. We use McAfee's security service to provide you with utmost security for your personal information & peace of mind.

<http://www.prepawayexam.com/>

High-efficient Exam Materials are the best high pass-rate Exam Dumps

**Exam** : **70-482**

**Title** : Advanced Windows Store App  
Dev using HTML5 and  
JavaScript

**Vendor** : Microsoft

**Version** : DEMO

NO.1 You need to enable the loading of the daily schedule when the inspector launches the application.

Which class should you use?

- A. BackgroundTaskBuilder
- B. BackgroundExecutionManager
- C. BackgroundTaskDeferral
- D. SystemCondition

**Answer:** B

Reference:

<http://msdn.microsoft.com/enus/library/windows/apps/windows.applicationmodel.background.backgroundexecutionmanager.requestaccessasync.aspx>

NO.2 You need to find out whether the app is still in trial mode.

Which property should you check in the app startup code?

- A. Windows.ApplicationModel.Store.CurrentApp.licenseInformation.isActive
- B. Windows.ApplicationModel.Store.CurrentApp.licenseInformation.isTrial
- C. Windows.ApplicationModel.Store.CurrentApp.licenseInformation.expiration.late
- D. Windows.ApplicationModel.Store.CurrentApp.licenseInformation.productLicenses

**Answer:** C

NO.3 You need to implement the Search features for the app.

What should you do?

- A. Add the app content to the Windows index.
- B. Map a KnownFolder property for the search files location.
- C. Modify the NeighboringFileQuery property.
- D. Implement a default Bing Search control.

**Answer:** A

NO.4 DRAG DROP

You are developing a Windows Store app by using HTML5 and JavaScript. The app will be used in multiple geographic regions.

The default.html file contains the following code segment: `<input type="text" placeholder="USA"/>`  
`<br/>`

`<u!>`

`<li>Currency:USD</li> </ul>`

The resources.resjson file contains the English (en-US) resources shown in the following code segment:

```
{  
  "Country" : "USA",  
  "ListElement1" : "Currency:USD"  
}
```

The app must access string resources from the resources.resjson file. The data-win-res attribute must be configured in the default.html file to globalize the app.

You need to modify the code segment to ensure that the app can be localized with minimum effort. How should you complete the relevant code? (To answer, drag the appropriate code segments to the correct locations in the answer area. Each code segment may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to view content.)

The screenshot shows a code editor interface. On the left, there is a list of code segments:
 

- 'Country'
- 'ListElement1'
- placeholder: 'Country'
- textContent: 'ListElement1'
- 'USA'
- 'Currency:USD'

 On the right, the code snippet is:
 

```
<input type="text" placeholder=""
  data-win-res="{ [ ] }"/>
<br/>
<ul>
  <li data-win-res="{ [ ] }"></li>
</ul>
```

**Answer:**

The screenshot shows the same code editor interface as above, but with the answer applied. The segments on the left are highlighted in green:
 

- 'Country'
- 'ListElement1'
- placeholder: 'Country'
- textContent: 'ListElement1'
- 'USA'
- 'Currency:USD'

 The code snippet on the right now has the placeholders filled:
 

```
<input type="text" placeholder=""
  data-win-res="{ placeholder: 'Country' }"/>
<br/>
<ul>
  <li data-win-res="{ textContent: 'ListElement1' }"></li>
</ul>
```

#### NO.5 HOTSPOT

You develop a Windows Store app by using JavaScript. The app displays a list of video controls. The user can add video controls to the list.

You need to animate the video control list when the user adds a control to the list.

How should you complete the relevant code? (To answer, select the correct code segment from each drop-down list in the answer area.)

The screenshot shows a code editor with a JavaScript function:
 

```
function addToList() {
  var affectedItems = document.querySelectorAll(".listItem");
  var newItem = document.createElement("video");
  newItem.className = "listItem";
  [ ]
  (newItem, affectedItems);
  list.insertBefore(newItem, list.firstChild);
  [ ]
}
```

 There are two drop-down menus, one above the empty line and one below the empty line, for selecting code segments to complete the function.

