

Department of Computer Science & Engineering

QUESTION BANK FOR VI SEMESTER (Term: Mar-Jul 2022)

Web Technologies Laboratory (CSL68)

I.A. Marks : 50
Credits : 0:0:1
Exam Hours: 03
Exam Marks: 50

1.	(a) Write a function <code>translate()</code> that will translate a text i.e, double every consonant and place an occurrence of "o" in between. For example, <code>translate("this is fun")</code> should return the string "tothohisos isos fofunon". (b) Using Node.js Express and Mongo, implement a program to accept USN, Name, Subject_code, CIE marks and store the information in a database and display students whose CIE<20
2.	(a) Write a java script program to convert month number to month name using closures. <ul style="list-style-type: none"> • If the user enters a number less than 1 or greater than 12 or a non-number, have the function write "Bad Number" in the monthName field. • If the user enters a decimal between 1 and 12 (inclusive), strip the decimal portion of the number. (b) Write a node.js Express and Mongo program to accept Student_name,USN,semester,exam_fee from web page and delete all the students who have not paid exam fees.
3.	(a) Write a javascript to implement a Calculator using prototype, which has add, subtract and <code>getAnswer</code> functions which supports chaining, that means we should be able to do <code>new Calculator(2).add(2).add(2).subtract(3).getAnswer()</code> to get 3 as the answer. (b) Write a node.js Express and Mongo program to create 'HR' database with the collection 'employees' having the fields like emp_name,email,phone, hire_date, job_title,salary. Accept these fields information from a web page and to store it in the database and display all the employee details whose salary>50000.
4.	(a) Write an REACT program to print Name, Address and Company of an Employee. When you Click on CHANGE button, the name and address should be changed. (b) Write a Node.js program using Express framework and create an on-line training site with three pages of content: Home, Registration, Announcements & Contact. Use routing to swap between them.
5.	(a) Write a java script function named pluralize that: <ul style="list-style-type: none"> • takes 2 arguments, a noun and a number. • returns the number and pluralized form, like "5 cats" or "1 dog". • Make it handle a few collective nouns like "sheep" and "geese". (b) Write a Node.js Express and Mongo program to accept 'Student' information viz. Name, USN, Dept, Grade from a web page and store the information in a database and update Student grade with the name specified by the user and display the results.
6.	(a) Write an REACT program which accepts the Name from the form. As you type, it updates the Name in the page with an h1 tag. (b) Write a Node.js program using Express framework to display different branch information offered in an Engineering College with different background color and fonts (Note: Use Routing, Min: 3 branches)
7.	(a) Design a Student Form using HTML5 which has following fields <ul style="list-style-type: none"> a) Name : Required must be characters b) Email : Validation placeholder: please enter valid email address c) Phone : accept numbers in the following format (080-555-5555) d) Semester : For the range 1 to 8 e) Branch :Data list f) Website :Required pattern of the form-http:// (b) Create an Exam Management system using MERN Stack Framework for creating student database and display students who have secure 'S' grade. (Use Appropriate fields)

8.	<p>(a) Write an npm script having a function vowelCount() that takes a string as input and counts number of occurrences of each vowels in the string. (Hint: run the program through npm start) For. Eg. Input : vowelCount('Le Tour de France') Output: a, e, i, o, and u appear, respectively, 1, 3, 0, 1, 1 times</p> <p>(b) Using node.js Express and Mongo to implement 'FinalYears' database which accepts 'USN','Name' and 'Company_name' (by campus selection) as fields and store it in a database. Display the list of students who are selected for 'Infosys'</p>
9.	<p>(a) Write a Node.js program using Express framework and create a on-line training site with three pages of content: Home, Registration, Announcements & Contact. Use routing to swap between them.</p> <p>(b) Write a Node.js Express and Mongo program to accept User Name, Branch, Semester, from web page and display all the students who belongs to 6th Semester and CSE branch.</p>
10.	<p>(a) Write a node.js Express program to create a custom middleware functions for</p> <ol style="list-style-type: none"> Logger No. of time the visitor visited the website <p>(b) Write a Node.js Express and Mongo program to accept the fields 'ID', 'Title', 'Name', and 'branch' of a faculty and store it in the database. Display all the faculty who belongs "CSE" branch and Title is "PROFESSOR".</p>
11.	<p>(a) Create a web page with the following characteristics using BOX Model</p> <ul style="list-style-type: none"> h1's have 1px red solid borders, background color #D18C1D, and 10px of space between the content and the border (padding) List items have 15px extra space around them (margin) and background color #C0A9DB Paragraphs are contained in 600px by 400px boxes with 2px black dotted borders and background color #D1D631 <p>(b) Create an Attendance Management system using Nodejs Express and Mongo for creating student database and display student's whose attendance is below 75% (Use Appropriate fields)</p>
12.	<p>(a) Write a java script program to implement Stack and Queue using modules</p> <p>(b) Create an Exam Management system using Nodejs Express and Mongo for creating student database and display Not Eligible Student List based on the Marks<20. (Use Appropriate fields)</p>

