

Microsoft

*70-357
Developing Mobile Apps*

**For More Information – Visit link below:
<http://www.certsgrade.com/>**

Version = Product



Case Study: 1

Background

Business requirements

In this section, you will see one or more sets of questions with the same scenario and problem. Each question presents a unique solution to the problem, and you must determine whether the solution meets the slated goals. Any of the solutions might solve the problem. It is also possible that none o' the solutions solve the problem.

Technical requirements

Application structure

Once you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appe.ir in the review screen.

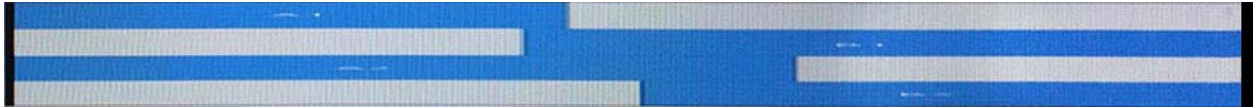
The timeline element of the app has the following layout requirements:

- the timeline must adapt to the screen size and orientation of the device.
- The timeline size must dynamically change if the window containing the content is resized by the user.
- The user must be able to scroll through the timeline horizontally when the device is in landscape mode.
- The user must be able to scroll through the timeline vertically when the device is in portrait mode.
- The timeline must begin scrolling as soon as a scroll is detected. Scrolling must continue for a short distance after the scroll input has stopped.
- Scroll bars or panning controls must always be visible.

The following image depicts the layout for the timeline section of the app when the device is using landscape orientation:



the following image depicts the layout tor the timeline section of the app when the device is using portrait orientation:



The content element of the app has the following layout requirements:

When a user selects an item on the timeline, the details for that item must display beneath or to the right of the timeline

- The content section must display one page of information. The element must be a child of the selected item in the timeline.

- Users must be able to return to a previously selected event by pressing the Back button.

the user must be able to navigate the application using the interface below:



- The Favorite button mark the Current content to be displayed in a Favorites panel.
- The Back and Forward buttons navigate through the app selection history. Both buttons must be available on all devices.
- The Note button allows the user to manage notes about the current content.
- The app must support touch, mouse, and stylus input.
- The app layout must automatically adapt to the screen size and orientation.

Layout Requirement:

You identify the following layout requirements:

General

- All user interface (UI) elements must continuously scale when a user resizes the window.
- UI controls must be smaller and spaced closer together if there is a mouse or stylus available.
- UI controls must be larger and spaced farther apart if the device supports touch and there is no mouse or pointer available.

Timeline

- The timeline must be displayed in a horizontal layout when the device is in a landscape orientation or when the horizontal width is greater than the vertical height.
- The timeline must be displayed in a vertical layout when the device is in a portrait orientation or when the vertical height is greater than the horizontal width.
- Each item in the past must be linked to the next item in the future.

- User must be able to scroll from past events to future events 01 from future events to past events.
- The app must only allow one level of detail to be linked to each item in the timeline

New Tab:

You must optimize the app using the following guidelines:

- You must minimize the time it takes to display content when an item on the timeline is selected.
- The app must respect memory and resource constraints for all devices.

XML coding style:

All code and markup must conform to the following style guidelines:

- Use resource dictionaries for styles that are used more than once.
- Limit the use of nested panels.
- Use built-in properties of existing panels instead of using separate style objects.
- Use the navigation structure that best models the data without exceeding the requirements of the app.

MainPage.xaml

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.)

```

MP01 <Page
    x: Class = "_70357rm.MainPage"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:local="using:_70357rm"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:m="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc: Ignorable = "d">
MP02     <Grid Background="{ThemeResource ApplicationPageBackgroundThemeBrush}">
MP03
MP04         <RelativePanel BorderBrush ="Gray" BorderThickness ="10">
MP05             <Rectangle x :Name="A1" Fill ="Red" MinHeight="200" MinWidth ="400"
                RelativePanel.AlignLeftWithPanel ="True"
                RelativePanel.AlignTopWithPanel ="False" />
MP06             <Rectangle x:Name ="B1" Fill="Blue" MinHeight="200" MinWidth ="400"
                RelativePanel.Below ="A1"
                RelativePanel.RightOf = ""
                RelativePanel.AlignRightWithPanel ="True"
                RelativePanel.AlignBottomWithPanel ="False" />
MP07         </ RelativePanel>
MP08     </ Grid>
MP09 </ Page>

```

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.)

```

AS01 <Page
    x: Class = "_70357rm.Settings"
    xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
    xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
    xmlns:local = "using:_70357rm"
    xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
    xmlns:m="http://schemas.openxmlformats.org/markup-compatibility/2006"
    mc:Ignorable="d">
AS02     <Grid Background="AliceBlue">
AS03         <Border BorderBrush="DarkBlue" BorderThickness="5" />
AS04         <Grid Margin="5 5 5 5">
AS05             <StackPanel HorizontalAlignment="Center">
AS06                 <TextBlock Text="Date Settings" Foreground="DarkBlue" FontFamily="Arial"
FontSize="20" FontStyle="Normal"
                    FontWeight="Bold" Margin="0 5 0 5" HorizontalAlignment="Center" />
AS07                 <StackPanel Orientation="Horizontal">
AS08                     <CheckBox Content="Center on Date" FontFamily="Arial" FontSize="14"
FontStyle="Normal" Margin="20,0,0,0"/>
AS09                     <CheckBox Content="Set Start Date" FontFamily="Arial" FontSize="14"
FontStyle="Normal" Margin="20,0,0,0"/>
AS10                 </StackPanel>
AS11                 <TextBlock Text="Start Date" Foreground="DarkBlue" FontFamily="Arial"
FontSize="20" FontStyle="Normal"
                    FontWeight="Bold" Margin="0 5 0 5" HorizontalAlignment="Center"/>
AS12                 <StackPanel Orientation="Horizontal">
AS13                     <TextBlock Text="Month:" Width="75" />
AS14                     <TextBox Width="200" />
AS15                 </StackPanel>
AS16                 <StackPanel Orientation="Horizontal">
AS17                     <TextBlock Text="Day:" Width="75" />
AS18                     <TextBox Width="200" />
AS19                 </StackPanel>
AS20                 <StackPanel Orientation="Horizontal">
AS21                     <TextBlock Text="Year:" Width="75" />
AS22                     <TextBox Width="200" />
AS23                 </StackPanel>
AS24                 <Ellipse Fill="Blue" Width="50" Height="50" Margin="0 5 0 5"/>
AS25                 <TextBlock FontFamily="Arial" FontSize="14" Foreground="White" Text="Save"
Margin="127 -38 0 0"/>
AS26             </StackPanel>
AS27         </Grid>
AS28     </Grid>
AS29 </Page>
AS30 </Page>

```

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.)

```

01 <ResourceDictionary
02     xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
03     xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
04     xmlns:local="using:_70357rm">
05     <Style x:Key="fill">
06         <Setter Property="Foreground" Value="DarkBlue"/>
07     </Style>
08     <Style x:Key="text">
09         <Setter Property="FontFamily" Value="Arial"/>
10     </Style>
11     <Style x:Key="big">
12         <Setter Property="FontSize" Value="20"/>
13     </Style>
14     <Style x:Key="small">
15         <Setter Property="FontSize" Value="14"/>
16     </Style>
17     <Style x:Key="strong">
18         <Setter Property="FontWeight" Value="Bold"/>
19     </Style>
20     <Style x:Key="light">
21         <Setter Property="FontWeight" Value="Normal"/>
22     </Style>
23     <Style x:Key="normal">
24         <Setter Property="FontStyle" Value="Normal"/>
25     </Style>
26     <Style x:Key="pad">
27         <Setter Property="Margin" Value="0 5 0 5"/>
28     </Style>
29     <Style x:Key="gap">
30         <Setter Property="Margin" Value="20 0 0 0"/>
31     </Style>
32     <Style x:Key="middle">
33         <Setter Property="HorizontalAlignment" Value="Center"/>
34     </Style>
35     <Style TargetType="CheckBox" x:Key="check">
36         <Setter Property="FontFamily" Value="Arial" />
37         <Setter Property="FontSize" Value="14" />
38         <Setter Property="FontStyle" Value="Normal" />
39         <Setter Property="Margin" Value="20 0 0 0" />
40     </Style>
41     <Style TargetType="TextBox" x:Key="heading">
42         <Setter Property="Foreground" Value="DarkBlue" />
43         <Setter Property="FontFamily" Value="Arial" />
44         <Setter Property="FontSize" Value="20" />
45         <Setter Property="FontStyle" Value="Normal" />
46         <Setter Property="FontWeight" Value="Bold" />
47         <Setter Property="Margin" Value="0 5 0 5" />
48         <Setter Property="HorizontalAlignment" Value="Center" />
49     </Style>
50 </ResourceDictionary>

```

New Tab:

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.)

```
MX01 private void App_BackRequest(object sender, Windows.UI.Core.BackRequestedEventArgs e)
MX02     {
MX03         Frame page = Window.Current.Content as Frame;
MX04         if ( page != null)
MX05         {
MX06             if ( page.CanGoBack )
MX07             {
MX08                 page.GoBack();
MX09             }
MX10         }
MX11     }
```

Question: 1

Note: This question is part of a series of questions that present the same scenario. Each Question in the series contains n unique solution. Determine whether the solution meets the stated goals.

You need to implement the appropriate XAML layout for the Timeline app.

Solution: You create an instance of a StackPanel class.

Does this meet the goal?

- A. Yes
- B. No

Answer: B

Explanation:

StackPanel is a simple layout panel that arranges its child elements into a single line that can be oriented horizontally or vertically. StackPanel controls are typically used in scenarios where you want to arrange a small subsection of the UI on your page.

The following XAML shows how to create a vertical StackPanel of items.

XAML

```
<StackPanel>
<Rectangle Fill="Red" Height="44"/>
<Rectangle Fill="Blue" Height="44"/>
<Rectangle Fill="Green" Height="44"/>
<Rectangle Fill="Orange" Height="44"/>
</StackPanel>
```

The result looks like this.



Question: 2

Note: This question is part of a series of questions that present the same scenario. Each Question in the series contains n unique solution. Determine whether the solution meets the stated goals.

You need to implement the appropriate XAML layout for the Timeline app.

Solution: You create an instance of a SplitView control.

Does this meet the goal?

