


Latest Braindump2go 70-481 E-Book PDF Free Download (1-10)

MICROSOFT OFFICIAL: New Updated 70-481 Exam Questions from Braindump2go 70-481 PDF Dumps and 70-481 VCE Dumps! Welcome to Download the Newest Braindump2go 70-481 VCE&PDF Dumps:

<http://www.braindump2go.com/70-481.html> (122 Q&As) We never believe in second chances and Braindump2go brings you the best 70-481 Exam Preparation Materials which will make you pass in the first attempt. We guarantee all questions and answers in our 70-481 Dumps are the latest released, we check all exam dumps questions from time to time according to Microsoft Official Center, in order to guarantee you can read the latest questions! Exam Code: 70-481 Exam Name: Essentials of Developing Windows Store Apps Using HTML5 and JavaScript Certification Provider: Microsoft Corresponding Certifications: MCSD, MCSD: Windows Store Apps Keywords: 70-481 Dumps, 70-481 Free Exam Questions, 70-481 VCE, 70-481 Dumps PDF, 70-481 Study Guide, 70-481 Essentials of Developing Windows Store Apps Using HTML5 and JavaScript

Microsoft Essentials of Developing Windows Store Apps Using HTML5 and JavaScript: 70-481



Product Description Exam Number/Code: 70-481

Exam Number/Code: 70-481

"Microsoft Essentials of Developing Windows Store Apps using HTML5 and JavaScript" known as 70-481 exam, is a Microsoft Certification. With the complete questions and answers, Braindump2go has assembled to take you through 122 Exam preparation. In the 70-481 exam resources, you will cover everything you need to know for Microsoft Others Microsoft Certification helping to ready you for your Microsoft Certification.

Questions and Answers : 122 Q&As

Updated: Sep 14, 2015

~~\$129.99~~ **\$99.99**

[PDF DEMO](#)

[CHECK OUT](#)

Printable PDF Premium VCE + VCE Simulator

Case Study 1 - Southbridge Videos (QUESTION 1 - QUESTION 15) Background Southbridge Video rents and sells video games, movies, and other multimedia content. The company is currently expanding its operations. Southbridge video is developing a Windows Store news reader and social media app. Users will be able to view and interact with news stories that are related to the video game, entertainment, and related industries. Southbridge Video employees will also use the app to manage sales leads, order fulfillment, and customer-service related activities. Business Requirements The app has the following requirements. Email Users must be able to email news items to their contacts. Navigation- Easy navigation access must be available to move to the next and previous pages. - Users searching for specific news items must be able to navigate directly to an item from the search results pane. News items - When a user selects a news item on newsPage.html, the news item should be displayed in the newsItem.html page. - The app must periodically retrieve new news items on the newsPage.html page. - The app must allow the user to save annotated news. - The app must display a short title for each news item. Search capabilities- The app must allow users to search within available news items to locate articles of interest. - The app must provide search suggestions for users. Authentication- Employees must be able to view the sales leads, order fulfillment, and customer- service related pages. - Authenticated users must be able to save news items to read at a later time. - Authenticated users must be able to add and save handwritten notes to the news items. - The app must allow users to choose whether their credentials will be saved after they first log in. User credentials must be saved by default. Technical Requirements Email- When a user clicks the To button to email news items, only contacts that have email addresses should be displayed. - When multiple contacts are selected, the email addresses must be displayed in the To text box as a semi-colon-delimited string. News items The app must periodically retrieve new items from a web service. The app must display a progress indicator while retrieving new items. The news feed service must respond to all requests within 15 seconds. The app must not throw an error if the service does not respond within the allotted time. Titles for news items must not overflow into other elements on the page. Authenticated users must be able to add handwritten notes to the news items by using a stylus or the mouse. The app must support saving annotations along with the news item. The app must cancel the navigation from newsItem.html to newsPage.html if there are unsaved changes in the annotations area. Search capabilities The app must retrieve a list of phrases from the news service and provide them as search suggestions to users. Authentication- Users connected to the corporate network must authenticate against an Active Directory server. - Users outside of the corporate network must authenticate by using forms-based authentication. - User credentials