Marks Distribution

Conduction and Result	Write Up	Execution	Viva	Change of Program	Total
Part – a	8	15 Marks	7 Marks	-10 Marks	50 Marks
Part – b		20 Marks			

1a

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title>1a</title>
    <script>
      function translate(input){
        input = input.toLowerCase();
        var newText = "";
        var chars = input.split("");
        for(i=0;i<chars.length;i++){
          if(chars[i] == 'a' || chars[i] == 'e' || chars[i] == 'i' || chars[i] == 'o' || chars[i] == 'u' || chars[i]
== ' '){
            newText += chars[i];
          }
          else{
            newText = newText+ chars[i] + 'o' + chars[i];
          }
        }
        document.write("The translated text is: "+newText);
      }
    </script>
  </head>
  <body>
    <script>
      input = prompt("Enter a sentence");
      translate(input);
    </script>
  </body>
</html>
```

1b

Index.html

```
<html>
<body>
<form action="process_get" method="GET">
Name: <input type="text" name="name"> <br>
USN: <input type="text" name="usn"><br>
Subject Code: <input type="text" name="subcode"><br>
Marks: <input type="text" name="marks"><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

Index.js

```

var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
res.sendFile(__dirname+"/"+index.html");
})
app.get('/process_get',function(req,res){
response = {
usn : req.query.usn,
name : req.query.name,
subcode : req.query.subcode,
marks : parseInt(req.query.marks)
};
console.log(response);
MongoClient.connect('mongodb://127.0.0.1:27017',function(err,client){
if(err) throw err;
var db = client.db('mydb');
var collection = db.collection('student1b');
collection.insert(response);
collection.find({ "marks" : { $lt : 20} }).toArray(function(err,results){
//console.dir(results);
if(err) throw err
res.render('disp.ejs',{students:results})
client.close();
});
});
});
});
var server = app.listen(5000,function(){
var host = server.address().address;
var port = server.address().port;
console.log("Example app listening at http://%s:%s",host,port);
});

```

Disp.ejs

Student Information

```

<% for(var i=0;i<students.length;i++){%>
<p><%= "USN: "+students[i].usn %></p>
<p><%= "Name: "+students[i].name %></p>
<p><%= "Subject Code: "+students[i].subcode %></p>
<p><%= "Marks: "+students[i].marks %></p>

<% } %>

```

2a

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <script>
      function month(){
        const months = { 1: 'January', 2: 'February', 3: 'March', 4: 'April', 5: 'May', 6: 'June', 7: 'July',
8: 'August', 9: 'September', 10: 'October', 11: 'November', 12: 'December'}
        function convert(num){
          if(num<1 || num>12 || isNaN(num))
            document.write("Bad number");
          else if(!Number.isInteger(num)){
            num = Math.floor(num);
            document.write('Month '+num+" : "+months[num]);
          }
          else{
            document.write('Month '+num+" : "+months[num]);
          }
        }
        return convert;
      }
      let a = prompt("Enter a month number (1-12) ");
      const convertFunc = month();
      convertFunc(a);
    </script>
  </body>
</html>
```

2b

Index.html