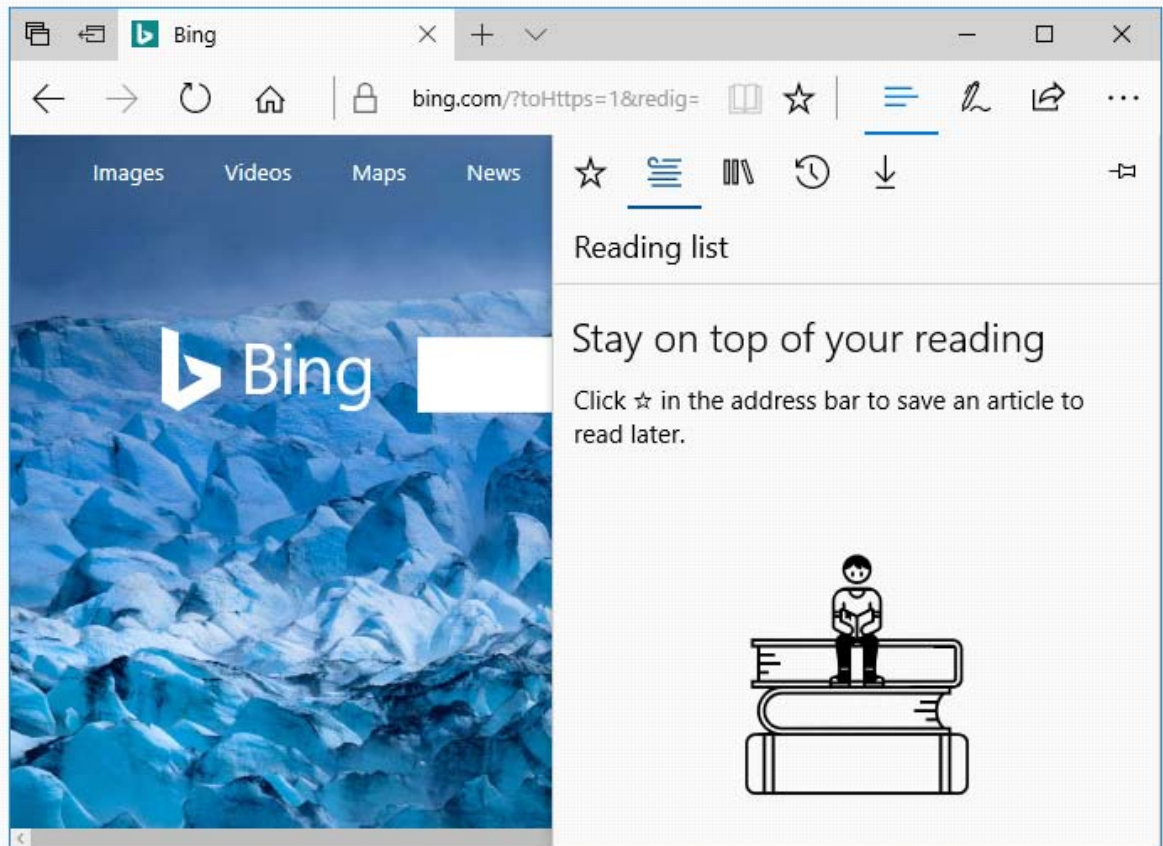
- A. Yes
- B. No

Answer: B

Explanation:

A split view control has an expandable/collapsible pane and a content area.

Here is an example of the Microsoft Edge app using SplitView to show its Hub.



Question: 3

Note: This question is part of a series of questions that present the same scenario. Each Question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You need to implement the appropriate XAML layout (or the Timeline app).

Solution: You create an instance of a RelativePanel class.

Does this meet the goal?

- A. Yes
- B. No

Answer: A

Explanation:

RelativePanel lets you layout UI elements by specifying where they go in relation to other elements and in relation to the panel. By default, an element is positioned in the upper left corner of the panel.

Question: 4

DRAG DROP

You need to create the user interface for the timeline.

Which four markup segments should you use to develop the solution? To answer, move the appropriate markup segments from the list of markup segments to the answer area and arrange them in the correct order.

NOTE: More than one order of answer choices is correct. You will receive credit for any of the correct orders you select.

Markup segments

```
<Button Content ="Favorite"
Click ="UiButton_Click"/>
<TextBlock Text ="|" Margin =
"0 2 0 2"/>
<Button Content ="Back" Click =
"UiButton_Click"/>
<Button Content = "Forward"
Click ="UiButton_Click"/>
```

```
<AppBar>
```

```
</CommandBar>
```

```
<CommandBar>
```

```
<AppBarButton Icon ="Favorite"
Label ="Favorite"
Click ="UiButton_Click"/>
<AppBarSeparator />
<AppBarButton Icon = "Back"
Label = "Back"
Click ="UiButton_Click" />
<AppBarButton Icon ="Forward"
Label = "Forward"
Click ="UiButton_Click"/>
```

```
<AppBar.Tag>
<Button Content = "Comment"
Click ="UiButton_Click"/>
</AppBar.Tag>
</AppBar.Content>
<TextBlock Text ="Navigation"
Margin ="12,14"/>
</AppBar.Content>
```

```
<CommandBar.SecondaryCommands>
<AppBarButton Icon ="Comment"
Label = "Notes"
Click ="UiButton_Click"/>
</CommandBar.SecondaryCommands>
<CommandBar.Content>
<TextBlock Text = "Navigation"
Margin ="12,14"/>
</CommandBar.Content>
```

```
</AppBar>
```

Answer Area



Answer:

Explanation:

Answer Area

```
<CommandBar>
```

```
<AppBarButton Icon="Favorite"
Label="Favorite"
Click="UiButton_Click"/>
<AppBarSeparator />
<AppBarButton Icon="Back"
Label="Back"
Click="UiButton_Click" />
<AppBarButton Icon="Forward"
Label="Forward"
Click="UiButton_Click"/>
```

```
<CommandBar.SecondaryCommands>
  <AppBarButton Icon="Comment"
Label="Notes"
Click="UiButton_Click"/>
</CommandBar.SecondaryCommands>
<CommandBar.Content>
  <TextBlock Text="Navigation"
Margin="12,14"/>
</CommandBar.Content>
```

```
</CommandBar>
```

Box 1: <Commandbar>

Command bars (also called "app bars") provide users with easy access to your app's most common tasks, and can be used to show commands or options that are specific to the user's context, such as a photo selection or drawing mode. They can also be used for navigation among app pages or between app sections. Command bars can be used with any navigation pattern.

XAML provides both the AppBar control and the CommandBar control. You should use the AppBar only when you are upgrading a Universal Windows 8 app that uses the AppBar, and need to minimize changes. For new apps in Windows 10, we recommend using the CommandBar control instead.

Box 2: <AppBarButton .. etc.

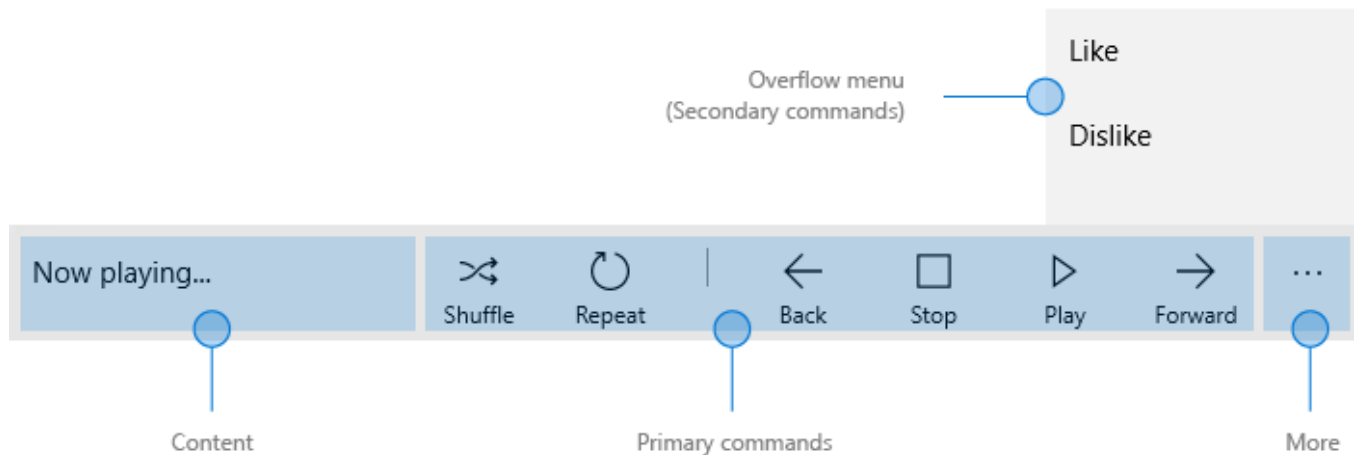
The CommandBar control is a general-purpose, flexible, light-weight control that can display both complex content, such as images or text blocks, as well as simple commands such as AppBarButton, AppBarToggleButton, and AppBarSeparator controls.

Box 3: <CommandBar.SecondaryCommands> etc.

The overflow menu is shown only when the command bar is open and the SecondaryCommands property is populated. The new dynamic overflow behavior will automatically move primary commands into the SecondaryCommands area when space is limited.

Box 4: </Commandbar>

Example: Here is a same command bar in its open state. The labels identify the main parts of the control.



This example creates the command bar shown above.

```
<CommandBar>
  <AppBarToggleButton Icon="Shuffle" Label="Shuffle" Click="AppBarButton_Click" />
  <AppBarToggleButton Icon="RepeatAll" Label="Repeat" Click="AppBarButton_Click"/>
  <AppBarSeparator/>
  <AppBarButton Icon="Back" Label="Back" Click="AppBarButton_Click"/>
  <AppBarButton Icon="Stop" Label="Stop" Click="AppBarButton_Click"/>
  <AppBarButton Icon="Play" Label="Play" Click="AppBarButton_Click"/>
  <AppBarButton Icon="Forward" Label="Forward" Click="AppBarButton_Click"/>

  <CommandBar.SecondaryCommands>
    <AppBarButton Icon="Like" Label="Like" Click="AppBarButton_Click"/>
    <AppBarButton Icon="Dislike" Label="Dislike" Click="AppBarButton_Click"/>
  </CommandBar.SecondaryCommands>

  <CommandBar.Content>
    <TextBlock Text="Now playing..." Margin="12,14"/>
  </CommandBar.Content>
</CommandBar>
```

Question: 5

HOTSPOT

You are reviewing the `App_BackRequested` method in the `MainPage.xaml.cs` file.

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

Answer Area

Statement	Yes	No
In tablet mode, the app will automatically return to the correct page.	<input type="radio"/>	<input type="radio"/>
In desktop mode, the app will automatically return to the correct page.	<input type="radio"/>	<input type="radio"/>
When a user selects an item on the timeline, the CanGoBack property will always return the value True .	<input type="radio"/>	<input type="radio"/>
The App_BackRequested method meets the navigation requirements.	<input type="radio"/>	<input type="radio"/>

Answer:

Explanation:

Answer Area

Statement	Yes	No
In tablet mode, the app will automatically return to the correct page.	<input type="radio"/>	<input checked="" type="radio"/>
In desktop mode, the app will automatically return to the correct page.	<input checked="" type="radio"/>	<input type="radio"/>
When a user selects an item on the timeline, the CanGoBack property will always return the value True .	<input type="radio"/>	<input checked="" type="radio"/>
The App_BackRequested method meets the navigation requirements.	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: No

If we assume that the client is a PC, then the back button is not available in tablet mode.

Box 2: Yes

If we assume that the client is a PC, then the back button is available in desktop mode.

Box 3: No

The user cannot go back from the first item on the timeline.

Box 4: No

From scenario: The Back and Forward buttons navigate through the app selection history. Both buttons must be available on all devices.

Question: 6

You need to design the navigation for the timeline.

What navigation should you use?

- A. hierarchy
- B. peer
- C. hub
- D. master/details

Answer: A

Explanation:

From scenario:

Here we can use a hierarchy with each parent node having only one single child node.

Hierarchical structures are good for organizing complex content that spans lots of pages or when pages should be viewed in a particular order. The downside is that hierarchical pages introduce some navigation overhead: the deeper the structure, the more clicks it takes for users to get from page to page.

We recommend a hierarchical structure when:

You expect the user to traverse the pages in a specific order. Arrange the hierarchy to enforce that order.

