-

CMR
INSTITUTE OF
TECHNOLOGY

USN | 1 | C | | | | | | | |

Internal Assessment Test II – April 2018

Sub:	**Advanced Web Programming ANSWER KEY**							**Code:**	16MCA42
Date:	19-04-18	Duration:	90 mins	Max Marks:	50	**Sem:**	IV	**Branch:**	**MCA**

Note: Answer any 5 questions. All questions carry equal marks. Total marks: 50

		Marks	OBE	
			CO	RBT
1a.	Explain Rails MVC Architecture with a neat diagram.	[10]	CO2	L4

The **M**odel **V**iew **C**ontroller principle divides the work of an application into three separate but closely cooperative subsystems.

Model (ActiveRecord)

It maintains the relationship between the objects and the database and handles validation, association, transactions, and more.

This subsystem is implemented in ActiveRecord library, which provides an interface and binding between the tables in a relational database and the Ruby program code that manipulates database records. Ruby method names are automatically generated from the field names of database tables.

View (ActionView)

It is a presentation of data in a particular format, triggered by a controller's decision to present the data. They are script-based template systems like JSP, ASP, PHP, and very easy to integrate with AJAX technology.

This subsystem is implemented in ActionView library, which is an **Embedded Ruby** (ERb) based system for defining presentation templates for data presentation. Every Web connection to a Rails application results in the displaying of a view.

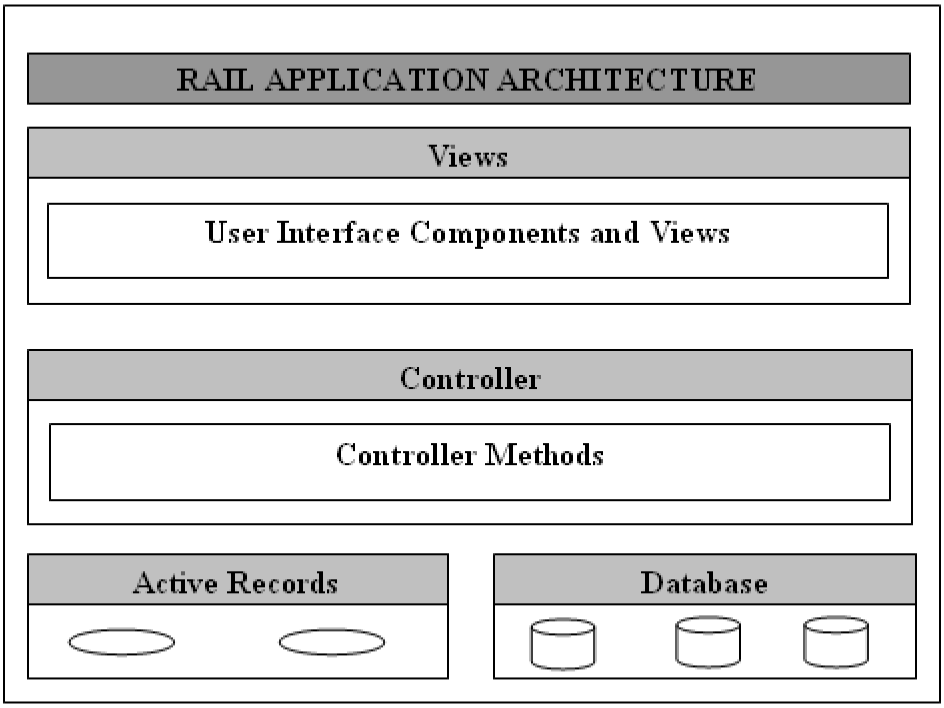
Controller (ActionController)

The facility within the application that directs traffic, on the one hand, querying the models for specific data, and on the other hand, organizing that data (searching, sorting, messaging it) into a form that fits the needs of a given view.

This subsystem is implemented in ActionController, which is a data broker sitting between ActiveRecord (the database interface) and ActionView (the presentation engine).

Pictorial Representation of MVC Framework

Given below is a pictorial representation of Ruby on Rails Framework –


RAIL APPLICATION ARCHITECTURE
Views
User Interface Components and Views
Controller
Controller Methods
Active Records
Database

2a.	Explain web services. Explain web service architecture in detail.	[10]	CO1	L4

Web services are open standard (XML, SOAP, HTTP etc.) based Web applications that interact with other web applications for the purpose of exchanging data. A web service is a collection of open protocols and standards used for exchanging data between applications or systems. Software applications written in various programming languages and running on various platforms can use web services to exchange data over computer networks like the Internet in a manner similar to inter-process communication on a single computer. Web services are self-contained, modular, distributed, dynamic applications that can be described, published, located, or invoked over the network to create products, processes, and supply chains. These applications can be local, distributed, or web-based. Web services are built on top of open standards such as TCP/IP, HTTP, Java, HTML, and XML.

