Virtual University	**CS411 Visual Programming Update/Current Final Term Solved Subjective 2023**	27/02/2023
100% Correct	**Solved by Jc FIA**	**VU**

Question#1: Example of jQuery post() method?

Answer: The post() method in jQuery is used to send an asynchronous HTTP (Ajax) request to the server, allowing you to load data from the server without having to refresh the entire page.

Example

```
post("example.php", { name: "John", time: "2pm" })
.done(function(data) {
alert("Data Loaded: " + data); });
```

Question#2: Write code to create and start thread in c#

Answer:

```
using System;
using System.Threading;

class Program
{
static void Main(string[] args)
{
Thread workerThread = new Thread(DoWork);
workerThread.Start();
Console.WriteLine("Main thread: starting worker thread...");
Console.ReadKey();
}

static void DoWork()
{
Console.WriteLine("Worker thread: working...");
}
}
```

Question#3: How to create listBox?

Answer: Here's an example of how you could create a ListBox control in Windows Forms using C#: csharp

```
using System.Windows.Forms;
namespace WindowsFormsApp
{
   public partial class Form1 : Form
   {
      public Form1()
      {
         InitializeComponent();

         ListBox listBox = new ListBox
         {
            Location = new System.Drawing.Point(10, 10),
            Size = new System.Drawing.Size(200, 100),
            Name = "listBox1"
         };
         this.Controls.Add(listBox);
      }
   }
}
```

Question#4: Write a code Asynchronous lambda expression will derived two number & returen the value.

Answer:

```
#import <Foundation/Foundation.h>
int main(int argc, const char * argv[]) {
   {
      int (^sum)(int, int) = ^(int a, int b) {
         return a + b;
      };
      NSLog(@"Sum of 2 and 3 is: %d", sum(2, 3));
   }
   return 0;
}
```

Question#5: Write a function in JavaScript that will add ten even numbers using a for loop.?

Answer: Here is an example of a function in JavaScript that adds ten even numbers using a for

```
loop:
function addTenEvenNumbers() {
  let sum = 0;
  for (let i = 0; i <= 20; i += 2) {
    sum += i;
  }
  return sum;
}

console.log(addTenEvenNumbers());
```

Question#6: What are the three major types in JavaScript?

Answer: In JavaScript, there are three major types:

1. **Primitive types**: Primitive types are the basic building blocks of data in JavaScript. There are six primitive types in JavaScript: undefined, null, boolean, number, string, and symbol.
2. **Object types**: Object types are complex data structures that can store and manipulate multiple values. There are several object types in JavaScript, including Object, Array, Map, Set, Date, and RegExp.
3. **Function types**: Function types are first-class objects in JavaScript that can be assigned to variables, passed as arguments to other functions, or returned from functions. Functions in JavaScript can also be assigned properties and methods, just like other objects.

Each type in JavaScript has its own set of properties, methods, and behaviors, and it's important to understand the differences between them to write effective and efficient code in JavaScript.

Question#7: **Write an equivalent C# code using task combinatory (ignore the exception)**

Answer: Here is an equivalent C# code that uses Task combinatory to run two tasks in parallel and wait for both of them to complete:

```
using System;
using System.Threading.Tasks;

namespace TaskCombinatorExample
{
   class Program
   {
      static void Main(string[] args)
      {
         Task task1 = Task.Run(() => {
            Console.WriteLine("Task 1 is running");
            // Some code here...
         });
```

```
        Task task2 = Task.Run(() => {
            Console.WriteLine("Task 2 is running");
            // Some code here...
        });

        Task.WhenAll(task1, task2).Wait();
        Console.WriteLine("Both tasks have completed");
    }
  }
}
```

Question#8: Create input field in html and JavaScript to calculate percentage. (5 Marks)

Answer: Here is an example of how to create an input field in HTML and JavaScript to calculate the percentage:

HTML

```
<!DOCTYPE html>
<html>
 <head>
  <title>Percentage Calculator</title>
 </head>
 <body>
  <div>
   <label for="number">Enter a number:</label>
   <input type="text" id="number">
  </div>
  <div>
   <label for="percent">Enter a percentage:</label>
   <input type="text" id="percent">
  </div>
  <div>
   <button id="calculate">Calculate</button>
  </div>
  <div id="result">
  </div>
  <script src="script.js"></script>
 </body>
</html>
```

JavaScript:

```
const numberInput = document.querySelector('#number');
const percentInput = document.querySelector('#percent');
const calculateButton = document.querySelector('#calculate');
const resultDiv = document.querySelector('#result');

calculateButton.addEventListener('click', () => {
  const number = parseFloat(numberInput.value);
  const percent = parseFloat(percentInput.value) / 100;
  const result = number * percent;

  resultDiv.textContent = `The result is ${result}.`;
});
```

Question#9: What will be the output of the code.