```
<html>
<body>
  <form action="process_get" method="GET">
    Name: <input type="text" name="name"> <br>
    USN: <input type="text" name="usn"><br>
    Semester: <input type="text" name="sem"><br>
    Exam Fees: <input type="text" name="examfee"><br>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
res.sendFile(__dirname+"/"+"index.html");
})
app.get('/process_get',function(req,res){
response = {
usn : req.query.usn,
name : req.query.name,
sem : req.query.sem,
examFee : req.query.examfee
};
console.log(response);
MongoClient.connect('mongodb://127.0.0.1:27017',function(err,client){
if(err) throw err;
var db = client.db('mydb');
var collection = db.collection('student2b');
collection.insert(response);
await collection.deleteMany( {"examFee" : "No"});
collection.find( {} ).toArray(function(err,results){
//console.dir(results);
if(err) throw err
res.render('disp.ejs',{students:results})
client.close();
});
});
});
});
var server = app.listen(5000,function(){
var host = server.address().address;
var port = server.address().port;
console.log("Example app listening at http://%s:%s",host,port);
});
```

Disp.ejs

Student Information

```
<% for(var i=0;i<students.length;i++){%>
<p><%= "USN: "+students[i].usn %></p>
<p><%= "Name: "+students[i].name %></p>
<p><%= "Semester: "+students[i].sem %></p>
<p><%= "Exam Fees: "+students[i].examFee %></p>

<% } %>
```

3a.

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <script>
      function Calculator(a){
        var result = a;
        this.Add = function(a){
          result = result+a;
          return this;
        }
        this.Subtract = function(a){
          result -= a;
          return this;
        }
        this.getAnswer = function(){
          return result;
        }
      }
      var t = new Calculator(2).Add(2).Add(2).Subtract(3).getAnswer();
      document.write(t);
    </script>
  </body>
</html>
```

3b

Index.html

```
<html>
<body>
  <form action="process_get" method="GET">
    Employee Name: <input type="text" name="emp_name"> <br>
    Email: <input type="text" name="email"><br>
    Phone: <input type="text" name="phone"><br>
    Hire date: <input type="text" name="hire_date"><br>
    Job Title: <input type="text" name="job_title"><br>
    Salary: <input type="text" name="salary"><br>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
res.sendFile(__dirname+"/"+"index.html");
})
app.get('/process_get',function(req,res){
response = {
emp_name : req.query.emp_name,
email : req.query.email,
phone : req.query.phone,
hire_date : req.query.hire_date,
job_title: req.query.job_title,
salary: parseInt(req.query.salary)
};
console.log(response);
MongoClient.connect('mongodb://127.0.0.1:27017',function(err,client){
if(err) throw err;
var db = client.db('mydb');
var collection = db.collection('employees3b');
collection.insert(response);
collection.find({"salary" : {$gt:50000}}).toArray(function(err,results){
//console.dir(results);
if(err) throw err
res.render('disp.ejs',{employees:results})
client.close();
});
});
});
var server = app.listen(5000,function(){
var host = server.address().address;
var port = server.address().port;
console.log("Example app listening at http://%s:%s",host,port);
});
```

Disp.ejs

Employee Information

```
<% for(var i=0;i<employees.length;i++){%>
<p><%= "Employee Name: "+employees[i].emp_name %></p>
<p><%= "Email: "+employees[i].email %></p>
<p><%= "Phone: "+employees[i].phone %></p>
<p><%= "Hire date: "+employees[i].hire_date %></p>
<p><%= "Job Title: "+employees[i].job_title %></p>
<p><%= "Salary: "+employees[i].salary %></p>

<% } %>
```


4a

REACT

4b.

