IBM

Exam C5050-408

IBM Worklight Foundation V6.2, Mobile Application Development

Verson: Demo

[Total Questions: 10]

Question No : 1

The application developer wants to use the Mobile Browser Simulator to preview and test Worklight applications.

Which of the following application environments can the application developer preview on the Mobile Browser Simulator?

A. Android, iPhone, iPad, and Adobe Air
B. Android, iPhone, iPad, Windows Phone 8 and Desktop Browser web page
C. Android, iPhone, iPad, Blackberry 6 and 7, Blackberry 10, Windows Phone 8, and Mobile Web
D. Android, iPhone, iPad, Adobe Air, Blackberry 7, Blackberry 10, Windows Phone 8, and Windows 8 desktopand tablets

Answer: C

Explanation:

You can use the Mobile Browser Simulator to preview Worklight applications on Android, iPhone, iPad, BlackBerry 6 and 7, BlackBerry 10, Windows Phone 8, and mobile web application environments.
References:
http://www.ibm.com/support/knowledgecenter/SSZH4A_6.2.0/com.ibm.worklight.dev.doc/wl_studio_tools/ topics/cmbswl.html?lang=en

Question No : 2

An application developer has created a Worklight Adapter that retrieves data from a remote RSS feed and returns the results. The developer is not seeing the expected results in the browser. The developer decides to add debug logging to the code in order to determine why the expected results are not seen.

Where can the application developer locate the output from the debug logging?

In the:

A. server console of Eclipse.
B. Worklight DevTools extension for the Chrome browser.
C. Worklight Server's First Failure Data Capture (FFDC) log.
D. web browser by setting a breakpoint after the adapter call to inspect the result.

Answer: B

Explanation:

Chrome devTools – Remote debugging in Worklight.
In 2012 Chrome implemented an awesome feature called ‘Remote debugging’. It introduced proper remote debugging for WebKit browsers.
References: http://www.optis.be/ibm/worklight/worklight-hybrid-app-debugging-with-chrome-devtools-for-mobile/

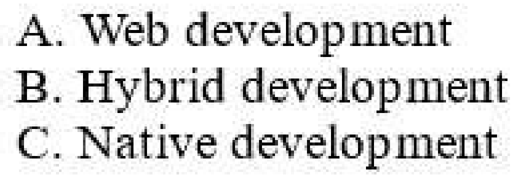
Question No : 3

Which characteristic(s) of an application map to the respective development approaches?

Characteristics of the Application

- The application is written for a specific platform and runs on that platform only.
- The application runs inside the browser of the mobile device, and uses standard technologies such asHTML5, CSS3, and JavaScript.
- The application runs inside a native container and uses the browser engine to display the applicationinterface.
- The application is not distributed through an application store.

Development Approach (see exhibit)



A. Web development
B. Hybrid development
C. Native development

A. 1-A, 2-B, 3-C, 4-B
B. 1-B, 2-C, 3-A, 4-C
C. 1-B, 2-B, 3-C, 4-C
D. 1-C, 2-A, 3-B, 4-A

Answer: D

Question No : 4

An application developer is working on a mobile application for a bank. The requirement is to globalize this application to display user messages, currency, date and time according to the locale and language settings on the device.

Which Worklight APIs can the application developer use to detect the specific device locale and language?

A. WL.Client.getLocale()WL.Client.getLanguage()
B. WL.Device.getLocale()WL.Device.getLanguage()
C. WL.App.getDeviceLocale()WL.App.getDeviceLanguage()
D. WL.G11N.getLocaleName()WL.G11N.getPrefferedLanguage()

Answer: D

Explanation:

The getLocaleName function gets the client current locale setting on the device. The getPreferredLanguage function gets the current language of the client.
Note: The following listings and figures show the Cordova globalization API. Dojo is used to demonstrate the user interface.
Table 2. Listing 1: Using the Cordova globalization API
function onDeviceReady(){
g11n = window.plugins.globalization;
}
References:
https://www.ibm.com/support/knowledgecenter/SSZH4A_6.1.0/com.ibm.worklight.dev.doc/dev/ c_globalization_in_ibm_worklight.html

Question No : 5

An application developer is developing a project in Worklight Studio that has one adapter and a hybrid application that has Android and iPhone environments.

