



- KEEPING MOBILE PHONE/ANY ELECTRONIC GADGETS, EVEN IN 'OFF' POSITION IS TREATED AS EXAM MALPRACTICE
- DON'T WRITE ANYTHING ON THE QUESTION PAPER

Answer ALL Questions

(10 X 10 = 100 Marks)

1. Write the HTML code to design the following table.

Tree Characteristics						
		John Smith & Co.				
		Tree				
		Pine	Maple	Oak	Fir	
		Average Height (feet)	55	50	50	65
		Average Width (inches)	18	26	24	28
Characteristic	Typical Lifespan (years)	150	250	310	135	
	Leaf Type	Long needles	Broadleaf	Split leaf	Short needles	

2. Write the HTML and CSS code to design the following Layout:

[Home](#)
[About](#)
[Blog](#)
[Careers](#)
[Contact Us](#)

This is the heading of the article

Lorem ipsum dolor sit amet, consectetur adipiscing elit. In erat mauris, faucibus quis pharetra sit amet, pretium ac libero. Etiam vehicula eleifend bibendum. Morbi gravida metus ut sapien condimentum sodales mollis augue sodales. Vestibulum quis quam at sem placerat aliquet. Curabitur a felis at sapien ullamcorper fermentum. Mauris molestie arcu et lectus iaculis sit amet eleifend eros posuere. Fusce nec porta orci.

Aside Heading

Mauris molestie arcu et lectus iaculis sit amet eleifend eros posuere. Fusce nec porta orci.

[Home](#)
[About](#)
[Blog](#)
[Careers](#)
[Contact Us](#)

Copyright © 2017 My Site

3. For the given form layout below, write ONLY the JavaScript code to perform the form validation using a regular expression on the following fields :

- First Name and Last Name should not exceed 30 characters and can be only alphabets.
 - The phone number should not exceed 10 digits.
 - Age and Resident state should be selected or checked.
 - Validate for a valid Email address.
 - Comments should be alphanumeric
- Display the possible validation error messages.

First Name:

Last Name:

E-Mail Address:

Phone Number:

Resident State: NJ ▼

Age Group:

☒ 0-20
☐ 20-40
☐ 41-60
☐ Over 60

Membership:

☒ AAA
 ☐ Costco
 ☐ IEEE
 ☐ ACM

Comments:

Register

Start Over

4. Write the JQuery code to develop the web application to add event behavior to the following web page to simulate creating a single paperclip whenever the "Create Paperclip" (#paperclip-it) button is clicked.

Paperclip-It

Paperclips: 49

Highscore: 49

Create Paperclip

Paperclip-It

Paperclips: 50

Highscore: 50

Create Paperclip

Feeling Lucky?

Before user unlocks "Feeling Lucky" button

After user unlocks "Feeling Lucky" button

- When the page is initially loaded, the user has 0 paperclips. When the button is clicked, the current paperclip count (in #count) should be incremented by 1.
- Feeling Lucky:**
- The first time users generate 50 paperclips, they unlock the "Feeling Lucky" option, which should display the #secret-div (this div is already hidden with a .hidden class when the page is loaded).

- iii) When visible, the user can test their luck at the paperclip jackpot by clicking on the "Feeling Lucky" button.
- iv) When a user clicks "Feeling Lucky" then there should be a 25% chance that the user will double their paperclips, and a 75% chance that they instead lose all their paperclips (the "Feeling Lucky" button remains visible).

Highscores:

- v) Below the paperclip count is the user's current highscore (the highest number of paperclips they have ever had).
 - vi) You should update the high score (in #highscore) whenever the current paperclip count increases and is higher than the current high score.
 - vii) For example, if a user's highscore is 12 paper clips and they go from having 12 to 13 paper clips, their highscore should become 13 as well. If their highscore was instead 57 and they went from 12 to 13 paper clips, the highscore would remain 57.
5. Describe the MVC Architecture with a real time scenario example.
6. Write the code to develop the AJAX web application that displays a list of possible flight prices and carriers by fetching the JSON data from the server.
- a) The page contains two text boxes where the user can specify the start and end location, a checkbox that checks if they want only see non-stop flights (flights with 0 stops), and a "Go!" button.
 - b) When the "Go!" button is clicked, clear previous results, and read the JSON data.
 - i) Add an h1 to the results div containing the text "Flights from" and then the start and destination locations.
 - ii) Turn each flight's data into a paragraph in the results div.
 - iii) In each paragraph, write the price of the ticket (with a "\$") followed by the word "from", the carrier's name, and then the word "with", the number of stops, and the word "stops". If the flight has a price below 1000 display its row in bold.
 - c) The JSON data returned has a list of flights associated with it. The list of flights contains lists that each comprises a price, a carrier, and the number of stops
- JSON format:
- ```
{
 "Edinburgh": {
 "start": "Seattle",
 "flights": [
 { "carrier": "Delta", "price": 812, "stops": 2 },
 { "carrier": "Air France", "price": 1020, "stops": 0 },
 { "carrier": "Air France", "price": 1190, "stops": 3 }
]
 },
 "New York": {
 "start": "Seattle",
 "flights": [
 { "carrier": "British Airlines", "price": 782, "stops": 1 },
 { "carrier": "Delta", "price": 1562, "stops": 2 },
 { "carrier": "United", "price": 957, "stops": 1 },
 { "carrier": "KLM", "price": 687, "stops": 3 },
 { "carrier": "KLM", "price": 1458, "stops": 1 }
]
 }
}
```

Start location: seattle

End location: Edinburgh

Non-stop? ☐ Go!

