



- 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 below form layout:

Application form

Full Name	Aryan
Email	aryan@domain.com
Gender	☒ Male ☐ Female
Nationality	Indian
Skills	☒ Asp.Net ☒ jQuery ☐ MVC ☒ Sql Server
Post applying for	☐ Trainee ☒ Software Engineer ☐ Team Leader ☐ Project manager
Upload Resume:	Choose File banner.png
Paste your resume	Sample resume text...
☒ Notify me the status via Email ☐ I agree to terms & conditions	
Submit Reset	

2. Write the HTML and CSS code to design the following Layout and choose your choice of background color for each division:

Name of Site

Section Title

menu item 1
menu item 2
menu item 3
menu item 4
menu item 5
menu item 6

An example using non-semantic mark-up

This is one of two sample pages created to demonstrate the difference between pages using semantic HTML mark-up and those that don't.

What you see isn't always what you get.

If you compare this page with the one using semantic mark-up, you won't see significant differences in your browser.

But what's on the inside can make a world of difference for accessibility, search engine optimization, and workflow. Read the accompanying blog entry to learn more.

© 2009 Heidi Adams Cool · Web Design & Strategy · Cleveland, OH, USA · About this site · Contact Me · Search

3. Write the JavaScript code to add behavior to the following HTML web page. The page is similar to the number guessing game. The following screenshots show the initial state and state after several guesses have been made.

Guessing Game

Your past guesses:

Guessing Game

 Too low!

Your past guesses:

- 50
- 25
- 37

Guessing Game

 You got it right in 6 tries!

Your past guesses:

- 50
- 25
- 37
- 43
- 40
- 42

When the page loads, your code should randomly choose a number from 1-100 inclusive.

- When the user types a number into the number text field and then clicks the make guess button the game will compare the user's guess to your randomly chosen number, and report whether it was "Too high!", "Too low!", or correct.
- The information will be shown as text in the result span. If the guess is correct, you should show a message such as, "You got it right in 6 tries!" (You can still say "tries" even if it takes 1 guess.) Once the user guesses correctly, disable the button so that no more guesses can be made.
- The game will also show a history of all guesses made as a bulleted list. Each guess's number is added as a bullet to the end of the list. If the guess was too low, this bullet should use the low CSS class, which displays it in blue italic. If it's too high, the bullet should use the high CSS class, which is red and bold. If it is correct, it should be displayed without any class.

4. For the given form layout in Question No. 1, write ONLY the JQuery code to perform the form validation using a regular expression in the following fields:

- Full Name should not exceed 20 characters and can be the only alphabets.
- Gender and Nationality should be selected or checked.
- Valid email address.
- Skills, Post applying for and I agree to terms and conditions fields should be checked.
- Paste Your resume field should be alphanumeric.

Display the possible validation error messages.

5. Write the AngularJS code for developing an application that displays the Shopping cart. It should calculate the amount based on the quantity given by the user. When the user clicks the remove button, the particular item should be removed from the cart.

Shopping Cart					
Quantity	Item	Order Code	Unit Price	Total	Remove
2	My Product (HSD-KSE)	HSD-KSE	\$9 99	\$19 98	Remove
1	My Product (KLS-IEN)	KLS-IEN	\$9 99	\$9 99	Remove
4	My Product (ELS-OWK)	ELS-OWK	\$9 99	\$39 96	Remove
3	My Product (KVL-JFE)	KVL-JFE	\$9 99	\$29 97	Remove

6. Enumerate the HTTP Request and Response message and its methods.

- 7.a) Write an Express JS web application with different URLs manage the cookie process as given below:
- A form to collect the cookie name and value.
 - A button either to insert cookie if it is not available or update the value of the cookie if it is existing. Display the corresponding details.
 - A button to list all available cookies which displays all cookies with name and value when the user clicks.

Cookie Details

Enter Cookie Name:

Enter Cookie Value:

OR

- 7.b) Write the code to develop a web application using NodeJS and ExpressJS to get details from the HTML form of the employee. Details as EmpID, Emp name, Designation, Basic Pay, and Other allowances to calculate the gross monthly salary by the following formula:

Gross Salary = Basic Pay + DA+HRA+other allowances

Dearness Allowance (DA) is 37% of Basic pay, and HRA is 8% of Basic Pay.

Generate the Salary pay slip with all the employee details along with details of calculated salary.

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

books

Id	name	author	price
1	Java for Duckies	Anonymous	199.99
2	A Dog's Guide to Web Development	Dubs	0.05
3	Design of Everyday Things	Don Norman	199.99
4	Psychology of Dogs	Caesar Milan	200.01
5	A Crash Course in Theoretical JS	Albert Einstein	10.43
6	Where Wizards Stay Up Late	Katie Hofman	19.99
			0.01

- Write ONLY the ExpressJs code to get details to create the above collection of books and insert 6 documents into the collection in MongoDB.
- Using the ExpressJS application display the following query in an ordered list with CSS
 - All book names, id, author name where the author name is Anonymous, and book title begins or ends with "d". Order the results alphabetically by book name.

- b) All book names, id, author name where the book name is on "web development" and prices between 100 to 200. Order the results in descending order of price.

OR

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

students

id	name	email	course_id	grade
123	Bart	bart@fox.com	10001	B-
456	Milhouse	milhouse@fox.com	10002	C
888	Lisa	lisa@fox.com	10001	B+
404	Ralph	ralph@fox.com	10002	A+

- i) Write ONLY the ExpressJS code to get details to create the above collection students and insert 5 documents into the collection in MongoDB.
 - ii) Using the ExpressJS application display the following query in an ordered list with CSS.
 - a) All student id, names, and course id where the grades of two high grades and student name starting or ending with "E". Order the results by ascending in student id.
 - b) All student id, names, course id where the students enrolled in course id "10002" and student grades between C and E. Order the results by student id in descending order.
9. Write the code in React to create a Class Component called 'Employee' containing name, age, gender, designation, and years of experience. Apply default values for all the properties. Write appropriate React JS script to receive the data from 'index.html', validate the age data, and display the data in the given format:
"Data of Mr./Ms. <Name> is valid" in case age is above 25 or "Data of Mr./Ms. <Name> is invalid" in case age is less than 25.
Apply necessary formatting for the display.
10. The railway department maintains a database 'testdb' in MongoDB. It has a collection trains with the fields train number, name, source, destination, distance, fare and train type (SF/Express/Passenger). Write the MongoDB shell queries for the following:
- a) Add 4 documents belonging to each train type in a single query.
 - b) List trains which has the distance more than 200 kilometers and the type is 'Express'.
 - c) The railway department wants to increase the SF train fares to 5%. So, write the update query to do this.
 - d) Remove the train(s) which has distance less than 50.
 - e) List the trains which has visit 'Bangalore'.

⇔⇔⇔ BI/D/TY ⇔⇔⇔