A web service is a method of communication between two devices over a network. There are two ways to view the web service architecture

1. To examine the individual roles of each service actor
2. Examine the emerging web service protocol stack

Web Service Roles

There are three major roles within the web service architecture:

1. Service Provider

This is the provider of the web service. The service provider implements the service and makes it available on the Internet.

2. Service Requestor

This is any consumer of the web service. The requestor utilizes an existing web service by opening a network connection and sending an XML request.

3. Service Registry

This is a logically centralized directory of services. The registry provides a central place where developers can publish new services or find existing ones. It therefore serves as a centralized clearing house for companies and their services.

Web Service Protocol Stack

A second option for viewing the web service architecture is to examine the emerging web service protocol stack. The stack is still evolving, but currently has four main layers.

1. Service Transport

This layer is responsible for transporting messages between applications. Currently, this layer includes Hyper Text Transport Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), File Transfer Protocol (FTP), and newer protocols such as Blocks Extensible Exchange Protocol (BEEP).

2. XML Messaging

This layer is responsible for encoding messages in a common XML format so that messages can be understood at either end. Currently, this layer includes XML-RPC and SOAP.

3. Service Description

This layer is responsible for describing the public interface to a specific web service. Currently, service description is handled via the Web Service Description Language (WSDL).

4. Service Discovery

This layer is responsible for centralizing services into a common registry and providing easy publish/find functionality. Currently, service discovery is handled via Universal Description, Discovery, and Integration (UDDI).

As web services evolve, additional layers may be added and additional technologies may be added to each layer.

3a. Explain arrays in Ruby. With examples, explain the various array functions. [10] CO3 L4

Ruby arrays are ordered, integer-indexed collections of any object. Each element in an array is associated with and referred to by an index.

Ruby arrays can hold objects such as String, Integer, Fixnum, Hash, Symbol, even other Array objects. Ruby arrays are not as rigid as arrays in other languages. Ruby arrays grow automatically while adding elements to them.

Creating Arrays:

There are many ways to create or initialize an array. One way is with the *new*class method:

```
names = Array.new
```

You can set the size of an array at the time of creating array:

names = Array.new(20)

```
#!/usr/bin/ruby


names = Array.new(20)

puts names.size  # This returns 20

puts names.length # This also returns 20


#!/usr/bin/ruby

names = Array.new(4, "mac")

puts "#{names}"
```

Array Built-in Methods:

```
#!/usr/bin/ruby

digits = Array(0..9)

num = digits.at(6)

puts "#{num}"
```

To initialize an array

```
a = [ "a", "b", "c" ]

n = [ 65, 66, 67 ]
```

Hashes

A Hash is a collection of key-value pairs like this: "employee" = > "salary". It is similar to an Array, except that indexing is done via arbitrary keys of any object type, not an integer index.

The order in which you traverse a hash by either key or value may seem arbitrary and will generally not be in the insertion order. If you attempt to access a hash with a key that does not exist, the method will return *nil*.

```
#!/usr/bin/ruby


H = Hash["a" => 100, "b" => 200]


puts "#{H['a']}"

puts "#{H['b']}"
```

Hash Built-in Methods

We need to have an instance of Hash object to call a Hash method. As we have seen, following is the way to create an instance of Hash object

–

```
#!/usr/bin/ruby


$, = ", "

months = Hash.new( 12 )

months = {"1" => "January", "2" => "February"}


keys = months.keys

puts "#{keys}"
```

4a. Build a Rails application to find the product of two integers and return the result to the client. [10] CO3 L4

Step 3 : Click on "Git bash" The CMD prompt points to $ with "/c/sites " directory
Step 4: create new bookstore using the following syntax
$ rails new addnumbers

Step 5: **C**hange the directory the following syntax
$ **cd addnumbers**

Step 6: Run the server using following command
$ rails server
Then open Firefox Web browser and type http://localhost:3000 Which gives ruby on rails welcome page.