## Flights from seattle to Edinburgh

S812 from Delta with 2 stops

S1020 from Air France with 0 stops

S1190 from Air France with 3 stops

Start location: seattle

End location: Edinburgh

Non-stop? ☒ Go!

## Flights from seattle to Edinburgh

S1020 from Air France with 0 stops

- 7.(a) Write the code in NodeJS and expressJS using Sessions for a Login form page that will output a form containing two fields: username and password.
- Upon submission of the form, the code should check whether the username-password pair was correct.
    - If so, display a welcome web page.
    - If not, display the message "Invalid username or password" followed by the same login form.
  - Add a "Log Out" button to the welcome message page that, when clicked, ends the session so that the user is logged out.
  - Clicking the button should redirect the user to the same login form. This behavior should be consistent with the lifetime of sessions.
  - That is, the user should stay logged in despite refreshing and opening the same page in multiple tabs, but should be logged out once the browser is closed.

OR

- 7.(b) Write the code to develop a web application using NodeJS and ExpressJS to get details from the HTML form of the Student details. Student details such as Student ID, Student name, Programme, Six-course marks, and Six Course credit hours. to calculate the grade point average(GPA) for a semester by the following formula:

$$\text{GPA} = ((\text{Mark}-1 * \text{Credit}-1) + \dots + (\text{Mark}-6 * \text{Credit}-6)) / \text{TC}$$

$$\text{Total Credit hours(TC)} = \text{Credit}-1 + \dots + \text{Credit}-6.$$

Generate the Marks statement with the student details and the calculated GPA.

- 8.(a) Write the code to develop a web application using NodeJS and ExpressJS.

| Name         | IPL Franchise               | Country      | Bid_Amount |
|--------------|-----------------------------|--------------|------------|
| M.S.Dhoni    | Rising Pune Super Gaints    | India        | 500000     |
| Raina        | Gujarat Lions               | India        | 50000      |
| Bravo        | Gujarat Lions               | West Indies  | 200000     |
| Chris Gayle  | Royal Challengers Bangalore | West Indies  | 100000     |
| du Plessis   | Rising Pune Super Gaints    | South Africa | 150000     |
| Virat Kohli  | Royal Challengers Bangalore | india        | 200000     |
| David Warner | Sunrisers hyderabad         | Australia    | 100000     |
| Sunil Narine | Kolkota Knight Riders       | SriLanka     | 160000     |

- i) Write ONLY the ExpressJS code to get details to create the above collection IPL and insert the documents into the collection in MongoDB.

- ii) Using the ExpressJS application display the following query in an ordered list with CSS.
- 1) All the player names, country, bid amounts where the IPL franchise from "Gujarat Lions" and player name starts or begins with "r". Order the results alphabetically by player name.
- 2) All the player names, country, IPL franchise where the Indian players and bid amount greater than 10 lakhs to 20 lakhs. Order the results by bid amount from lowest to highest bid amount.

OR

8.(b) Write the code to develop a web application using NodeJS and ExpressJS

| Menu |                                     |       |      |          |             |
|------|-------------------------------------|-------|------|----------|-------------|
| mid  | item                                | price | cost | category | subcategory |
| 1    | Brewed Coffee (Black)               | 1.25  | 0.65 | Drink    | Coffee      |
| 2    | Espresso                            | 3.75  | 1.15 | Drink    | Coffee      |
| 3    | Jeremy's Unicorn Frappe             | 8.00  | 2.75 | Drink    | Coffee      |
| 4    | Bubble Tea                          | 3.00  | 1.75 | Drink    | Tea         |
| 5    | Blueberry Muffin                    | 3.00  | 1.20 | Bakery   | Muffin      |
| 6    | Honey Toast                         | 4.00  | 1.85 | Bakery   | Bread       |
| 7    | Chocolate Chip Cookie               | 1.25  | 0.50 | Bakery   | Cookie      |
| 8    | Chocolate Chip Cookie (Gluten-Free) | 1.25  | 0.60 | Bakery   | Cookie      |
| ...  | ...                                 | ...   | ...  | ...      | ...         |

- i) Write ONLY the ExpressJS code to get details to create the above collection Menu and insert the documents into the collection in MongoDB.
  - ii) Using the ExpressJS application display the following query in an ordered list with CSS
    - 1) All the item, category, mid, price and cost for which the subcategory is "Coffee". Order the results by price value increasing numerically.
    - 2) All the item, mid, category, and subcategory where the for which the item name starts or ends with "e". Order the results by mid value decreasing numerically.
9. Write the code to design the React-based Controlled form components for capturing the details of a book. Properties of the book include book name, author name, publisher name, Edition (From 1 to 10), book genre (like science, fiction, comedy, and horror), and price. On submission of the form display the details captured in JSON format in an alert box. Ensure the form is displayed with default values.
10. Assume that the database 'test' is available in MongoDB with the collection 'student'. It has fields like register number (VIT register number format), name, cgpa, email id, college town, and native town. Write the MongoDB shell commands for the following:
- a) Insert 4 documents into the collection through a single query.
  - b) List the students who has joined in the year '2022'.
  - c) Increase the CGPA by 0.01 for students with more than 8 CGPA.
  - d) List the students who have either come from 'Vellore' city or studying in 'Vellore'.
  - e) List the student details whose email has the word "stud".

Ca/D/TY