Async void displayprimecounts()
{ For (int i = 0; i < 10; i++)
Console.writeline(await getprimescountasync(i) } (5 Marks)

Answer: The code is missing some required components and will not run as-is. The **getprimescountasync** method is not defined and the **await** keyword can only be used within an async method. The code would also need to include a using System.Threading.Tasks; statement at the top in order to use the Task type.

Assuming that the **getprimescountasync** method returns a Task<int> and that the code is part of a larger program**, the output of the code would be the number of prime numbers that are less than or equal to each number from 0 to 9, printed to the console.** The numbers would be printed asynchronously, which means that the method would return immediately and the results would be printed to the console as they become available. The exact order in which the results are printed may not be predictable.

Question#10: What is the purpose of the following objective-C code

NSData *imageData = [NSURLconnection sendSynchronousRequest:urlRequest ReturningResponse:nil error:&downloadErro r]; (5 Marks)

Answer: This is Objective-C code that sends a synchronous request to a URL and stores the response data in an NSData object named "imageData". The request is made using the sendSynchronousRequest:returningResponse:error: method of the NSURLConnection class, which sends a URL request and returns the response data.

The first argument, urlRequest, is the URL request that needs to be sent. The second argument, nil, is a pointer to a variable that will store the response from the server. The third argument, &downloadError, is a reference to an error object that will be filled if there is an error with the request.

If the request is successful, the response data will be stored in the imageData object, which can then be used as needed in the code. If there is an error with the request, the error will be stored in the downloadError object, which can be checked to determine the cause of the error

Question#11: Consider the following code lines and explain what we are trying to achieve from them.
Msbuild/t:updateuid
Projectname.csproj.locBaml/parse
Projectname.g.en-US .resources/out.en US.csv (5 Marks)
Answer: These lines of code are related to building a Microsoft .NET project using MSBuild, a build automation tool used for building .NET projects.
The first line, Msbuild/t:updateuid, specifies the target of the build. The target updateuid updates the unique identifier (UID) of the project files to match the current values in the project file.
The second line, Projectname.csproj.locBaml/parse, specifies the parsing of the .locBaml file in the project. The .locBaml file contains the localization information for the project.

The third line, Projectname.g.en-US.resources/out.en-US.csv, specifies the output of the .resources file in the project. The .resources file contains the resource strings in a specific culture (in this case, the culture is en-US). The output is specified as a .csv file.
These lines of code demonstrate how to use MSBuild to perform specific tasks related to localization and resource management in a .NET project.

Question#12: What is the code of "for loop" to add the n even number in JavaScript?
Answer: The following is an example of a "for loop" in JavaScript that adds up the first n even numbers:

```
let n = 10; // number of even numbers to add up
let sum = 0; // variable to store the sum

for (let i = 2; i <= 2 * n; i += 2) {
  sum += i;
}
console.log("The sum of the first " + n + " even numbers is: " + sum);
```

Some Explain
In this code, the for loop starts from i = 2 and continues until i is less than or equal to 2 * n (the maximum value of n even numbers). The i variable is incremented by 2 in each iteration, as we are only interested in adding even numbers. The sum of the even numbers is stored in the sum variable, which is then displayed in the console.

Question#13: Create a single instance application using WPF. (5 Marks)

Answer: Here is an example of how you can create a single instance application using WPF in C#:

```
using System;
using System.Windows;

namespace SingleInstanceWPFApp
{
   public partial class App : Application
   {
      private static readonly Mutex mutex = new Mutex(true, "{Your Unique ID Here}");

      [STAThread]
      public static void Main()
      {
         if (mutex.WaitOne(TimeSpan.Zero, true))
         {
            App app = new App();
            app.InitializeComponent();
            app.Run();
            mutex.ReleaseMutex();
         }
         else
         {
            MessageBox.Show("Another instance of the application is already running.",
"Information", MessageBoxButton.OK, MessageBoxImage.Information);
            Application.Current.Shutdown();
         }
      }
   }
}
```

In this code, a mutex (short for "mutual exclusion") is created using the Mutex class. The mutex is used to ensure that only one instance of the application is running at a time.

Question#14: Write the difference between onetime binding and onewayTo Source binding. Also give example of both in XAML.

Answer: In WPF, there are two types of bindings: OneTime binding and OneWayToSource binding.

OneTime binding is used when the source of the binding is only needed to be read once and the target of the binding will not be updated when the source changes.

OneWayToSource binding is used when the target of the binding is only needed to be read once and the source of the binding will not be updated when the target changes.