Step 9: Use rake command to flush function paths to routes.rb
$ rake routes

Step 10: Run the server again
$ rails server

Step 11: Creating a new controller which helps to enter numbers
$ rails generate controller addnu result

Go to C:\Sites\addnumbers\app\controllers\ addnu_controller.rb
Add this code in this page

```
class AddnuController < ApplicationController
 def result
 @a = params[:a]
 @b = params[:b]
 @c = @a.to_i * @b.to_i
 end
end
```

Go to C:\Sites\addnumbers\app\views\addnu\ result.html.erb
Add this code here

```
<html>
```

```
<title> welcome template for books </title>
<body>
<p> sum = <%=@c%> </p>
<form action="result">
<input type="text" name="a" />
<input type="text" name="b" />
<input type=submit value="search" />
</form>
</body>
</html>
```

Run the server again using

Go to Gitbash and type
rails server

Go to browser and type http://localhost:3000/addnu/result

5a	What is social networking. Briefly explain about folksonomy.	[5]	CO1	L4

A folksonomy is a system in which users apply public tags to online items, typically to aid them in re-finding those items. This can give rise to a classification system based on those tags and their frequencies. A folksonomy emerges when users tag content or information, such web pages, photos, videos, podcasts, tweets, scientific papers and others.
Folksonomies are a trade-off between traditional centralized classification and no classification at all.

Advantages:

- tagging is easy to understand and do, even without training and previous knowledge in classification or indexing
- the vocabulary in a folksonomy directly reflects the user’s vocabulary
- folksonomies are flexible, in the sense that the user can add or remove tags
- tags comprises both popular content and long-tail content, facilitating users to browse and discover new content even in narrow topics
- folksonomies are multi-dimensional, in the sense that users can assign any number and combination of tags to express a concept

Folksonomies consist of three basic entities: users, tags, and resource. Users create tags to mark resources such as: web pages, photos, videos, and podcasts. These tags are used to manage, categorize and summarize online content. This collaborative tagging system also uses these tags as a way to index information, facilitate searches and navigate resources. Folksonomy also includes a set of URLs that are used to identify resources that have been referred to by users of different websites. These systems also include category schemes that have the ability to organize tags at different levels of granularity

The social network is a theoretical construct useful in the social sciences to study relationships between individuals, groups, organizations, or even entire societies (social units, see differentiation).

	A social network is a social structure made up of a set of social actors (such as individuals or organizations), sets of dyadic ties, and other social interactions between actors. The social network perspective provides a set of methods for analyzing the structure of whole social entities as well as a variety of theories explaining the patterns observed in these structures. Social networks and the analysis of them is an inherently interdisciplinary academic field which emerged from social psychology ,sociology, statistics, and graph theory. The term is used to describe a social structure determined by such interactions. The ties through which any given social unit connects represent the convergence of the various social contacts of that unit.			

b	Write function definition in Ruby for swapping two numbers.
	Def swap()
	a=gets.to_i
	b=gets.to_i
	t=a
	a=b
	b=t
	puts “a=#{a}”
	puts “b=#{b}”
	end
	swap()
		[5]	CO3	L4

6a. What Is JSON. Explain the various literals used in JSON with examples. [10] CO1 L4

JSON: **J**ava**S**cript **O**bject **N**otation. JSON is a syntax for storing and exchanging data. JSON is an easier-to-use alternative to XML. A common use of JSON is to read data from a web server, and display the data in a web page.

JSON Arrays Literals

JSON arrays are written inside square brackets.

Just like JavaScript, a JSON array can contain multiple objects:

```
{"employees":[
  {"firstName":"John", "lastName":"Doe"},
  {"firstName":"Anna", "lastName":"Smith"},
  {"firstName":"Peter", "lastName":"Jones"}
]}
```

The JSON format is syntactically identical to the code for creating JavaScript objects. Because of this similarity, instead of using a parser (like XML does), a JavaScript program can use standard JavaScript functions to convert JSON data into native JavaScript objects.

For AJAX applications, JSON is faster and easier than XML:

Using XML

- Fetch an XML document
- Use the XML DOM to loop through the document
- Extract values and store in variables

Using JSON

- Fetch a JSON string
- JSON.Parse the JSON string

JSON Syntax Rules

JSON syntax is derived from JavaScript object notation syntax:

- Data is in name/value pairs
- Data is separated by commas
- Curly braces hold objects
- Square brackets hold arrays

JSON values can be:

- A number (integer or floating point)
- A string (in double quotes)
- A Boolean (true or false)
- An array (in square brackets)

- An object (in curly braces)
- null

JSON format supports the following data types –

Type	Description
Number	double- precision floating-point format in JavaScript
String	double-quoted Unicode with backslash escaping
Boolean	true or false
Array	an ordered sequence of values
Value	it can be a string, a number, true or false, null etc
Object	an unordered collection of key:value pairs
Whitespace	can be used between any pair of tokens
null	empty

String Literals

- It is a sequence of zero or more double quoted Unicode characters with backslash escaping.
- Character is a single character string i.e. a string with length 1.