must not be transmitted in plain text. Application Structure Relevant portions of the app files are shown below. (Line numbers in the code segments are included for reference only and include a two-character prefix that denotes the specific file to which they belong.)

```
emailPage.html
EH01 <body>
EH02 ...
EH03 <div>
EH04 <buttonid="BtnTo">To</button>
EH05 <inputtype="text" id="To"/>
EH06 ...
EH07 </div>
EH08 </body>

emailPage.js
EJ01 function pickContactAsync() {
EJ02 var picker = Windows.ApplicationModel.Contacts.ContactPicker();
EJ03
EJ04 picker.pickSingleContactAsync().then(function (contact) {
EJ05 });
EJ06 }

getCredential.js
GC01 function launchCredPicker() {
GC02 try {
GC03 var options = new Windows.Security.Credentials.UI.CredentialPickerOptions();
GC04
```

```
getCredential.js
GC01 function launchCredPicker() {
GC02 try {
GC03 var options = new Windows.Security.Credentials.UI.CredentialPickerOptions();
GC04
GC05 Windows.Security.Credentials.UI.CredentialPicker.pickAsync(options)
GC06 .then(function (results) {
GC07 ...
GC08 });
GC09 catch (err) {
GC10 WinJS.log && WinJS.log("Error message: " + err.message, "sample", "error");
GC11 }
GC12 }

newsItem.html
NI01 <body>
NI02 ...
NI03 <div>
NI04 <canvasid="inkCanvas"></canvas>
NI05 <div>
NI06 <buttonid="draw">Draw</button>
NI07 <buttonid="select">Select</button>
NI08 <buttonid="erase">Erase</button>
NI09 </div>
NI10 </div>
```

```
newsItem.js
NI01 function isInkInput(evt)
NI02 {
NI03
NI04 }
NI05
NI06 function drawStrokes() {
NI07
NI08 strokeColor = "black";
NI09 strokeWidth = 2;
NI10 }
NI11
NI12 function selectStrokes() {
NI13
NI14 }
NI15 }

newsPage.html
NH01 <body>
NH02 ...
NH03 <divclass="newsitemtemplate" data-win-control="WinJS.Binding.Template">
NH04 <imgclass="news-item-image" src="#" data-win-bind="src: backgroundImage; alt:
title" />
NH05 <divclass="news-item-info">
NH06
NH07 <h4class="news-item-description" data-win-bind="textContent: description"></h4>
NH09 </div>
NH10 </div>
NH11 </body>
```

```
newsPage.js
NP01 var nav = WinJS.Navigation;
NP02 ItemInvoked: function (eventObject) {
NP03 var item = data.items.getAt(eventObject.detail.itemIndex);
NP04
NP05 }
NP06 },
NP07
NP08 function provideSuggestions(queryText, suggestionRequest) {
NP09 var suggestionList = getSuggestions();
NP10 var query = queryText.toLowerCase();
NP11
NP12 }
NP13 }
NP14 }
NP15 }
NP16
NP17 function OnActivated(e) {
NP18 var detail = e.detail;
NP19
NP20 WinJS.Navigation.navigate("/html/NewsItem.html", id);
NP21 }
NP22 }
```

QUESTION 1 You need to configure the CredentialPicker object to meet the requirements. Which code segment should you insert at line GC04? A. options.callerSavesCredential = false; B. options.CredentialSaveOption.unselected; C.

options.CredentialSaveOption.selected; D. options.callerSavesCredential = true; E. options.CredentialSaveOption.hidden;

Answer: C QUESTION 2 You need to ensure that the navigation requirements for displaying news items are met. Which code segment should you insert at line NP04? A. nav.navigate("/html/newsPage.html"); B. nav.navigate("/html/newsItem.html", {item: item}); C. nav.forward("/html/newsItem.html"); D. nav.forward("/html/newsPage.html", {item: item}); Answer: B