There is a clear parent-child relationship between one of the pages and the other pages in the group.

There are more than 7 pages in the group.

When there are more than 7 pages in the group, it might be difficult for users to understand how the pages are unique or to understand their current location within the group. If you don't think that's an issue for your app, go ahead and make the pages peers

Question: 7

You need to configure the app to meet the load time requirements.

What should you do?

- A. Set the value of the CacheSize to 0.
- B. Set the value of the CacheMode property to BitmapCache.
- C. Set the value of the NavigationCacheMode property to Enabled.
- D. Set the value of the NavigationCacheMode property to Disabled.

Answer: C

Explanation:

Scenario: You must optimize the app using the following guidelines:

You must minimize the time it takes to display content when an item on the timeline is selected.
The app must respect memory and resource constraints for all devices.

You use the NavigationCacheMode property to specify whether a new instance of the page is created for each visit to the page or if a previously constructed instance of the page that has been saved in the cache is used for each visit.

The default value for the NavigationCacheMode property is Disabled. Set the NavigationCacheMode property to Enabled or Required when a new instance of the page is not essential for each visit. By using a cached instance of the page, you can improve the performance of your application and reduce the load on your server.

Question: 8

You need to ensure that the Timeline app meets the XAML coding requirements.
In Settings.xaml, which markup segment should you select to replace the markup segment at line AS06?

A

```
<TextBlock Text="Date Settings" Foreground="{StaticResource fill}"
FontFamily="{StaticResource text}"
FontSize="{StaticResource big}" FontStyle="{StaticResource normal}"
FontWeight="{StaticResource strong}"
Margin="{StaticResource pad}" HorizontalAlignment="{StaticResource middle}"/>
```

B

```
<TextBlock Text="Date Settings" Foreground="{StaticResource fill}"
FontFamily="Normal"
FontSize="{StaticResource big}" FontStyle="Normal"
FontWeight="{StaticResource strong}"
Margin="{StaticResource pad}" HorizontalAlignment="{StaticResource middle}"/>
```

C

```
<TextBlock Text="Date Settings" Style="{ ThemeResource heading }" />
```

D

```
<TextBlock Text="Date Settings" Style="{ StaticResource heading }" />
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: A

From scenario: All code and markup must conform to the following style guidelines:

Use resource dictionaries for styles that are used more than once.

Use built-in properties of existing panels instead of using separate style objects.

XAML resources are objects that are referenced from markup more than once. Resources are defined in a ResourceDictionary, typically in a separate file or at the top of the markup page. In this scenario the ResourceDictionary is defined in the ResourceDictionary.xaml file.

You access members of the resource dictionary like any other dictionary.

Question: 9

HOTSPOT

You need to properly handle the size of the user interface objects.

How should you complete the method? To answer, select the appropriate code segment from each list in the answer area.

NOTE: Each correct selection is worth one point.

Answer Area

```
public enum UiMode
{
    UI_LARGE,
    UI_SMALL
}
```

```
private UiMode GetUiMode()
{
```

▼

```
    IReadOnlyList<PointerDevice> pd = PointerDevice.GetPointerDevices();
    MouseCapabilities pd = new Windows.Devices.Input.MouseCapabilities();
```

▼

```
    int pointer = pd.Count;
    int pointer = pd.MousePresent;
```

▼

```
    if (pointer > 0)
    if (pointer == 1)
    if (pointer == 0)
```

```
    {
        return UiMode .UI_SMALL;
    }
    else
    {
        return UiMode UI_LARGE;
    }
}
```

Answer:

Explanation:

```

private UiMode GetUiMode()
{
    IReadOnlyList<PointerDevice> pd = PointerDevice.GetPointerDevices();
    MouseCapabilities pd = new Windows.Devices.Input.MouseCapabilities();

    int pointer = pd.Count;
    int pointer = pd.MousePresent;

    if (pointer > 0)
    if (pointer == 1)
    if (pointer == 0)

```

From scenario:

UI controls must be smaller and spaced closer together if there is a mouse or stylus available.

UI controls must be larger and spaced farther apart if the device supports touch and there is no mouse or pointer available.

Box 1: MouseCapabilities pd = new Windows.Devices.Input.MouseCapabilities();

The Windows.Devices.Input namespace contains the MouseCapabilities class used to retrieve the properties exposed by one or more connected mice. Just create a new MouseCapabilities object and get the properties you're interested in.

Box 2: int pointer = pd.MousePresent;

Example:

MouseCapabilities mouseCapabilities = new Windows.Devices.Input.MouseCapabilities();

MousePresent.Text = mouseCapabilities.MousePresent != 0 ? "Yes" : "No";

Box 3: if (pointer == 1)

This is true if a mouse is present.

Mix questions

Question: 10

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You are developing a Universal Windows Platform (UWP) app.

Your app stores files on a user's device.

You need to be able to replace the existing files with new files generated by the user.

Solution you run the `StorageFile.GetParentAsync` method to get a reference to the existing file. Then, you run the `StorageFile.CreateStreamedFileAsync` method to create the- new file at that same location.

Does this meet the goal?

A. yes

B. No

Answer: A

The `GetParentAsync()` method gets the parent folder of the current file.

The `CreateStreamedFileAsync` method can be used to create a `StorageFile` that can be passed to other methods or passed to another app through app contracts.

Question: 11

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You are developing a Universal Windows Platform (UWP) app.

Your app stores files on a user's device.

You need to be able to replace the existing files with new files generated by the user.

Solution You run the `StorageFile.OpenSequentialReadAsync` method to replace the existing file.

Does this meet the goal?

A. yes

B. No

Answer: B

The `OpenSequentialReadAsync()` method opens a sequential-access stream over the current file for reading file contents.

Question: 12

You have two Universal Windows Platform (UWP) apps named Catalog and Research, respectively.

You need to create a service in the Catalog app that can be queried by the Research app.

Which three tasks should you perform? Each correct answer presents part of the solution.

A. Enter the package family name of the Catalog app in the Catalog app.

B. Add a Windows Runtime component to the Catalog app.

C. Enter the package family name of the Catalog app in the Research app.

D. Add an app service extension to package.appmanifest file in the Research app.

E. Add a Windows Runtime component to the Research app.

F. Add an app service extension to package.appmanifest file in the Catalog app.

Answer: B,C,F

Explanation:

F: Example: Add an app service extension to package.appxmanifest

In the AppServiceProvider project's Package.appxmanifest file, add the following AppService extension to the <Application> element. This example advertises the com.Microsoft.Inventory service and is what identifies this app as an app service provider. The actual service will be implemented as a background task. The app service app exposes the service to other apps

B: Create the app service

An app service is implemented as a background task. This enables a foreground application to invoke an app service in another application to perform tasks behind the scenes. Add a new Windows Runtime Component project to the solution.

C: Deploy the service app and get the package family name

The app service provider app must be deployed before you can call it from a client. You will also need the package family name of the app service app in order to call it.

Question: 13

HOTSPOT

You are developing a Universal Windows Platform (UWP) app by using XAML and C#. A team member has written a XAML page that includes a button with an event handler method named `ButtonSendNotification_Click()` registered to the Click event.

You are reviewing the following code segment written by the team member (line numbers are added for reference only):

```

01 public sealed partial class MainPage : Page
02 {
03     public MainPage()
04     {
05         InitializeComponent();
06         TileUpdateManager.CreateTileUpdaterForApplication().EnableNo-
tificationQueue( true);
07     }
08     private void ButtonSendNotification_Click( object sender, RoutedEventArgs e)
09     {
10         SendTileNotification();
11     }
12     private static string GetNewsTitle()
13     {
14         ...
15     }
16     private void SendTileNotification()
17     {
18         TileNotification tileNotification = GenerateTileNotification();
19         tileNotification.Tag = "newsItem" + GetNewsTitle();
20         TileUpdateManager .CreateTileUpdaterForApplication().Update(tileNotifica-
tion);
21     }
22     private TileNotification GenerateTileNotification()
23     {
24         string xml = @"
25             <tile version='3'>
26                 <visual branding='name'>
27                     <binding template='TileMedium'>
28                         <text hint-wrap='true'>This just in...</text>
29                         <text hint-wrap='true' hint-style='captionSubtle' />
30                     </binding>
31                     <binding template='TileWide'>
32                         <text hint-wrap='true'>This just in...</text>
33                         <text hint-wrap='true' hint-style='captionSubtle' />
34                     </binding>
35                     <binding template='TileLarge'>
36                         <text hint-wrap='true'>This just in...</text>
37                         <text hint-wrap='true' hint-style='captionSubtle' />
38                     </binding>
39                 </visual>
40             </tile>";
41         XmlDocument doc = new XmlDocument ();
42         doc.LoadXml(xml);
43         string nowTimeString = DateTime.Now.ToString();
44         foreach (XmlElement textEl in doc.SelectNodes("//text").OfType<XmlEle-
ment>())
45             if (textEl.InnerText.Length == 0)
46                 textEl.InnerText = "(" + nowTimeString + ") Top News: " + GetNew-
sTitle();
47         TileNotification tileNotification = new TileNotification (doc);
48         return tileNotification;
49     }
50 }

```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each selection is worth one point.

Answer Area

Statement	Yes	No
The code segment will generate a tile notification for all platform tile sizes.	<input type="radio"/>	<input type="radio"/>
The code segment will generate a tile notification successfully when a user clicks the button on the XAML page.	<input type="radio"/>	<input type="radio"/>
The app will display only one tile notification, regardless of the number of button clicks.	<input type="radio"/>	<input type="radio"/>
An exception will be thrown at Line 42 of the code segment.	<input type="radio"/>	<input type="radio"/>

Answer:

Explanation:

Answer Area

Statement	Yes	No
The code segment will generate a tile notification for all platform tile sizes.	<input type="radio"/>	<input checked="" type="radio"/>
The code segment will generate a tile notification successfully when a user clicks the button on the XAML page.	<input checked="" type="radio"/>	<input type="radio"/>
The app will display only one tile notification, regardless of the number of button clicks.	<input type="radio"/>	<input checked="" type="radio"/>
An exception will be thrown at Line 42 of the code segment.	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: No

There are four tile sizes: small, medium, wide, large. Only three are reference in the TileNotification definition.

Box 2: Yes

Box 3: No

Box 4: No

Line 42 is doc.LoadXml(xml);

The LoadXml method loads an XML document from a string. Returns TRUE on success or FALSE on failure.

If called statically, returns a XmlDocument or FALSE on failure.

If an empty string is passed as the source, a warning will be generated. This warning is not generated by libxml and cannot be handled using libxml's error handling functions.

Question: 14

DRAG DROP

You are developing a Universal Windows Platform (UWP) app.

Users can drag and drop images on the screen to share them with their friends.

You need to implement drag and drop for the app. Users must be able to drop images onto a blue rectangle that the app displays.

How should you complete the relevant markup? To answer, drag the appropriate markup segments to the correct targets. Each markup 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.

Markup segments

AllowDrop

DragOver

Grid_DragOver

Drop

Answer area

```
<Grid
  = "True"
  = " "
  ="Grid_Drop" Background="LightBlue" Margin ="10,10,10,353"
>
  ...
  <TextBlock>Drop anywhere in the blue area </TextBlock>
</Grid>
```

Answer:

Explanation:

Answer area

```
<Grid
  AllowDrop = "True"
  DragOver =" Grid_DragOver "
  Drop ="Grid_Drop" Background="LightBlue" Margin ="10,10,10,353"
>
  ...
  <TextBlock>Drop anywhere in the blue area </TextBlock>
</Grid>
```

Box 1: AllowDrop

Box 2: DragOver

Box 3: Grid_DragOver

Box 4: Drop

Use the AllowDrop and CanDrag properties to designate the areas of your app valid for dragging and dropping.

The following markup shows how to set a specific area of the app as valid for dropping by using the AllowDrop in XAML. If a user tries to drop somewhere else, the system won't let them. If you want users to be able to drop items anywhere on your app, set the entire background as a drop target.