When the application developer builds the application using the 'Build All environments' command, what are the complete set of artifacts created?

A. A .war file, a .adapter file and a .wlapp file for all environments.
B. A .war file, a .wlapp file for all environments and a .wlapp file for common resources.
C. A .war file, a .wlapp file for all environments, a .wlapp file for common resources, a .wlapp for eachenvironment and a native project for the target platform of each environment.
D. A .wlapp file for all environments, a .wlapp file for common resources, a .wlapp for each environment and anative project for the target platform of each environment.

Answer: B

Explanation:

- A WAR file is created by Worklight Studio for every Worklight project, regardless of the number of apps itcontains.
- If you build an entire app, a file called app-name.wlapp is created, containing the

code and resources of allenvironments that are supported by your app. For example: myApp-all.wlapp.

- If you build an app only for specific environments, a file called app-name-env-version.wlapp is created perenvironment. For example: myApp-android-1.0.wlapp.

References:

https://www.ibm.com/support/knowledgecenter/SSZH4A_6.1.0/com.ibm.worklight.deploy.doc/devref/ t_transporting_the_app.html

Question No : 6

The following is part of a HTTP adapter invocation result.

```
{
“college”:{
“class”:{
“description”: “Class object”
“student”: {
“description”: “Students marks details”,
“title”:Top Stories”
},
“studentmarks”: [
{
“name”: “Mirjana Lucic-Baroni.”,
“marks”: “90”,
“level”: “Level one”
}
{
“name”: “Rose Way.”,
“marks”: “87”,
“level”: “null”
}

]

}

} }
```

An application developer needs to send only the ‘name’ and ‘marks’ information to a mobile application using the adapter xsl file.

Which xsl file code snippet will enable the application developer to achieve this?

A. {

```
‘Items’: [
<xsl:for-each select=”//college”>
{
‘name: ‘<xsl:value-of select=”college.name”/>’,
‘marks’: ‘<xsl:value-of select=”college.marks”/>’,
},
</xsl:for-each>
]
}
```

B. {

```
‘Items’: [
<xsl:for-each select=”//studentmarks”>
{
‘name’: ‘<xsl:value-of select=”name”/>’,
‘marks’: ‘<xsl:value-of select=”marks”/>’,
},
</xsl:for-each>
]
}
```

C. {

```
‘Items’: [
<xsl:for-each select=”//item”>
{
‘name’: ‘<xsl:value-of select=”name”/>’,
‘marks’: ‘<xsl:value-of select=”marks”/>’,
},
</xsl:for-each>
]
}
```

D. {

```
‘Items’: [
<xsl:for-each select=”//student”>
{
```

```
‘name’: ‘<xsl:value-of select=”student.name”/>’,
‘marks’: ‘<xsl:value-of select=”student.marks”/>’,
},
</xsl:for-each>
]
}
```

Answer: B

Explanation:

For each studentmarks select the the name and marks.

Question No : 7

An application developer wants to use the Rich Page Editor to create and edit web pages for a hybrid application. The application developer wishes to use the Mobile Navigation view to manage the mobile views in the application.

Which action can the application developer perform inside the Mobile Navigation view?

A. Create a view by right clicking in the Mobile Navigation view and select Create.
B. Create a view by dragging a view object from the Palette view into the Mobile Navigation view.
C. Link a button to a mobile view by dragging a button from the Design view to a mobile view in the MobileNavigation view.
D. Link a button to a mobile view by dragging a button from the Palette view to the mobile view in the MobileNavigation view.

Answer: C

Explanation:
You can link widgets, such as buttons or list view items, to mobile views or pages. You can drag a widget from the Design view in Rich Page Editor to a mobile view or page in the Mobile Navigation view. You can also drag mobile views or pages from the Mobile Navigation view to widgets in the Design view within Rich Page Editor.
References:
http://www.ibm.com/support/knowledgecenter/SSHS8R_7.1.0/com.ibm.worklight.dev.doc/com.ibm.rad.worklight.doc/topics/cmobileviewswl.html

Question No : 8

An application developer needs to detect when the mobile device has lost network connectivity. To do so, the developer has added a listener for event WL.Events.WORKLIGHT_IS_DISCONNECTED.