The table shows string types –

Type	Description
"	double quotation
\	reverse solidus
/	solidus
b	backspace

f	form feed
n	new line
r	carriage return
t	horizontal tab
u	four hexadecimal digits

JSON Mixed Literals

It is possible to mix object and array literals creating an array of objects or an object containing an array.

```
Var cars = [
{
“color”:”red”,
“doors”:2,
“paid”:true
},
{
“color”:”blue”,
“doors”:4,
“paid”:false
}
];
```

The array cars has 3 objects, The three objects has properties named color, doors, paid.
Alert(cars[1].doors) will display 2

7a. Write a JQuery script to insert text into a paragraph. [5] CO1 L4

```
<!DOCTYPE html>
<title>My Example</title>

<script src="jquery.min.js"></script>
<script>
  $( function() {
    $( "button" ).click( function() {
      $( "p" ).text( "contents" );
    });
  });
</script>

<button>Click me!</button>
  <p></p>
```

b Write a JQuery script to search for a nested list in an unordered list and highlight them on click of a button. [5] CO1 L4

```
<!doctype html>
<title>Example</title>
<script src="jquery.min.js"></script>
<script>
  $( function() {
    $( "button" ).click( function() {
      $( "ul" ).find( "li ul" ).css( "background-color", "yellow" );
    });
  });
</script>

<button>Highlight all nested lists</button>

<ul>
  <li>Apples
    <ul>
      <li>Big ones</li>
      <li>Juicy ones</li>
      <li>Red ones</li>
    </ul>
  </li>
  <li>Oranges</li>
  <li>Papaya
    <ul>
      <li>Papaya carica</li>
      <li>Carica</li>
      <li>Carica pubescens</li>
    </ul>
  </li>
  <li>Lychee</li>
  <li>Rambutan</li>
  <li>Passionfruit</li>
  <li>Mango</li>
</ul>
```

8a. Write short notes on SOAP, WSDL and REST services. [10] CO1 L4

SOAP

SOAP is an XML-based protocol for exchanging information between computers.

- SOAP is a communication protocol.
- SOAP is for communication between applications.
- SOAP is a format for sending messages.
- SOAP is designed to communicate via Internet.
- SOAP is platform independent.
- SOAP is language independent.
- SOAP is simple and extensible.
- SOAP allows you to get around firewalls.
- SOAP will be developed as a W3C standard.

WSDL

WSDL is an XML-based language for describing web services and how to access them.

- WSDL stands for Web Services Description Language.
- WSDL was developed jointly by Microsoft and IBM.
- WSDL is an XML based protocol for information exchange in decentralized and distributed environments.
- WSDL is the standard format for describing a web service.
- WSDL definition describes how to access a web service and what operations it will perform.
- WSDL is a language for describing how to interface with XML-based services.
- WSDL is an integral part of UDDI, an XML-based worldwide business registry.
- WSDL is the language that UDDI uses.

REST

Representational State Transfer (REST) is an architectural **style** that specifies constraints, such as the uniform interface, that if applied to a web service induce desirable properties, such as performance, scalability, and modifiability, that enable services to work best on the Web.

RESTful Web Services are REST architecture based web services. In REST Architecture everything is a resource. RESTful web services are light weight, highly scalable and maintainable and are very commonly used to create APIs for web based applications.

In REST architecture, a REST Server simply provides access to resources and REST client accesses and presents the resources. Here each resource is identified by URIs/ global IDs. REST uses various representations to represent a resource like text, JSON and XML. Now a days JSON is the most popular format being used in web services.

HTTP Methods

Following well known HTTP methods are commonly used in REST based architecture.

- **GET** - Provides a read only access to a resource.
- **PUT** - Used to create a new resource.
- **DELETE** - Used to remove a resource.
- **POST** - Used to update a existing resource or create a new resource.

Properties of a REST Application

The REST style is characterized by the following properties:

Communication takes place on call. The Client is active and requests a representation from the passive server and/or modifies a resource.
A resource can be addressed by an unique URI.
The client can request the representation of a resource in form of a document.

	Representations can refer to further resources. The server does not monitor the status of its clients. Each query to the server must contain all information that are necessary for interpreting itself. Caching is supported. The server can mark its answers as cacheable or not cacheable.			