```
<Grid AllowDrop="True" DragOver="Grid_DragOver" Drop="Grid_Drop"
Background="LightBlue" Margin="10,10,10,353">
<TextBlock>Drop anywhere in the blue area</TextBlock>
</Grid>
```

Question: 15

You are developing a Universal Windows Platform (UWP) app. The app must allow the user to select only one file at a time.

You need to ensure that the app displays the appropriate dialog window.

Which method should you use?

- A. FileOpenPicker.PickSingleFileAsync()
- B. FileOpenPicker.PickMultipleFilesAsync()
- C. StorageItem.OpenSequentialReadAsync()
- D. StorageItem.GetFileFromPathAsync()
- E. StorageItem.OpenReadAsync()

Answer: A

Explanation:

To pick a single file.

```
Windows.Storage.StorageFile file = await picker.PickSingleFileAsync();
if (file != null)
{
    // Application now has read/write access to the picked file
    this.textBlock.Text = "Picked photo: " + file.Name;
}
else
{
    this.textBlock.Text = "Operation cancelled.";
}
```

Question: 16

HOTSPOT

You are developing a Universal Windows Platform (UWP) app that plays audio recordings.

You are creating a page where the user can set a volume level for the app using a slider control. You need to display the volume level in a TextBox right below the slider. You have C# class named VolumeConverter that converts slider values to a number.

You have a page that includes the following markup:

```
<Page ...>
  <Page.Resources>
    <local : S2Formatter x : Key="VolumeConverter"/>
  </Page.Resources>
  <Slider x: Name ="VolumeSlider"/>
    <TextBox Text="{Binding Path=Value, ElementName =VolumeSlider,
Mode=OneWay, Converter={StaticResource VolumeConverter}}"/>
  </Page>
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

Answer Area

Statement	Yes	No
VolumeConverter must implement the IValueConverter interface.	<input type="radio"/>	<input type="radio"/>
When a user moves the slider control, the corresponding volume level displays in the TextBox control.	<input type="radio"/>	<input type="radio"/>
When a user changes the value in the TextBox the slider will automatically move to the corresponding position.	<input type="radio"/>	<input type="radio"/>

Answer:

Explanation:

Answer Area

Statement	Yes	No
VolumeConverter must implement the IValueConverter interface.	<input checked="" type="radio"/>	<input type="radio"/>
When a user moves the slider control, the corresponding volume level displays in the TextBox control.	<input checked="" type="radio"/>	<input type="radio"/>
When a user changes the value in the TextBox the slider will automatically move to the corresponding position.	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: Yes

Box 2: Yes

Box 3: No

Question: 17

HOTSPOT

You are developing a Universal Windows Platform (UWP) app that processes and displays data from your company's personnel database.

Users report that one of the views in the UWP app loads slowly.

You need to optimize the load time.

How should you complete the relevant markup? To answer, select the appropriate markup segment from each list in the answer area.

Answer Area

```
<DataTemplate x:Key
```

	▼
uwp:Control="GridView"	
x:Name="GridViewName"	
x:DataType="local:Person"	

```
>
```

```
= "GridViewDataTemplate"
```

```
<StackPanel>
```

```
<TextBlock Text="
```

	▼
{x:Bind FirstName}	
{Binding FirstName}	
{StaticResource FirstName}	

```
" />
```

```
</StackPanel>
```

```
</DataTemplate>
```

Answer:

Explanation:

Answer Area

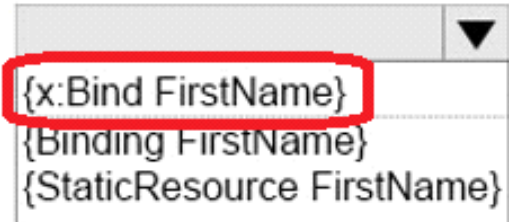
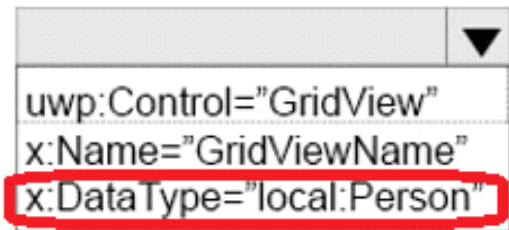
```
<DataTemplate x:Key
```

```
= "GridViewDataTemplate"
```

```
<StackPanel>
```

```
<TextBlock Text="
```

```
</StackPanel>
```



The {x:Bind} markup extension—new for Windows 10—is an alternative to {Binding}. {x:Bind} lacks some of the features of {Binding}, but it runs in less time and less memory than {Binding} and supports better debugging.

In the following example, the background and foreground of the item are bound to functions to do conversion based on the color parameter

```
<DataTemplate x:DataType="local:ColorEntry">
<Grid Background="{x:Bind Brushify(Color)}" Width="240">
<TextBlock Text="{x:Bind ColorName}" Foreground="{x:Bind TextColor(Color)}" Margin="10,5" />
</Grid>
</DataTemplate>
```

Question: 18

You are developing a Universal Windows Platform (UWP) app that uses XAML and C#. The app must use the Model-View-ViewModel (MVVM) pattern.

The user interface (UI) triggers an event.

You need to bind the event to a view model method.

What should you do?

A. Create a custom behavior and attach the behavior to the UI element. Bind the behavior's event trigger to the command declared in the view model.

-
- B. Create an attached property of type ICommand. Bind the UI element's event to the attached property.
- C. Assign the value of the DataContext property to the view model. Use the BindingExpression.UpdateSource() method to update the data source.
- D. Add a strongly-typed view model property to the view. In the code behind file for the view, invoke the view model method.

Answer: B

Explanation:

Commands are an implementation of the ICommand interface that is part of the .NET Framework. This interface is used a lot in MVVM applications.

Question: 19

DRAG DROP

You are building a Universal Windows Platform (UWP) app that displays a list of books.

The books will be displayed in a ListView control. You are binding to a collection of type BookDataGroup.

You need to create a DataTemplate for your ListView that displays the book's title and author. The title must be displayed before the author.

Which four markup segments should you use to develop the solution? To answer, move the appropriate markup segments from the list of markup segments to the answer area and arrange them in the correct order.

Actions

```
<DataTemplate x:Key="BookDataGroup"
DataType="data:BookItemTemplate">
```

```
</StackPanel>
</DataTemplate>
```

```
<DataTemplate x:Key ="BookItemTemplate"
x:DataType ="data:BookDataGroup">
<StackPanel>
```

```
<TextBlock Text="{x : Bind Title}"/>
```

```
<TextBlock Text="{x : Bind Author}"/>
```

```
<TextBlock Text="{x : Bind Title,
Source =OneWay}"/>
```

```
<TextBlock Text="{x : Bind Author,
Source =OneWay}"/>
```

Answer Area



Answer:

Explanation:

Answer Area

```
<DataTemplate x:Key ="BookItemTemplate"
x:DataType ="data:BookDataGroup">
<StackPanel>
```

```
<TextBlock Text="{x : Bind Title}"/>
```

```
<TextBlock Text="{x : Bind Author}"/>
```

```
</StackPanel>
</DataTemplate>
```


Question: 20

You have a Universal Windows Platform (UWP) app. The app has a page that includes the following XAML markup. Line numbers are included for reference only.

```
01 <DataTemplate >
02     <Border Background="Blue" Width="400" Height="300" Margin ="20">
03         <Grid>
04             <Grid.ColumnDefinitions >
05                 <ColumnDefinition Width ="*" />
06                 <ColumnDefinition Width ="*" />
07             </Grid.ColumnDefinitions >
08             <Rectangle Grid.Column ="1" Fill="White" Opacity =".66"/>
09             <TextBlock Text ="{Binding LastName }"/>
10         </Grid>
11     </Border>
12 </DataTemplate>
```

Users report that the page takes a long time to refresh.

You need to improve the load time for the page while maintaining the same layout and functionality.

What should you do?

- A. Move the attributes from the BORDER element at line 02 to the GRID element at line 03. Then, remove the BORDER elements at line 02 and line 11.
- B. Replace the TEXTBLOCK element at line 09 with a TEXTBOX element.
- C. Swap the markup at line 02 with the markup at line 03. Swap the markup at line 10 with the markup at line 11.
- D. Move the Fill and Opacity attributes and value from the RECTANGLE element at line 08 to the GRID element at line 03. Then, Remove the RECTANGLE element.

Answer: D

Explanation:

Use single-cell grids for overlapping UI

A common UI requirement is to have a layout where elements overlap each other. Typically padding, margins, alignments, and transforms are used to position the elements this way. The XAML Grid control is optimized to improve layout performance for elements that overlap.

Question: 21

HOTSPOT

You are developing a Universal Windows Platform (UWP) app.

The app does not display content properly on mobile devices.

You need to support smaller window sizes.

How should you complete the relevant XAML markup? To answer, select the appropriate markup segment from each list in the answer area.

Answer Area

```
<SplitView :Name="SplitViewControl" OpenPaneLength="128">
  <VisualStateManager.VisualStateGroups>
    <VisualStateGroup>
      <VisualState>
        <VisualState.StateTriggers>
          <MediaQuery MinWindowWidth="0" MaxWindowWidth="256" />
        </VisualState.StateTriggers>
        <VisualState.Setters>
          <Setter Target="SplitViewControl.IsPaneOpen" Value="False" />
          <Setter Target="SplitViewControl.DisplayMode" Value="CompactInline" />
        </VisualState.Setters>
      </VisualState>
      <VisualState>
        <VisualState.StateTriggers>
          <MediaQuery MinWindowWidth="768" MaxWindowWidth="768" />
        </VisualState.StateTriggers>
        <VisualState.Setters>
          <Setter Target="SplitViewControl.IsPaneOpen" Value="True" />
          <Setter Target="SplitViewControl.DisplayMode" Value="Inline" />
        </VisualState.Setters>
      </VisualState>
    </VisualStateGroup>
  </VisualStateManager.VisualStateGroups>
</SplitView>
```

Answer:

Explanation:

Answer Area

```
<SplitView :Name="SplitViewControl" OpenPanelLength="128">
  <VisualStateManager.VisualStateGroups>
    <VisualStateGroup>
      <VisualState>
        <VisualState.StateTriggers>
          <AdaptiveTrigger MinWindowWidth="768" />
        </VisualState.StateTriggers>
        <VisualState.Setters>
          <Setter Target="SplitViewControl.IsPaneOpen" Value="False" />
          <Setter Target="SplitViewControl.DisplayMode" Value="CompactInline" />
        </VisualState.Setters>
      </VisualState>
      <VisualState>
        <VisualState.StateTriggers>
          <AdaptiveTrigger MinWindowWidth="0" />
        </VisualState.StateTriggers>
        <VisualState.Setters>
          <Setter Target="SplitViewControl.IsPaneOpen" Value="True" />
          <Setter Target="SplitViewControl.DisplayMode" Value="Inline" />
        </VisualState.Setters>
      </VisualState>
    </VisualStateGroup>
  </VisualStateManager.VisualStateGroups>
</SplitView>
```

One of the tools that Microsoft gives us for building adaptive UIs in UWP apps is state triggers. The version of Windows 10 released at BUILD 2015 features one state trigger: a class named `AdaptiveTrigger`. `AdaptiveTrigger` has two important properties: `MinWindowWidth` and `MinWindowHeight`. You use `AdaptiveTrigger` in conjunction with Visual State Manager to adapt the UI to screens and windows of various sizes.