When will the application developer be notified of the loss of connectivity if the heartbeat is not enabled?

A. After the event “resume” has been fired.
B. When WL.Client.connect is called and it fails to connect to the Worklight Server.
C. When WL.Client.getEnvironment is called and the device does not have network connectivity.
D. When WL.Device.getNetworkInfo is called and the device does not have network connectivity.

Answer: D

Explanation:

WL.Device.getNetworkInfo() – This method is available for the Android and iOS environments.
A callback must be specified as a function parameter. The callback receives an object with the following properties: isNetworkConnected – true/false isAirplaneMode – true/false carrierName – string. For example, AT&T or VERIZON. telephonyNetworkType – string. For example, UMTS or GPRS. isRoaming – true/false
networkConnectionType – mobile/WiFi ipAddress – string
References: https://developer.ibm.com/mobilefirstplatform/documentation/getting-started-7-0/data/working-offline/

Question No : 9

A mobile application developer intends to use JSONStore to encrypt local data of an application.

What does the application developer need to do to ensure data is encrypted?

When using:

A. WL.JSONStore.init(), pass a password in the options parameter.
B. WL.JSONStore.init(), pass what collections will be encrypted in the options parameter.
C. WL.JSONStore.documentify(), pass parameter encryption set to true.
D. WL.JSONStore.encrypt(), pass a user name and a password in the options parameter.

Answer: B

Explanation:

WL.JSONStore.init
{Promise} init(collections, options) Starts one or more collections.
Option parameters include options.password Optional - Password that is used to secure the contents of the store, by default there is no data encryption.
References:
https://www.ibm.com/support/knowledgecenter/SSHS8R_6.3.0/com.ibm.worklight.apiref.doc/html/refjavascriptclient/html/WL.JSONStore.html%23init

Question No : 10

An application developer wants to use the IBM Mobile Test Workbench to automate the creation, execution, and analysis of functional tests for a Worklight hybrid application on Android devices. The application developer has successfully installed and configured the IBM Mobile Test Workbench. After creating a test project, the application developer is now ready to generate a test from recording a session on the Android device.

After selecting the application in the list of managed applications available in the Test Workbench Android client view, what must the application developer do to generate the test?

A. 1. Click Record in the mobile client on the device
Interact with the device as needed
Tap the Back button
B. 1. Click Record in the mobile client on the device
Interact with the device as needed.
Click Stop in the mobile client on the device
C. 1. Click Install to install the application on the device
Click Record in the mobile client on the device
Interact with the device as needed
Close the application
D. 1. Click Install to install the application on the device
Click Record in the mobile client on the device
Interact with the device as needed
Click Stop in the mobile client on the device

Answer: D
Explanation:

The goal of mobile testing is to ensure that your mobile application meets the requirements that guided its design and development. To help you meet this goal, IBM Mobile Test

Workbench for Worklight implements the following stages in the testing process: Configuration: Set up your test environment with IBM Mobile Test Workbench for Worklight and the SDKs for the mobile operating systems. Install the mobile test client on one or several mobile devices. Ensure that the mobile devices have connectivity through WiFi, 3G, or 4G, and add those devices to the test workbench. * Application preparation: Import the application that you want to test into the test workbench, or use the device to upload the application under test to the test workbench.

- Test recording: Run the app from the mobile test client to start a recording. The recorder records all userinteractions, sensor inputs, and application behavior, and uploads the recorded data to the test workbench, where it can be converted into a mobile test.
- Test editing: After recording, you can edit the test in the natural language editor. You can use the mobile dataview to display and select UI elements from the recorded applications. You can replace recorded test values with variable test data, or add dynamic data to the test.
- Testing: You can deploy and run automated tests on multiple devices to ensure that the app matches theexpected behavior that is defined in verification points. During the run, each verification point is checked and receives a pass, fail, or inconclusive status and functional data is recorded.

Evaluation of results:

References:

https://www.ibm.com/support/knowledgecenter/SSZH4A_6.1.0/com.ibm.worklight.mobtest.doc/topics/ c_wl_mobile_test_ovw.html