Examples:
OneTime binding is XAML
<TextBlock Text="{Binding UserName, Mode=OneTime}" />

OneWayToSource binding in XAML
<TextBox Text="{Binding UserName, Mode=OneWayToSource}" />

Question#15: Given below in XAML code for stack panel. <StackPanel> </StackPanel> You are required to modify the given code to give the stack panel gradient background. Use three colors in gradient. (5 Marks)
Answer: An example of modifying the given XAML code to give the StackPanel a gradient background with three colors

```
<StackPanel>
  <StackPanel.Background>
    <LinearGradientBrush StartPoint="0,0" EndPoint="1,1">
      <GradientStop Color="Red" Offset="0" />
      <GradientStop Color="Yellow" Offset="0.5" />
      <GradientStop Color="Green" Offset="1" />
    </LinearGradientBrush>
  </StackPanel.Background>
</StackPanel>
```

Question#16: Write XAML code which rotates the button at the angle of 90 Degree? (5 Marks)
Answer: XAML code that rotates a Button 90 degrees.

```
<Button Content="Rotated Button">
  <Button.RenderTransform>
    <RotateTransform Angle="90"/>
  </Button.RenderTransform>
</Button>
```

Question#17: **Consider the XAML code**
<TextBox Name= “txt1”>< /TextBox>
<Button Name= “btn1”></Button>
Modify the given code so, whatever the text is written inside textbox it is automatically written as Button text.

Answer: how you can modify the XAML code to achieve the desired behavior:
<TextBox Name="txt1" TextChanged="txt1_TextChanged"></TextBox>
<Button Name="btn1"></Button>

Question#18: How can we create objects in objective-C? Also provide syntax.
Answer: In Objective-C, objects are created using the alloc and init methods. The alloc method is used to allocate memory for an object, and the init method is used to initialize the object. The basic syntax to create an object in Objective-C is as follows:
ClassName *objectName = [[ClassName alloc] init];

Question#19: Separate server side language and Client side language.
Answer: server-side languages

1. PHP
2. Ruby on Rails
3. Python
4. Java
5. Node.js
6. ASP.NET

Client-side languages

1. HTML
2. CSS
3. JavaScript
4. AJAX

Question#20: Write C# code which navigates to URI using the Navigate Method.
Answer: C# code that uses the Navigate method to navigate to a URI in a WebBrowser control:

```
using System.Windows.Forms;
namespace NavigateToUriExample
{
   public partial class Form1 : Form
   {
      public Form1()
      {
         InitializeComponent();
         webBrowser1.Navigate("https://www.example.com");
      }
   }
}
```

Question#21: What does it means?
[Locationmanager setdistanceFilter:kcldistancefilternone];
Answer: [LocationManager setDistanceFilter:kCLDistanceFilterNone]; is a line of code in Objective-C that sets the distance filter for the LocationManager object.
The LocationManager class is part of the Core Location framework and is used to manage the device's location and heading information.
The setDistanceFilter method sets the distance filter
The kCLDistanceFilterNone constant is a value that indicates that the location manager should deliver all location updates, regardless of their distance from the previous update.

Question#22: Give the syntax to hide all tags in html using JS and JQuery.?
Answer:
Here's the syntax to hide all <p> tags in HTML using JavaScript:

```
document.querySelectorAll("p").forEach(function(p) {
  p.style.display = "none";
  });
```

And here's the syntax to hide all <p> tags in HTML using JQuery:

```
$("p").hide();
```

Question#23: **What are the five types of HTML?** **5 marks**
Answer: Five types of html in use:

- HTML 4.01 Transitional
- HTML 4.01 Strict
- XHTML 1.0 Transitional
- XHTML 1.0 Strict
- HTML5

Question#24: How to define binary resource and what are the categories of binary resources in WPF?
Answer: In WPF, a binary resource is a type of resource that is stored in binary format, such as images, audio files, video files, and other types of files that are not text-based. These resources are typically embedded in the application or assembly, making them easily accessible to the application without needing to read from external files.
There are several Categories of binary resources in WPF
Icons: This includes icon files (.ico) that can be used to represent an application or its components.

Fonts: This includes TrueType fonts (.ttf) and OpenType fonts (.otf) that can be used to change the appearance of text in the application.
Binary files: This includes any other type of binary file that the application may need, such as configuration files or data files.

Question#24: Write the properties of touch point in C#

Answer: The properties that are available for a touch point object in C#

- Position,
- Size,
- Bounds,
- Touch device,
- Action (Down, Up, Move).

These properties can be used in C# code to perform actions based on touch input, such as moving or resizing objects on the screen or responding to touch events.

Question#25: Write the XAML code to create a list box with three items. **5 marks**

Answer:

```
<ListBox>
  <ListBoxItem>Item 1</ListBoxItem>
  <ListBoxItem>Item 2</ListBoxItem>
  <ListBoxItem>Item 3</ListBoxItem>
</ListBox>
```