* Inline

The pane is always visible and doesn't overlay the content area. The pane and content areas divide the available screen real estate.

* CompactInline

A narrow portion of the pane is always visible in this mode, which is just wide enough to show icons. The default closed pane width is 48px, which can be modified with `CompactPanelLength`. If the pane is opened, it will reduce the space available for content, pushing the content out of its way.

Question: 22

You must create a control that meets the following requirements:

- allows you to extend the behavior of a combo box allows the arrow image is located at the right edge of a standard control to be replaced with a new image
- has a property that sets and returns the image
- has a visual interface of the control that is defined by using XAML
- defines the properties for the control in code

You need to create the control.

Which object should you use?

- A. ContentDialog
- B. StaticResource
- C. ThemeResource
- D. UserControl

Answer: A

Explanation:

ContentDialog represents a dialog box that can be customized to contain checkboxes, hyperlinks, buttons and any other XAML content.

Question: 23

HOTSPOT

You are developing a Universal Windows Platform (UWP) app.

You need to implement responsive user design patterns.

Which of the following techniques are supported? To answer, select the appropriate option from each list in the answer area.

Answer Area

Technique	Response
Reposition	<div><div>▼</div><div>Supported</div><div>Not supported</div></div>
Resize	<div><div>▼</div><div>Supported</div><div>Not supported</div></div>
Replace	<div><div>▼</div><div>Supported</div><div>Not supported</div></div>
Reflow	<div><div>▼</div><div>Supported</div><div>Not supported</div></div>
Reveal	<div><div>▼</div><div>Supported</div><div>Not supported</div></div>

Answer:

Explanation:

Technique	Response
Reposition	<div> <div>Supported</div> <div>Not supported</div> </div>
Resize	<div> <div>Supported</div> <div>Not supported</div> </div>
Replace	<div> <div>Supported</div> <div>Not supported</div> </div>
Reflow	<div> <div>Supported</div> <div>Not supported</div> </div>
Reveal	<div> <div>Supported</div> <div>Not supported</div> </div>

Responsive design techniques

When you optimize your app's UI for specific screen widths, we say that you're creating a responsive design. Here are six responsive design techniques you can use to customize your app's UI.

* Reposition

You can alter the location and position of app UI elements to get the most out of each device

* Resize

You can optimize the frame size by adjusting the margins and size of UI elements.

* Reflow

By changing the flow of UI elements based on device and orientation, your app can offer an optimal display of content-

*** Show/hide**

You can show or hide UI elements based on screen real estate, or when the device supports additional functionality, specific situations, or preferred screen orientations.

*** Replace**

This technique lets you switch the user interface for a specific device size-class or orientation. In this example, the nav pane and its compact, transient UI works well for a smaller device, but on a larger device tabs might be a better choice.

*** Re-architect**

You can collapse or fork the architecture of your app to better target specific devices.

Question: 24

You have to connect your app to an online identity provider that uses OAuth authentication protocol.

The app must securely use the WebAuthenticationBroker object for authentication.

You need to ensure that the app registers with the provider.

Which two actions should you perform? Each correct answer presents part of the solution.

- A. Construct a HTTP request URI.
- B. Call the GetCurrentApplicationCallbackUri method.
- C. Call the AuthenticateAsync method.
- D. Construct a HTTPS request URI.

Answer: C,D

Explanation:

The current application callback URI is used as an implicit value of the callbackUri parameter of the AuthenticateAsync method. However, applications need the URI value to add it to the request URI as required by the online provider.

The requestUri parameter must be a HTTPS address: an exception will be thrown if an HTTP address is used, even for local testing scenarios.

Question: 25

HOTSPOT

You are developing a Universal Windows Platform (UWP) app that stores credentials by using the Credential Locker service.

You need to securely retrieve credentials for the current user.

How should you complete the method? To answer, select the appropriate code segment from each list in the answer area.

Answer Area

```
private void RetrieveCredentials(string resource, string userName)
{
    var identityStorage = 

new PasswordVault();
        KeyCredentialManager.OpenAsync(userName);
        CredentialPicker.PickAsync(resource, userName);


    var credentials = 

identityStorage.GetResults();
        identityStorage.Retrieve(resource, userName)


}
```

Answer:

Explanation:

Answer Area

```
private void RetrieveCredentials(string resource, string userName)
```

```
{
```

```
    var identityStorage =
```

```
        new PasswordVault();
```

```
        KeyCredentialManager.OpenAsync(userName);
```

```
        CredentialPicker.PickAsync(resource, userName);
```

```
    var credentials =
```

```
        identityStorage.GetResults();
```

```
        identityStorage.Retrieve(resource, userName)
```

Box 1:

Example:

```
var vault = new Windows.Security.Credentials.PasswordVault();
```

Box 2:

Example continued:

```
// When there are multiple usernames,  
// retrieve the default username. If one doesn't  
// exist, then display UI to have the user select  
// a default username.
```

```
defaultUserName = GetDefaultUserNameUI();
```

```
credential = vault.Retrieve(resourceName, defaultUserName);
```

Question: 26

DRAG DROP

You are developing a Universal Windows Platform (UWP) app.

The app has the following requirements:

- . Users must be able to authenticate with the app by using a third-party OAuth provider.
- . Users must have the option of using Single Sign-On.

You obtain the security identifier (SID) for the app from the Windows Dev Center.

You need to implement authentication for the app.

In which order should you perform the actions? To answer, move all actions from the list of actions to the answer area and arrange them in the correct order.

Actions

Register the app with the OAuth provider.

Retrieve user information from the OAuth provider.

Call the AuthenticateAsync method of the Web Authentication Broker.

Construct the authentication request.

Answer Area



Answer:

Explanation:

Answer Area

Register the app with the OAuth provider.

Call the AuthenticateAsync method of the Web Authentication Broker.

Retrieve user information from the OAuth provider.

Construct the authentication request.

Step 1: Register your app with your online provider

You must register your app with the online identity provider to which you want to connect. You can find out how to register your app from the identity provider. After registering, the online provider typically gives you an Id or secret key for your app.

Step 2: Build the authentication request URI

The request URI consists of the address where you send the authentication request to your online provider appended with other required information, such as an app ID or secret, a redirect URI where the user is sent after completing authentication, and the expected response type. You can find out from your provider what parameters are required.

Step 3-4: Connect to the online provider

You call the `AuthenticateAsync` method to connect to the online identity provider and get an access token. The method takes the URI constructed in the previous step as the `requestUri` parameter, and a URI to which you want the user to be redirected as the `callbackUri` parameter.

Note: Step 5: Connecting with single sign-on (SSO).

By default, Web authentication broker does not allow cookies to persist. Because of this, even if the app user indicates that they want to stay logged in (for example, by selecting a check box in the provider's login dialog), they will have to login each time they want to access resources for that provider. To login with SSO, your online identity provider must have enabled SSO for Web authentication broker, and your app must call the overload of `AuthenticateAsync` that does not take a `callbackUri` parameter. This will allow persisted cookies to be stored by the web authentication broker, so that future authentication calls by the same app will not require repeated sign-in by the user (the user is effectively "logged in" until the access token expires).

Question: 27

You are developing a Universal Windows Platform (UWP) app.

A user terminates the app.

You need to ensure that the app initializes when the user starts the app.

How should you complete the method? To answer, drag the appropriate code segment to the correct location or locations. 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.

Code segments

OnLaunched

SearchActivatedEventArgs args

IActivatedEventArgs args

LaunchActivatedEventArgs e

OnActivated

OnSearchActivated

OnShareTargetActivated

ShareTargetActivatedEventArgs args

• • • •

Answer Area

```
protected async override void  
{  
    ...  
}
```

Code segment

{ Code segment

Answer:

Explanation:

Answer Area

```
protected async override void  
{  
    ...  
}
```

OnLaunched

LaunchActivatedEventArgs e

The OnLaunched method is called when an app is launched. It is passed a LaunchActivatedEventArgs parameter which provides, among other things, the arguments passed to the app, the identifier of the tile that launched the app, and the previous state that the app was in.

Box 1: OnLaunched

Box 2: LaunchActivatedEventArgs e

Question: 28

You are designing a roadside assistance mobile app. The app displays a persistent list of links to pages. The app displays a persistent list of links to pages. The pages provide a quick way to move between different views of the app.

You need to recommend a user interface pattern that meets the following requirements:

- Allow users to navigate to frequently accessed, distinct content categories,
- Provide two or more content panes that have corresponding category headers.
- Display the navigation controls on the top of the screen.
- Highlight the currently selected navigation control.

Which pattern should you recommend?

- A. hub
- B. tabs and pivots
- C. active canvas
- D. master/details

Answer: B

Explanation:

The Pivot control and related tabs pattern are used for navigating frequently accessed, distinct content categories. Pivots allow for navigation between two or more content panes and relies on text headers to articulate the different sections of content.

Tabs are a visual variant of Pivot that use a combination of icons and text or just icons to articulate section content. Tabs are built using the Pivot control.

Question: 29

You are developing a Universal Windows Platform (UWP) app.

The app must be available on Windows Phone, Windows tablet devices, and Xbox.

When the app is running on a device, you need to determine which members of a specific class you can use.

Which of the following methods should you use?

- A. ApiInformation.IsPropertyPresent
- B. UserInformation.NameAccessAllowed
- C. Selector.GetIsSelectionActive
- D. AppExtensionCatalog.FindAllAsync

Answer: D

Explanation:

The AppExtensionCatalog class represents a device. This class allows access to well-known device properties as well as additional properties specified during device enumeration.

A Successful completion of FindAllAsync results in a DeviceInformationCollection containing DeviceInformation objects.

Question: 30

DRAG DROP

You are developing a Universal Windows Platform (UWP) app. The app runs on multiple device families, including desktop, Windows Phone, and Xbox.

The app must be able to access a user's media playlists if the device supports this feature. If the device does not support this feature, the app must continue to function.

You need to detect whether a device supports accessing user playlists.

How should you complete the relevant code? To answers, drag the appropriate code segment to the correct location or locations. 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.

Code segments

ApiInformation

Application

BindingExpression

WeakReference

Playlist

Windows.Media.Playlists.Playlist

System.Audio.Songlist

Songlist

...

Answer Area

```
using Windows.Foundation.Metadata;
```

```
if (  .IsTypePresent("  "))  
{  
    await myAwesomePlaylist.SaveAsAsync( ... );  
}
```

Answer:

Explanation:

Answer Area

```
using Windows.Foundation.Metadata;  
  
if ( ApiInformation .IsTypePresent(" Windows.Media.Playlists.Playlist "))  
{  
    await myAwesomePlaylist.SaveAsAsync( ... );  
}
```

Box 1: ApiInformation

Box 2: Windows.Media.Playlists.Playlist

example:

```
using Windows.Foundation.Metadata;
```

```
if(ApiInformation.IsTypePresent("Windows.Media.Playlists.Playlist"))  
{  
    await myAwesomePlaylist.SaveAsAsync( ... );  
}
```

This code makes a runtime check for the presence of the Playlist class, then statically references and calls the SaveAsAsync method on the class.