## Work Area

```
function addToList() {
    var affectedItems = document.querySelectorAll(".listItem");
    var newItem = document.createElement("video");
    newItem.className = "listItem";

    addToList.beginAnimation();
    addToList.execute();
    list.refresh();
    var addToList = WinJS.UI.Animation.AddToListAnimation
    var addToList = WinJS.UI.Animation.createAddToListAnimation
        (newItem, affectedItems);
    list.insertBefore(newItem, list.firstChild);

    addToList.beginAnimation();
    addToList.execute();
    list.refresh();
    var addToList = WinJS.UI.Animation.AddToListAnimation
    var addToList = WinJS.UI.Animation.createAddToListAnimation

}
}
```

**Answer:**

## Work Area

```
function addToList() {
    var affectedItems = document.querySelectorAll(".listItem");
    var newItem = document.createElement("video");
    newItem.className = "listItem";

    addToList.beginAnimation();
    addToList.execute();
    list.refresh();
    var addToList = WinJS.UI.Animation.AddToListAnimation
    var addToList = WinJS.UI.Animation.createAddToListAnimation

        (newItem, affectedItems);
    list.insertBefore(newItem, list.firstChild);

    addToList.beginAnimation();
    addToList.execute();
    list.refresh();
    var addToList = WinJS.UI.Animation.AddToListAnimation
    var addToList = WinJS.UI.Animation.createAddToListAnimation

}
}
```

NO.6 You need to implement the database polling.  
Which code segment should you insert at line BG14?



## Work Area

```
function encryptString(string, encoding, protectionDescriptor) {
    var dataProtectionProvider = new Windows.Security
        .Cryptography.DataProtection
        .DataProtectionProvider(protectionDescriptor);
    var dataBuffer = Windows.Security
        .Cryptography.CryptographicBuffer
        .convertStringToBinary(string, encoding);
```

```
dataProtectionProvider.encryptAsync(dataBuffer).done(
dataProtectionProvider.encryptBase64Async(dataBuffer).done(
} dataProtectionProvider.protectAsync(dataBuffer).done(
dataProtectionProvider.protectAsync(dataBuffer).encrypt(
```

NO.8 You are creating a Windows Store app by using JavaScript. The app includes the following code segment:

```
01 function openChannel() {
02     var channel;
03     var chanOpt = Windows.Networking.PushNotifications
04         .PushNotificationChannelManager
05
06     return chanOpt.then (function (chan) {
07         channel = chan;
08     },
09     errorHandler)
10 }
```

You need to ensure that the app can receive push notifications from a calling app.

Which code segment should you insert at line 05?

- A. `.createPushNotificationChannelForSecondaryTileAsync()` ;
- B. `.createPuahNotificaticnChannelForApplicationAsync(id)`;
- C. `.createPushNotiflcacionChannelForApplicationAsync()`;
- D. `.createPushNotif icationChannelForSecondaryTileAsync (id)`;

**Answer:** C

NO.9 You need to separate the business and complex logic into components.

Which actions should you perform? (Each correct answer presents part of the solution. Choose all that apply.)

- A. In the JavaScript code, register the handler for the extension/mime-type.
- B. In the package.appxmanifest file, create an Extensions section and register the component .dll file.
- C. In Windows Explorer, drag the component .dll file to the project bin directory.
- D. In Microsoft Visual Studio Solution Explorer, right-click the References folder and then click Scope to This.

**Answer:** A,B

Explanation:

A: In Microsoft Internet Explorer 4.0 and later, MIME type determination occurs in URL monikers through the FindMimeFromData method. Determining the MIME type allows URL monikers and

other components to find and launch the correct object server or application to handle the associated content

B: \*An application that registers a background task needs to declare the feature in the application manifest as an extension, as well as the events that will trigger the task. If you forget these steps, the registration will fail. There is no <Extensions> section in the application manifest of the Microsoft Visual Studio standard template by default, so you need to insert it as a child of the Application tag.