QUESTION 3 You need to implement the code to retrieve news items according to the requirements. Which code segment should you add to newsPage.js?

```
A. function fetchDailyNews() {
  WinJS.xhr({ url: "http://www.southridgevideo.com/feed" })
  .then(function progress(result) {
    progressDiv.innerText = result.readyState;
    function complete(result) {
      ...
    }
  });
}

B. function fetchDailyNews() {
  WinJS.xhr({ url: "http://www.southridgevideo.com/feed" })
  .then(function progress(result) {
    progressDiv.innerText = result.readyState;
  })
  .then(function complete(result) {
    ...
  });
}

C. function fetchDailyNews() {
  WinJS.xhr({ url: "http://www.southridgevideo.com/feed" })
  .done(function progress(result) {
    progressDiv.innerText = result.readyState;
  });
}

D. function fetchDailyNews() {
  WinJS.xhr({ url: "http://www.southridgevideo.com/feed" })
  .done(function complete(result) {
    ...
  }), null,
  function progress(result) {
    progressDiv.innerText = result.readyState;
  });
}
```

A. Option AB. Option BC. Option CD. Option D Answer: D QUESTION 4 You need to implement the code to meet the search requirements. Which code segment should you insert at line NP11?

```
A. for (var i = 0, len = suggestionList.length; i < len; i++) {
  if (suggestionList[i] === query) {
    suggestionRequest.searchSuggestionCollection.appendQuerySuggestion(
      suggestionList[i]);
  }
}

B. for (var i = 0, len = suggestionList.length; i < len; i++) {
  if (suggestionList[i] === query) {
    suggestionRequest.searchSuggestionCollection.appendResultSuggestion(
      suggestionList[i]);
  }
}

C. for (var i = 0, len = suggestionList.length; i < len; i++) {
  if (suggestionList[i].substr(0, query.length).toLowerCase() === query) {
    suggestionRequest.searchSuggestionCollection.appendResultSuggestion(
      suggestionList[i]);
  }
}

D. for (var i = 0, len = suggestionList.length; i < len; i++) {
  if (suggestionList[i].substr(0, query.length).toLowerCase() === query) {
    suggestionRequest.searchSuggestionCollection.appendQuerySuggestion(
      suggestionList[i]);
  }
}
```

A. Option AB. Option BC. Option CD. Option D Answer: D QUESTION 5 You need to ensure that the app displays only contacts that meet the requirements. Which code segment should you insert at line EJ03?

```
A. picker.selectionMode =
  Windows.ApplicationModel.Contacts.ContactSelectionMode.fields;

B. picker.selectionMode =
  Windows.ApplicationModel.Contacts.ContactSelectionMode.contacts;

C. picker.desiredFields.append(Windows.ApplicationModel.Contacts.ContactField(
  "Email", string));

D. picker.desiredFields.append(
  Windows.ApplicationModel.Contacts.KnownContactField.email);
```

A. Option AB. Option BC. Option CD. Option D Answer: D QUESTION 6 You need to ensure that the user can annotate news items according to the requirements. Which code segment should you insert at line NJ03?

```
A. return (pointerDeviceType === "Pen" || (pointerDeviceType === "Mouse" && evt.button === 0));

B. return (pointerDeviceType === "Pen" || (pointerDeviceType === "Mouse" && evt.button === 1));

C. return (pointerDeviceType === "Pen" || (pointerDeviceType === "Mouse" && evt.type === "click"));

D. return (pointerDeviceType === "Pen" || (pointerDeviceType === "Mouse" && evt.type === "mousedown"));
```

A. Option AB. Option BC. Option CD. Option D Answer: B QUESTION 7 You need to implement the code to meet the requirements for handling changes in the annotation area of the newsItem.html page. Which event handler should you add to newsItem.js?

```
Ⓐ WinJS.Navigation.addEventListener("beforenavigate", onbeforeNavigate);
function onbeforeNavigate(eventObject) {
  if (hasChanges) {
    eventObject.detail.setPromise(WinJS.Promise.wrap(true));
  }
  else {
    eventObject.detail.setPromise(WinJS.Promise.wrap(false));
  }
}

Ⓑ WinJS.Navigation.addEventListener("navigating", onNavigating);
function onNavigating(eventObject) {
  if (hasChanges) {
    eventObject.detail.setPromise(WinJS.Promise.wrap(true));
  }
  eventObject.detail.setPromise(WinJS.Promise.wrap(false));
}

Ⓒ WinJS.Navigation.addEventListener("beforenavigate", onbeforeNavigate);
function onbeforeNavigate(eventObject) {
  eventObject.detail = null;
}

Ⓓ WinJS.Navigation.addEventListener("navigating", onNavigating);
function onNavigating(eventObject) {
  eventObject.detail = null;
}
```