Question: 31

DRAG DROP

You are developing a Universal Windows Platform (UWP) app that needs to run on multiple types of devices.

The app must detect whether a device has a physical camera button.

For devices that have a physical camera button, you need to ensure that the app continues to function.

How should you complete the code? To answer, drag the appropriate code segments to the correct location or locations. 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.

Code segments

Windows.Phone.UI.Input

ApiInformation

IsTypePresent

Windows.Phone.UI.Input.HardwareButtons

HardwareButtons

••••

Answer Area

```
using Windows.Foundation.Metadata;
```

```
using  ;
```

```
...
```

```
bool isPresent =
```

```
 *   
    ("  ");
```

```
if (isPresent)
```

```
{  
    HardwareButtons.CameraPressed += HardwareButtons_CameraPressed;  
}
```

Answer:

Explanation:

Answer Area

```
using Windows.Foundation.Metadata;

using Windows.Phone.UI.Input

...

bool isPresent =
    ApiInformation.IsTypePresent
        ("HardwareButtons

if (isPresent)
{
    HardwareButtons.CameraPressed += HardwareButtons_CameraPressed;
}
```

Box 1: Windows.Phone.UI.Input

Box 2: ApiInformation

Box 3: IsTypePresent

Box 4: HardwareButtons

You could check and see if the backButton is present to see if it is a mobile device

```
bool isHardwareButtonsAPIPresent =
Windows.Foundation.Metadata.ApiInformation.IsTypePresent("Windows.Phone.UI.Input.HardwareButtons")
```

Question: 32

You are developing a Universal Windows Platform (UWP) app that allows users to take photos and record videos. The photos and videos must be stored in the user's Photos library and Videos library, respectively. The app must not display a user interface for saving files.

You need to configure the app.

Which set of capabilities should you declare in the app manifest?

- A. Internet (client), microphone, location and proximity
- B. webcam, microphone, Pictures library and Video library
- C. Internet (client), Documents library, Videos library, and proximity
- D. webcam, location, proximity and Pictures library

Answer: B

Explanation:

You must specify the webcam or microphone capabilities in your app manifest file if you are using MediaCapture to capture audio, photos, or video programmatically.

Question: 33

You are developing a Universal Windows Platform (UWP) app.

You need to provide a solution that moves the scroll bars of the ScrollViewer when a user rotates the mouse wheel.

Which two actions should you perform? Each correct answer presents part of the solution.

- A. Evaluate the `CurrentPoint.Properties.MouseWheelDelta` property of the `PointerEventArgs` object. Call the `ChangeView()` method of the `ScrollViewer`.
- B. Update the XAML of the `ScrollViewer` to include the `PointerWheelChanged` event with a new event handler. Evaluate the `Pointer.IsInRange` property of the `PointerRoutedEventArgs` object within the event handler. Call the `ChangeView()` method of the `ScrollViewer`.
- C. Add an event handler to the `PointerRoutedEventArgs` event for the current window.
- D. Evaluate the `CurrentPoint.Properties.IsHorizontalMouseWheel` property of the `PointerEventArgs` object. Call the `ChangeView()` method of the `ScrollViewer`.
- E. Add an event handler to the `PointerWheelChanged` event for the current window.

Answer: C,D

Explanation:

The `PointerRoutedEventArgs` event occurs on the process receiving input when the pointer input is routed to another process.

Question: 34

DRAG DROP

You are creating a Universal Windows Platform (UWP) app that takes pictures.

You want to use the camera's built-in interface for taking the pictures.

You need to capture an image from the device's built-in camera.

How should you complete the method? To answer, drag the appropriate code segments to the correct location or locations. Each code segments 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.

Code segments

CameraCaptureUI

CameraCaptureUIPhotoFormat

MediaCapture

ImageEncodingProperties

PhotoSettings

••••

Answer Area

```
public async void GetImage()  
{  
    [Code segment] capture = new [Code segment] ();  
    capture.PhotoSettings.Format = CameraCaptureUIPhotoFormat.Jpeg;  
    StorageFile photo = await capture.CaptureFileAsync(CameraCaptureUIMode.Photo);  
}
```

Answer:

Explanation:

Answer Area

```
public async void GetImage()  
{  
    [CameraCaptureUI] capture = new [CameraCaptureUI] ();  
    capture.PhotoSettings.Format = CameraCaptureUIPhotoFormat.Jpeg;  
    StorageFile photo = await capture.CaptureFileAsync(CameraCaptureUIMode.Photo);  
}
```

Box 1: CameraCaptureUI

Box 2: CameraCaptureUI

Example: Using Windows.Media.Capture.CameraCaptureUI API to capture a photo

CameraCaptureUI dialog = new CameraCaptureUI();

Size aspectRatio = new Size(16, 9);

dialog.PhotoSettings.CroppedAspectRatio = aspectRatio;

StorageFile file = await dialog.CaptureFileAsync(CameraCaptureUIMode.Photo);

Question: 35

You are developing an app that displays photos.

You need to create a method that displays informational text when a user hovers the pointer over a photo.

How should you complete the method? To answer, select the appropriate code segment from each list in the answer area.

Answer Area

```
private void CreateToolTip( DependencyObject  
String target, DependencyObject  
String content)
{
    ToolTip  
ToolTipService tip = new ToolTip  
ToolTipService ();
    tip.Content = content;

    ToolTip  
ToolTipService .SetToolTip(target, tip);
}
```

Answer:

Explanation:

Answer Area

```
private void CreateToolTip( DependencyObject  
String target, DependencyObject  
String content)
{
    ToolTip  
ToolTipService tip = new ToolTip  
ToolTipService ();
    tip.Content = content;

    ToolTip  
ToolTipService .SetToolTip(target, tip);
}
```

Box 1: DependencyObject
Box 2: String
Box 3: ToolTip
Box 4: ToolTip
Box 5: ToolTipService

A ToolTip must be assigned to another UI element that is its owner. In Extensible Application Markup Language (XAML), use the ToolTipService.ToolTip attached property to assign the ToolTip to an owner. In code, use the ToolTipService.SetToolTip method to assign the ToolTip to an owner.

The SetToolTip(DependencyObject, Object) method sets the value of the ToolTipService.ToolTip XAML attached property.

Question: 36

HOTSPOT

You have an app that includes the following method:

```
private void GeoMode ()
{
    Geolocator geo = new Geolocator();
    geo.DesiredAccuracy = Windows.Devices.Geolocation.PositionAccuracy.High;
}
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

Answer Area

Statement	Yes	No
The position accuracy will be 50 meters.	<input type="radio"/>	<input type="radio"/>
The code will configure the device to use the most conservative power settings.	<input type="radio"/>	<input type="radio"/>

Answer:

Explanation:

Answer Area

Statement	Yes	No
The position accuracy will be 50 meters.	<input type="radio"/>	<input checked="" type="radio"/>
The code will configure the device to use the most conservative power settings.	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: No

When neither `DesiredAccuracyInMeters` nor `DesiredAccuracy` are set, your app will use an accuracy setting of 500 meters (which corresponds to the `DesiredAccuracy` setting of Default). Setting `DesiredAccuracy` to Default or High indirectly sets `DesiredAccuracyInMeters` to 500 or 10 meters, respectively.

Box 2: No

The Default value should be used to optimize for power, performance, and other cost considerations. The High value should be used to deliver the most accurate report possible. This includes using services that might charge money, or consuming higher levels of battery power or connection bandwidth. An accuracy level of High may degrade system performance and should be used only when necessary.

Case Study 2

Background

You are developing a Universal Windows Platform (UWP) app for LitWare, Inc. that will assist video artists. The app allows artists to create videos, share videos through other mobile apps, and upload the videos through LitWare, Inc's web services. What helps set LitWare Inc's app apart from competitors is their focus on speed and performance.

Business requirements

Support many devices

. Users may have phones, tablets, or laptops. The app must support all devices with a fluid layout that grows off-screen and adapts to each device.

Record video

-
- Users must be able to record videos and view them in their videos library.
 - The app must display information about the recorded video.
 - Users must be able to edit the upload queue in the app.

Branding

The app must allow deferral and scheduling of video uploads.

Users must be able to view the status of video uploads.

Any videos created with this app or shared with this app must be uploaded without user interaction.

Download video

The app must have the option to automatically download videos.

Users must be able to initiate downloading of videos.

Share video

The app must allow be allowed to receive videos from other apps.

- Technical requirements
- Application structure

Technical requirements

Support multiple devices

The app must support the following:

- Use horizontal layout for larger screens.
- Use vertical layout for smaller screens.
- Use one layout control per view.

The app must be compatible with current and future XBOX app that use C++.

Uploading and downloading

The app must use a background operations to upload and download videos.

Code reuse

- The app must use a common pool of XAML resources and custom controls. All custom controls must use a consistent theme throughout the app.
- You must create code that can be reused in C++, C#, JavaScript, whenever possible.
- The app must call the background service to avoid duplication of code.

Security

- End users must be authenticated using OAuth.
- Web services must be authenticated.
- Users must have the option to use single sign-on.

Recording

The app must use the microphone and webcam to support audio and video recording. In addition, the app must support the use of the back camera buttons, if present.

Integration:

Other apps must be able to share videos with this app through a Universal Windows Platform (UWP)

Architecture and coding style

- The app must follow the Model-View-ViewModel (MVVM) pattern
- The app's user interface (UI) must be optimized for performance.
- The app must use compile time coding techniques over runtime.

Package appxmanifest

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.

```

PM01 <Package>
PM02     <Applications>
PM03         <Application id="App" Executable=" $targetnametoken$.exe" EntryPoint="Litware.-
MultiMediaApp.App">
PM04             <Extensions>
PM05                 <Extension Category = "windows.appService"
                    EntryPoint="Litware.MultiMediaApp.VideoUploadService.VideoUpload">
PM06
PM07             </Extension>
PM08             <Extensions>
PM09                 <uap:VisualElements DisplayName="Litware.MultiMediaApp"
Square150x150Logo=" Assets\Square150x150Logo.png"
Square44x44Logo = "Assets\Square44x44Logo.png" Description="
Litware.MultiMediaApp"
                    BackgroundColor = "transparent" >
PM10                 <uap:DefaultTile Wide310x150Logo="Assets\Wide310x150Logo.png">
PM11                 </uap:DefaultTile>
PM12                 <uap:SplashScreen Image = "Assets\SplashScreen.png" />
PM13                 </uap:VisualElements>
PM14             </Application>
PM15         </Applications>
PM16 </Package>

```

Background Task

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.