\*You can implement Windows RunTime components for your apps, but you must register those components with the operating system for them to run correctly. To register a Windows RunTime component, you must put the registration information in the WinMD files and in the app manifest. If a project implements a Windows RunTime component, the build output of the project will contain a WinMD file. Visual Studio extracts the Windows RunTime registration information from the WinMD file and generates the appropriate Extensions elements in the app manifest. The system supports two forms of servers: .dll servers (in-process) and .exe servers (outof-process). These servers require similar but different registration information that must be copied into the app manifest. Visual Studio supports generating manifest only for .dll servers, and the DLLServer extension is required to register .dll servers. The following values in the app manifest are taken from the WinMD files to construct the DLLServer Extension: DllPath ActivatableClassId ThreadingModel ActivatableClass (ActivatableClassId attribute) Here's an example of the output XML: <extension

```
category="Microsoft.Windows.ActivatableClass"> <dllServer> <dllPath>Fabrikam.dll</dllPath>
<activatableClass activatableClassId="Fabrikam.MyClass" threadingModel="sta" /> </dllServer>
</extension>
```

#### NO.10 HOTSPOT

You are developing a Windows Store app. The app contains the following HTML markup segment:

```
<video id="videoplayer" style="width: 627px; height: 305px;" src="/Navaul.asp" controls=""></video>
<div id="messageDiv"></div>
```

The app must stream video to available devices by using the Play to feature.

You need to implement the Play to contract for the app.

How should you complete the relevant code? To answer, select the appropriate code segments from the lists in the answer area.

```
var pt = Windows.Media.PlayTo. 
pt.addEventListener("sourcerequested", sourceRequestHandler, false);
function sourceRequestHandler(e) {
    try {
        e.sourceRequest.setSource(document.
            getElementById("videoplayer"). 
    } catch (ex) {
        id("messageDiv").innerHTML += "An error occurred";
    }
}
```

```

var pt = Windows.Media.PlayTo.
pt.addEventListener("sourcerequested", sourceRequestHandler, false);

function sourceRequestHandler(e) {
  try {
    e.sourceRequest.setSource(document.
      getElementById("videoplayer").
  } catch (ex) {
    id("messageDiv").innerHTML += "An error occurred";
  }
}

```

**Answer:**

```

var pt = Windows.Media.PlayTo.
pt.addEventListener("sourcerequested", sourceRequestHandler, false);

function sourceRequestHandler(e) {
  try {
    e.sourceRequest.setSource(document.
      getElementById("videoplayer").
  } catch (ex) {
    id("messageDiv").innerHTML += "An error occurred";
  }
}

```

NO.11 You are developing a Windows Store app by using JavaScript. The app contains a custom C# Windows Runtime Metadata (WinMD) component.

You receive unexpected results when you run the app.

You need to ensure that you can debug the WinMD component while running the JavaScript app.

What should you do?

- A. In the JavaScript project, set the Debug Type to Mixed (Managed and Native).
- B. Enable Just-In-Time debugging for all types of code.
- C. In the JavaScript project, change the Debug Type to Native with Script.
- D. In the C# project, set the Debug Type to Mixed (Managed and Native).

**Answer: A**

NO.12 You are developing a Windows Store app that will access a device's webcam.

The app will use a custom control panel to enable camera modifications.

You need to specify that the app will use the custom control panel.

What should you do in the Visual Studio IDE?

A. In the Extension Manager, set a reference to the Windows.Devices.Enumeration.winmd file.

B. On the Capabilities tab of the Manifest Designer, select the Webcam check box.

C. On the Declarations tab of the Manifest Designer, choose Camera Settings.

D. In the Reference Manager, set a reference to the Windows.Devices.Sensors.winmd file.

**Answer:** C