A. Option AB. Option BC. Option CD. Option D Answer: A QUESTION 8 You add an element with an ID of MyFlexbox to newsPage.html. You need to ensure that the layout for MyFlexbox meets the requirements. Which layout should you choose? A.

```
Ⓐ #MyFlexbox {
  display: flex;
  border: 1px solid black;
  -ms-flex-direction: column;
}

Ⓑ #MyFlexbox {
  display: flex;
  border: 1px solid black;
  -ms-flex-direction: row;
}

Ⓒ #MyFlexbox {
  display: flex;
  border: 1px solid black;
  -ms-flex-direction: row;
}

Ⓓ #MyFlexbox {
  display: flex;
  border: 1px solid black;
  -ms-flex-direction: column;
}
```

A. Option AB. Option BC. Option CD. Option D Answer: A QUESTION 9 You need to implement the code to meet the requirements for displaying content from search results. Which code segment should you insert at line NP19? A

```
Ⓐ if (detail.kind === Windows.ApplicationModel.Activation.SearchResults) {
  var id = FindNewsId(detail.arguments);
}

Ⓑ if (detail.kind === Windows.ApplicationModel.Activation.SearchResults) {
  var id = FindNewsId(detail.queryText);
}


Ⓒ if (detail.kind === Windows.ApplicationModel.Activation.SearchResults) {
  var id = FindNewsId(detail.queryText);
}

Ⓓ if (detail.kind === Windows.ApplicationModel.Activation.SearchResults) {
  var id = FindNewsId(detail.arguments);
}
```

A. Option AB. Option BC. Option CD. Option D Answer: B QUESTION 10 You need to ensure that the requirements for capturing user input on the newsItem.html page are met. Which code segments should you use? (Each correct answer presents part of the solution. Choose all that apply.) A. InkManager.mode = Windows.UI.Input.Inking.InkManipulationMode.selecting; B. InkManager.mode = Windows.UI.Input.Inking.InkRecognitionTarget.selected; C. InkManager.mode = Windows.UI.Input.Inking.InkManipulationMode.inking; D. InkManager.mode = Windows.UI.Input.Inking.InkRecognitionTarget.all; Answer: AB All the 122 Questions and Answers in Braindump2go 70-481

Exam Dumps are the latest 70-481 Real Exam Questions not just 70-481 Practice Tests Questions! Braindump2go Microsoft 70-481 Exam Dumps PDF&VCE Guarantees you 100% Pass 70-481 Exam! Braindump2go Can Provide the Latest 70-481 Dumps Questions from Microsoft Official Exam Center for You!

Microsoft Essentials of Developing Windows Store Apps using HTML5 and JavaScript: 70-481



Product Description Exam Number/Code: 70-481

Exam Number/Code: 70-481

"Microsoft Essentials of Developing Windows Store Apps using HTML5 and JavaScript", also known as 70-481 exam, is a Microsoft Certification. With the complete collection of questions and answers, Braindump2go has assembled to take you through 122 Q&As to your 70-481 Exam preparation. In the 70-481 exam resources, you will cover every field and category in Microsoft Others Microsoft Certification helping to ready you for your successful Microsoft Certification.

Questions and Answers : 122 Q&As

Updated: Sep 14, 2015

~~\$129.99~~ **\$99.99**

[PDF DEMO](#)

[CHECK OUT](#)

Printable PDF **Premium VCE + VCE Simulator**

Free Demo Download

Braindump2go offers free demo for 70-481 exam (Microsoft Essentials of Developing Windows Store Apps using HTML5 and JavaScript). You can check out the interface, question quality and usability of our practice exams before you decide to buy it.

FREE DOWNLOAD: Latest 70-481 PDF Dumps & VCE Dumps from Braindump2go:
[http://www.braindump2go.com/70-481.html\(122Q&As\)](http://www.braindump2go.com/70-481.html(122Q&As))