Announcements.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1>Welcome to announcements page</h1>
    <a href = "/"> Back to Home page </a>

  </body>
</html>
```

Contact.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1> Welcome to contact page </h1>
    <a href = "/">Back to home page</a>
  </body>
</html>
```

Registration.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1> Welcome to registration page </h1>
    <a href = "/">Back to home page </a>
  </body>
</html>
```

Index.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1>Home page</h1>
    <a href = "registration"> Registration page</a><br>
    <a href = "announcements"> Announcements page</a><br>
    <a href = "contact"> Contact page</a><br>
  </body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
  res.sendFile(__dirname+"/"+"index.html");
})
app.get('/registration',function(req,res){
  res.sendFile(__dirname+"/"+"registration.html");
})
app.get('/announcements',function(req,res){
  res.sendFile(__dirname+"/"+"announcements.html");
})
app.get('/contact',function(req,res){
  res.sendFile(__dirname+"/"+"contact.html");
})
var server = app.listen(5000,function(){
  var host = server.address().address;
  var port = server.address().port;
  console.log("Example app listening at http://%s:%s",host,port);
});
```

5a.

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <script>
      function pluralize(noun, number) {
        if (number != 1 && noun != 'sheep' && noun != 'geese') {
          return number + ' ' + noun + 's';
        } else {
          return number + ' ' + noun;
        }
      }
    </script>
    noun = prompt("Enter the type of animal");
    num = prompt("Enter the number");
    document.write(pluralize(noun,num));
  </body>
</html>
```

5b.

Index.html

```
<html>
<body>
  <form action="process_get" method="GET">
    Name: <input type="text" name="name"> <br>
    USN: <input type="text" name="usn"><br>
    Department: <input type="text" name="dept"><br>
    Grade: <input type="text" name="grade"><br>
    <input type="submit" value="Submit">
  </form>
</body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
  res.sendFile(__dirname+"/"+ "index.html");
})
app.get('/process_get',function(req,res){
```

```

response = {
  usn : req.query.usn,
  name : req.query.name,
  dept : req.query.dept,
  grade : req.query.grade
};
console.log(response);
MongoClient.connect('mongodb://127.0.0.1:27017',function(err,client){
  if(err) throw err;
  var db = client.db('mydb');
  var collection = db.collection('student5b');
  collection.insert(response);
  collection.update( {"name" : req.query.name},{ $set: {"grade" : req.query.grade} },(err , collection)
=> {
    if(err) throw err;
    console.log("Record updated successfully");
    console.log(collection);
  });
  collection.find({}).toArray(function(err,results){
    //console.dir(results);
    if(err) throw err
    res.render('disp.ejs',{students:results})
    client.close();
  });
});
var server = app.listen(5000,function(){
  var host = server.address().address;
  var port = server.address().port;
  console.log("Example app listening at http://%s:%s",host,port);
});

```

Disp.ejs

Student Information

```

<% for(var i=0;i<students.length;i++){%>
<p><%= "USN: "+students[i].usn %></p>
<p><%= "Name: "+students[i].name %></p>
<p><%= "Department: "+students[i].dept %></p>
<p><%= "Grade: "+students[i].grade %></p>

<% } %>

```

6a.

REACT

6b.

CSE.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1>Welcome to CSE page</h1>
    <a href = "/"> Back to Home page </a>

  </body>
</html>
```

ECE.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1> Welcome to ECE page </h1>
    <a href = "/">Back to home page</a>
  </body>
</html>
```

ISE.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1> Welcome to ISE page </h1>
    <a href = "/">Back to home page </a>
  </body>
</html>
```

Index.html

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <h1>Home page</h1>
    <a href = "CSE"> CSE page</a><br>
    <a href = "ISE"> ISE page</a><br>
    <a href = "ECE"> ECE page</a><br>
  </body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
  res.sendFile(__dirname+"/"+"index.html");
})
app.get('/CSE',function(req,res){
  res.sendFile(__dirname+"/"+"CSE.html");
})
app.get('/ISE',function(req,res){
  res.sendFile(__dirname+"/"+"ISE.html");
})
app.get('/ECE',function(req,res){
  res.sendFile(__dirname+"/"+"ECE.html");
})
var server = app.listen(5000,function(){
  var host = server.address().address;
  var port = server.address().port;
  console.log("Example app listening at http://%s:%s",host,port);
});
```

7a.

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
  </head>
  <body>
    <form>
      Name <input type = "text" name = "name"><br>
      Email <input type = "email" placeholder="please enter valid email address" name =
"email"><br>
      Phone <input type = "text" pattern = "080-[0-9]{3}-[0-9]{4}" name = "phone" placeholder =
"080-555-5555"><br>
      Semester <input type = "range" name = "sem" min = "1" max = "8"><br>
      Branch <input list = "branches" name = "branch">
        <datalist id = "branches">
          <option value = "CSE">
          <option value = "ISE">
          <option value = "ECE">
          <option value = "ETE">
          <option value = "Mechanical">
          <option value = "Biotechnology">
          <option value = "EIE">
          <option value = "Medical Electronics">
          <option value = "Civil">
          <option value = "Chemical">
          <option value = "IEM">
        </datalist>
      Website <input type = "text" pattern = "/^http:\\\\/" name = "website" placeholder =
"http://abc.com"><br>
      <input type = "Submit" name = "Submit">
    </form>
  </body>
</html>
```