```
BA01 private void RegisterBackgroundTask(object sender, RoutedEventArgs e)
BA02 {
BA03     if (! this.IsAlreadyRegistered(this .videoDownloadBackgroundTaskName))
BA04     {
BA05         var backgroundTaskRegistration = this.RegisterBackgroundTask(
BA06             this .videoDownloadBackgroundTaskName,
BA07             this .videoDownloadBackgroundTaskEntryPoint,
BA08
BA09             null);
BA10         backgroundTaskRegistration.Progress += OnProgress;
BA11         backgroundTaskRegistration.Completed += OnCompleted;
BA12     }
BA13 }
```

Question: 37

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You must create a project for shared code.

Solution: You implement the shared code in a Windows Runtime component.

Does this meet the goal?

A. Yes

B. No

Answer: A

Question: 38

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You must create a project for shared code.

Solution: You implement the shared code in a Shared Project.

Does this meet the goal?

A. Yes

B. No

Answer: B

Explanation:

The .NET Framework Portable Class Library project type in Visual Studio helps you build cross-platform apps and libraries for Microsoft platforms quickly and easily.

Portable class libraries can help you reduce the time and costs of developing and testing code. Use this project type to write and build portable .NET Framework assemblies, and then reference those assemblies from apps that target multiple platforms such as Windows and Windows Phone.

Even after you create a Portable Class Library project in Visual Studio and start developing it, you can change the target platforms. Visual Studio will compile your library with the new assemblies, which helps you identify the changes you need to make in your code.

From scenario:

The app must be compatible with current and future XBOX apps that use C++.

Question: 39

Note: This question is part of a series of that present the same scenario. Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You must create a project for shared code.

Solution: You implement the shared code in a .NET class library.

Does this meet goal?

A. Yes

B. No

Answer: B

The .NET Framework Portable Class Library, not a .NET class library, project type in Visual Studio helps you build cross-platform apps and libraries for Microsoft platforms quickly and easily.

Portable class libraries can help you reduce the time and costs of developing and testing code. Use this project type to write and build portable .NET Framework assemblies, and then reference those assemblies from apps that target multiple platforms such as Windows and Windows Phone.

Even after you create a Portable Class Library project in Visual Studio and start developing it, you can change the target platforms. Visual Studio will compile your library with the new assemblies, which helps you identify the changes you need to make in your code.

From scenario:

The app must be compatible with current and future XBOX apps that use C++.

Question: 40

You are a developer for LitWare, Inc.'s Universal Windows Platform (UWP) app.

Access to the hardware within the app is not functioning correctly.

You need to add the capabilities to the package.appmanifest file.

Which markup segment should you insert at line PM16?

A

```
<Capabilities>
  <Capability Name = "internetClient" />
  <uap:Capability Name = "videosLibrary" />
  <DeviceCapability Name = "microphone" />
  <DeviceCapability Name ="location" />
</Capabilities>
```

B

```
<Capabilities>
  <uap:Capability Name = "videosLibrary" />
  <uap:Capability Name = "removableStorage" />
  <DeviceCapability Name = "microphone" />
  <DeviceCapability Name ="webcam" />
</Capabilities>
```

C

```
<Capabilities>
  <Capability Name = "internetClient" />
  <uap:Capability Name = "picturesLibrary" />
  <DeviceCapability Name = "microphone" />
  <DeviceCapability Name ="webcam" />
</Capabilities>
```

D

```
<Capabilities>
  <Capability Name = "internetClient" />
  <uap:Capability Name = "vodeosLibrary" />
  <DeviceCapability Name = "microphone" />
  <DeviceCapability Name ="webcam" />
</Capabilities>
```

- A. Option A
- B. Option B
- C. Option C
- D. Option D

Answer: D

From scenario:

The app must use the microphone and webcam to support audio and video recording. In addition, the app must support the use of the back and camera buttons, if present.

Users must be able to record videos and view them in their videos library.

Question: 41

HOTSPOT

You need to implement the following code segment according to the requirements.

How should you complete the code? To answer, select the appropriate code segment from each list in the answer area.

Answer Area

```
if ( ApiInformation . IsTypePresent ( "Windows.Phone.UI.Input.HardwareButtons" ) )  
{  
    ...  
}
```

Answer:

Explanation:

Answer Area

```
if ( ApiInformation . IsTypePresent ( "Windows.Phone.UI.Input.HardwareButtons" ) )  
{  
    ...  
}
```

From scenario:

Box 1: IsTypePresent

Box 2: Windows.Phone.UI.Input.HardwareButtons

Here is an example. We'll assume our app wants to take advantage of a Mobile device's dedicated camera button. If I directly referenced the HardwareButtons object for the CameraPressed event while on a desktop without checking that HardwareButtons is present, my app would crash.

// Note: Cache the value instead of querying it more than once.

bool isHardwareButtonsAPIPresent =

Windows.Foundation.Metadata.ApiInformation.IsTypePresent("Windows.Phone.UI.Input.HardwareButtons");

Question: 42

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution. Determine whether the solution meets the stated goals.

You must create a project for shared code.

Solution: You implement the shared code in a Class Library (Universal Windows).

Does this meet the goal?

- A. Yes
- B. No

Answer: A

Explanation:

The .NET Framework Portable Class Library project type in Visual Studio helps you build cross-platform apps and libraries for Microsoft platforms quickly and easily.

Portable class libraries can help you reduce the time and costs of developing and testing code. Use this project type to write and build portable .NET Framework assemblies, and then reference those assemblies from apps that target multiple platforms such as Windows and Windows Phone.

Even after you create a Portable Class Library project in Visual Studio and start developing it, you can change the target platforms. Visual Studio will compile your library with the new assemblies, which helps you identify the changes you need to make in your code.

From scenario:

The app must be compatible with current and future XBOX apps that use C++.

Fabrikam inc

Background:

Fabrikam is a commercial bank. The primary customers are individuals and employers with up to 10,000 employees. Fabrikam provides Internet banking services to customers.

You are developing a Universal Windows Platform (UWP) app for Fabrikam that extends the Internet banking to a UWP app.

Business Requirements:

Core functionality

Users must be able to access accounts, view balances, view recent transactions, and deposit checks by using the UWP app.

Usability

The app must use industry proven design patterns across the app. All navigational elements must be visible at all times.

Security

The app must provide secure transactions to protect customer privacy.

Technical Requirements

Data

The app must use a file based database. You must use a code first entity framework approach.

User interface

- You must use a Model-View-ViewModel (MVVM) pattern.
- Users must be able to access all content through the top-level navigation after they sign in
- The app must allow the user to upload up to 50 images (front and back) of checks to deposit.
- During the upload process, the app must be responsive to any other user actions.
- The app must only upload images when no other pending inputs are in the queue.

You must implement the following pages:

Page	Description
Sign-In	This page displays when the app is launched. It prompts users to enter credentials.
Transactions	This page allows users to view transactions for the last 30 days.
Balances	This page allows users to view current balance amount for all accounts.
Deposit	This page allows users to deposit checks by uploading images of checks.
Statements	This page lists the available bank statements for all accounts.

Network and web service

The app must meet the following requirements related to networking and web services:

Connect to a web service over a secure HTTP connection to upload images.

- Connect to Fabrikam's core web service to retrieve account information.
- Use networking technology already available in the .Net Framework.
- Consume the JSON that the Fabrikam core web service provides.

User data and alerts

The app must meet the following requirements related to user data and alerts:

Download new monthly bank statements when possible. The download process must not affect the performance of the app.

Network and web service

The app must meet the following requirements related to networking and web services:

- Connect to a web service over a secure HTTP connection to upload images.
- Connect to Fabrikame's core web service to retrieve account information.
- Use networking technology already available in the .Net Framework.
- Consume the JSON that the Fabrikam core web service provides.

User data and alerts

The app must meet the following requirements related to user data and alerts:

- Download new monthly bank statements when possible. The download process must not affect the performance of the app.
- Report to the user when the statements are downloaded to the device.
- Write a log entry when statements downloads are not successful.
- Periodically check for user activity and automatically log the user out when there is no activity for more than 15 minutes.

Security:

The app must meet the following requirements related to security.

Use a multi-factor authentication (MFA) by using email and a verification code to identify the user.

Security store credentials and retrieve credentials.

Automatically sign in the user irrespective of the device that is used to sign in to the app.

Store the resource name within the app itself.

Connect to an authentication app by using the URI schema fabrikam-secutity://oauth/.

AccountContext.cs

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 prefix that denotes the specific file to which they belong.

```

AC01 using Microsoft.EntityFrameworkCore;
AC02 using FabrikamApp.Model;
AC03 using System.Data.Common;
AC04 using System.Security;
AC05 namespace Fabrikam.Contexts
AC06 {
AC07
AC08     protected override void OnConfiguring(DbContextOptionsBuilder opt-
Builder
AC09     {
AC10         ...
AC11     }
AC12 }

```

ImageMeanger.cs

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 prefix that denotes the specific file to which they belong.

```

IM01 public static class ImageManager
IM02 {
IM03     private static CoreDispatcher _dispatcher;
IM04     public static void InitImageManager()
IM05     {
IM06         ...
IM07     }
IM08     public static void Upload(Action < List < byte []>> image)
IM09     {
IM10         ExecuteUpload(image).Wait();
IM11     }
IM12     private static Task ExecuteUpload(Action < List < byte []>> image)
IM13     {
IM14
IM15     }
IM16     private static void UploadImage(Action < List < byte []>> image)
IM17     {
IM18         ...
IM19     }
IM20 }

```

ClientProxy.cs

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 prefix that denotes the specific file to which they belong.


```

CP01 using System;
CP02 using System.IO;
CP03 using System.Net.Http;
CP04 using System.Runtime.Serialization.Json;
CP05 using System.Text;
CP06 using System.Threading.Tasks;
CP07 using FabrikamBanking.Model;
CP08
CP09 namespace FabrikamBanking.Services
CP10 {
CP11     public class ClientProxy
CP12     {
CP13         public async Task<Decimal> GetBalance(AccountRequest accountReq
CP14         {
CP15
CP16             var ms = new MemoryStream(Encoding.UTF8.GetBytes(result));
CP17             var data = (decimal)serializer.ReadObject(ms);
CP18
CP19             return data;
CP20         }
CP21     }
CP22 }

```

BkgTaskMgr.cs

```

BT01 public sealed class BackgroundTaskManager
BT02 {
BT03     public static BackgroundTaskRegistration RegisterBackgroundTask(string taskEntry-
Point, string taskName, IBackgroundTrigger trigger, IBackgroundCondition condition)
BT04     {
BT05         var builder = new BackgroundTaskBuilder();
BT06         builder.Name = taskName;
BT07         builder.TaskEntryPoint = taskEntryPoint;
BT08         builder.SetTrigger(trigger);
BT09         BackgroundTaskRegistration task = builder.Register();
BT10         task.Completed += new BackgroundTaskCompletedEventHandler(OnCompleted);
BT11         return task;
BT12     }
BT13
BT14     private static void OnCompleted(BackgroundTaskRegistration sender, BackgroundTask-
CompletedEventArgs args)
BT15     {
BT16         ...
BT17     }
BT18     public static IBackgroundTrigger GetTrigger()
BT19     {
BT20
BT21         return trigger;
BT22     }
BT23 }

```

CredentialManagers.cs

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.

```

CM01 using Windows.Security.Credentials;
CM02
CM03 namespace FabrikamBanking
CM04 {
CM05     class CredentialManager
CM06     {
CM07         private PasswordCredential GetCredentialFromLocker()
CM08         {
CM09             PasswordVault vault = new PasswordVault();
CM10             var credential = vault.RetrieveAll();
CM11             ...
CM12         }
CM13     }
CM14 }

```

MainPage.cs

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.

```

FB01 using System;
FB02 using Windows.UI.Xaml;
FB03 using Windows.UI.Xaml.Controls;
FB04 using System.Security;
FB05 namespace FabrikamBanking
FB06 {
FB07     public sealed partial class MainPage : Page
FB08     {
FB09         public MainPage()
FB10         {
FB11             this.InitializeComponent();
FB12         }
FB13         private async void LaunchAppURI(object sender,
RoutedEventArgs e)
FB14         {
FB15
FB16             ...
FB20         }
FB21     }
FB22 }

```

Question: 43

You need to insert code at line AC07 to create the database entities.

How should you complete the relevant code? To answer, drag the appropriate code segments to the correct location. Each code segments 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.

Code segments

```
public class AccountContext : DbContext
```

```
public class AccountContext : IRepository
```

```
public DbSet Accounts { get; set; }
```

```
public List Accounts { get; set; }
```

```
optBuilder.UseSqlite("Filename=Fabrikam.db", null)
```

```
optBuilder.UseModel("Filename=Fabrikam.db", null)
```



Answer Area

Code segment

```
{
```

Code segment

```
protected override void OnConfiguring(DbContextOptionsBuilder optBuilder)
```

```
{
```

Code segment

```
}
```

```
}
```

Answer:

Explanation:

Answer Area

```
public class AccountContext : DbContext
{
    public DbSet Accounts { get; set; }

    protected override void OnConfiguring(DbContextOptionsBuilder optBuilder)
    {
        optBuilder.UseModel("Filename=Fabrikam.db", null)
    }
}
```

From scenario:

The app must use a file based database. You must use a code first entity framework approach.

The DbContextOptionsBuilder Class provides a simple API surface for configuring DbContextOptions. Databases (and other extensions) typically define extension methods on this object that allow you to configure the database connection (and other options) to be used for a context.

A DbSet<TEntity> can be used to query and save instances of TEntity. LINQ queries against a DbSet<TEntity> will be translated into queries against the database.

UseModel(IModel) sets the model to be used for the context. If the model is set, then OnModelCreating(ModelBuilder) will not be run.

Question: 44

You need to configure authentication for the app.

Which two technologies should you use? Each correct answer presents part of the solution.

- A. Windows Hello
- B. Windows Kerberos
- C. Azure Active Directory
- D. Microsoft Passport

Answer: A,D

Explanation:

Microsoft Hello

Microsoft Hello provides simple multi-factor authentication using facial recognition (or iris, or fingerprints) that is used to access the Microsoft Passport private key stored in the secure TPM chip. For

the first time, Microsoft has included the biometric software (middleware) in Windows 10 to support biometrics for authentication. In previous versions of Windows, the OEM (HP, Dell, Lenovo, etc) needed to add its own biometric middleware to support biometric authentication.

From scenario: The app must meet the following requirements related to security:

Note: Microsoft Passport

Microsoft has resurrected the Passport moniker for a new PKI credential system that requires multi-factor authentication. Most interesting about Microsoft Passport is that it fully supports the Fast IDentity Online (FIDO) Alliance standards which means it will work with many web/cloud services without modification. The plan is that users of cloud services supporting FIDO is that there will no longer be passwords associated with the user's account.

Microsoft Passport involves a user logging onto the Windows 10 computer with multi-factor (PIN, face, iris, fingerprint, etc) and either creating a new account or associating an existing account with an IDentity Provider (IDP). Windows generates a public/private key pair with the private key stored securely outside of the Windows 10 OS. The public key is associated with the account so that a challenge can be sent that can only correctly respond to the IDP. Another key point to the Microsoft Passport credential system is that the user needs to enroll every device used to access the service (IDP).

Question: 45

DRAG DROP

You need to insert code at line BT19 to configure the trigger to meet the technical requirements for User Data and Alerts.

How should you complete the code segment? To answer, drag the appropriate code segments to the correct location or locations. Each code segments 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.

Code segments

false

TimeTrigger

SystemTrigger

UserPresent

Answer area

```
var trigger = new Code segment (15, Code segment );
```

Answer:

Explanation:

Answer area

```
var trigger = new TimeTrigger (15, false );
```

Box 1: TimeTrigger

Box 2: false

TimeTrigger(UInt32, Boolean) initializes a new instance of a time event trigger.

Syntax: public TimeTrigger(UInt32 freshnessTime, Boolean oneShot)

Parameters

freshnessTime UInt32

Specifies the number of minutes to wait before scheduling the background task. The system schedules the task within 15 minutes after freshnessTime elapses. If the OneShot property is false, freshnessTime specifies the interval between recurring tasks.

oneShot Boolean

True if the time event trigger will be used once; false if it will be used each time freshnessTime elapses

Question: 46

DRAG DROP

You need to add code at line IM14 to meet the user interface requirement.

How should you complete the relevant code segment? To answer, drag the appropriate code segments to the correct location or locations. Each code segments 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.

Code segments

BeginInvoke

CoreProcessEventsOption

CoreDispatcherPriority

RunAsync

Normal

High

Idle



Answer Area

```
await _dispatcher.  (  .  
 , () => UploadImage(image);
```

Answer:

Explanation:

Answer Area

```
await _dispatcher.  (  .  
 , () => UploadImage(image);
```

Box 1: RunAsync

Box 2: CoreDispatcherPriority

Box 3: Normal

If you are on a worker thread and want to schedule work on the UI thread, use `CoreDispatcher::RunAsync`. Always set the priority to `CoreDispatcherPriority::Normal` or `CoreDispatcherPriority::Low`.

Callbacks scheduled with `CoreDispatcherPriority::Low` priority are called when there are no pending input events. Use the `CoreDispatcherPriority::Low` priority to make your app UI more responsive.

Question: 47

DRAG DROP

You need to meet the storage requirements?

Which three actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

Actions

Install the SQLite package and define database entities.

Launch the NuGet Package Manager console.

Use Entity Framework migrations to create the database.

Run the **Enable-Migrations** Windows PowerShell cmdlet.

Install the MySQL package and define database entities.

Use MySQL to create the database.

Launch the Visual Studio 2015 command prompt.

Answer Area



Answer:

Explanation:

Answer Area

Launch the NuGet Package Manager console.

Install the SQLite package and define database entities.

Use Entity Framework migrations to create the database.

From scenario: The app must use a file based database. You must use a code first entity framework approach.

Install Entity Framework

To use EF Core, install the package for the database provider(s) you want to target. This walkthrough uses SQLite. For a list of available providers see Database Providers.

Tools -> NuGet Package Manager -> Package Manager Console

Run Install-Package Microsoft.EntityFrameworkCore.Sqlite

Create your database

Now that you have a model, you can use migrations to create a database for you.

Tools -> NuGet Package Manager -> Package Manager Console

Run Add-Migration MyFirstMigration to scaffold a migration to create the initial set of tables for your model.

Question: 48

HOTSPOT

You must evaluate the code in CredentialManager.cs.

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

Answer Area

Statements	Yes	No
The code retrieves credentials for all the users that have run the app.	<input type="radio"/>	<input type="radio"/>
The code uses Credential Locker to retrieve user credentials.	<input type="radio"/>	<input type="radio"/>
The code displays the user interface to collect user credentials.	<input type="radio"/>	<input type="radio"/>

Answer:

Explanation:

Answer Area

Statements	Yes	No
The code retrieves credentials for all the users that have run the app.	<input type="radio"/>	<input checked="" type="radio"/>
The code uses Credential Locker to retrieve user credentials.	<input checked="" type="radio"/>	<input type="radio"/>
The code displays the user interface to collect user credentials.	<input type="radio"/>	<input checked="" type="radio"/>

Box 1: No

Box 2: Yes

Box 3: No

From scenario: the following code was used

```
PassWordVault vault = new PasswordVault();  
var credential = vault.RetrieveAll();
```

You have several options for retrieving user credentials from the Credential Locker after you have a reference to the PasswordVault object.

You can, as in this question, retrieve all the credentials the user has supplied for your app in the locker with the PasswordVault.RetrieveAll method.

Question: 49

HOTSPOT

You need to insert code at line CP15 to retrieve the account balance for a user.

How should you complete the relevant code? To answer, select the appropriate code segment from each list in the answer area.

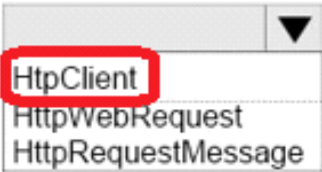
Answer Area

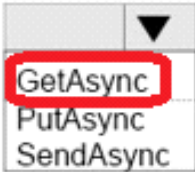
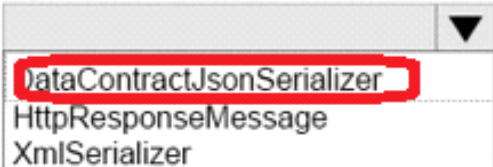
```
var objRequest = new ▼ ();  
HttpClient  
HttpWebRequest  
HttpRequestMessage  
  
var response = await objRequest.▼  
GetAsync  
PutAsync  
SendAsync  
  
( "http://core.fabrikam.com/accounts/balance?number=" +  
accountReq.AccountInfo.Number + "&token=" + accountReq.Token);  
var result = await response.Content.ReadAsStringAsync ();  
var serializer = new ▼ (typeof (decimal));  
DataContractJsonSerializer  
HttpResponseMessage  
XmlSerializer
```

Answer:

Explanation:

Answer Area

```
var objRequest = new  ();

var response = await objRequest. (
    "http://core.fabrikam.com/accounts/balance?number=" +
    accountReq.AccountInfo.Number + "&token=" + accountReq.Token);
var result = await response.Content.ReadAsStringAsync ();
var serializer = new  (typeof (decimal));
```

From scenario:

Users must be able to access accounts, view balances, view recent transactions, and deposit checks by using the UWP app.

Box 1: HttpClient

Box 2: GetAsync

Box 3: DataContractJsonSerializer

From scenario: Consume the JSON that the Fabrikam core web service provides.

Question: 50

You need to configure networking.

Which two networking technologies should you use? Each correct answer presents a complete solution.

- A. Background Transfer API
- B. StreamWebSocket class
- C. HttpClient class
- D. Custom WebSocket class
- E. MessageWebSocket class

Answer: A,C

Question: 51

DRAG DROP

You need to launch the authentication app.

How should you complete the relevant code? To answer, drag the appropriate code segments to the correct location or locations. Each code segments 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.

Code segments

UriKind.Absolute

UriKind.Relative

Launcher.UriAsync

Launcher.LaunchFolderAsync

Launcher.FindUriSchemeHandlersAsync

Uri

UriBuilder

• • • • •

Answer Area

```
var appURI = new [Code segment] ("fabrikam-security://oauth/ ", `
    [Code segment] );
var response = await [Code segment] (appURI);
```

Answer:

Explanation:

Answer Area









```
var appURI = new Uri [UriKind.Relative] ("fabrikam-security://oauth/ ", `
    [UriKind.Relative] );
var response = await [Launcher.UriAsync] (appURI);
```

Box 1: Uri
Box 2: UriKind.Relative
Box 3: Launcher.LaunchUriAsync

The Uri Constructor (Uri,

For More Information – **Visit link below:**
<http://www.certsgrade.com/>

PRODUCT FEATURES

-  **100% Money Back Guarantee**
-  **90 Days Free updates**
-  **Special Discounts on Bulk Orders**
-  **Guaranteed Success**
-  **50,000 Satisfied Customers**
-  **100% Secure Shopping**
-  **Privacy Policy**
-  **Refund Policy**



WE ACCEPT