7b.

REACT

8a.

```
function vowel_count(str1)
{
  var str1 = str1.toLowerCase();
  var vowel_list = 'aeiou';
  var vcount_list = new Array(0,0,0,0,0);

  for(var x = 0; x < str1.length ; x++)
  {
    if ((index = vowel_list.indexOf(str1[x])) !== -1)
    {
      vcount_list[index] += 1;
    }
  }
  return vcount_list;
}
console.log(vowel_count("Le Tour de France"));
```

8b.

Index.html

```
<html>
<body>
<form action="process_get" method="GET">
Name: <input type="text" name="name"> <br>
USN: <input type="text" name="usn"><br>
Company: <input type="text" name="company"><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
res.sendFile(__dirname+"/"+"index.html");
})
app.get('/process_get',function(req,res){
response = {
usn : req.query.usn,
name : req.query.name,
company : req.query.company
```



```

};
console.log(response);
MongoClient.connect('mongodb://127.0.0.1:27017',function(err,client){
if(err) throw err;
var db = client.db('mydb');
var collection = db.collection('finalYears');
collection.insert(response);
collection.find({"company" : "Infosys"}).toArray(function(err,results){
//console.dir(results);
if(err) throw err
res.render('disp.ejs',{students:results})
client.close();
});
});
});
});
var server = app.listen(5000,function(){
var host = server.address().address;
var port = server.address().port;
console.log("Example app listening at http://%s:%s",host,port);
});

```

Disp.ejs

Student Information

```

<% for(var i=0;i<students.length;i++){%>
<p><%= "USN: "+students[i].usn %></p>
<p><%= "Name: "+students[i].name %></p>
<p><%= "Department: "+students[i].company %></p>

```

```

<% } %>

```

9a.

Same as 4b

9b.

Index.html

```
<html>
<body>
<form action="process_get" method="GET">
Name: <input type="text" name="name"> <br>
Branch: <input type="text" name="branch"><br>
Semester: <input type="text" name="sem"><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
res.sendFile(__dirname+"/"+"index.html");
})
app.get('/process_get',function(req,res){
response = {
branch : req.query.branch,
name : req.query.name,
sem : req.query.sem
};
console.log(response);
MongoClient.connect('mongodb://127.0.0.1:27017',function(err,client){
if(err) throw err;
var db = client.db('mydb');
var collection = db.collection('student9b');
collection.insert(response);
collection.find({$and:[{"sem" : "6"}, {"branch" : "CSE"}]}).toArray(function(err,results){
//console.dir(results);
if(err) throw err
res.render('disp.ejs',{students:results})
client.close();
});
});
});
var server = app.listen(5000,function(){
var host = server.address().address;
var port = server.address().port;
console.log("Example app listening at http://%s:%s",host,port);
```

```
});
```

Disp.ejs

Student Information

```
<% for(var i=0;i<students.length;i++){%>
<p><%= "Name: "+students[i].name %></p>
<p><%= "Branch: "+students[i].branch %></p>
<p><%= "Semester: "+students[i].sem %></p>
```

```
<% } %>
```

10a.

```
var express = require('express');
var app = express();
var userCount = 0;
var myLogger = function(req,res,next){
  userCount++;
  console.log("LOGGED");
  console.log("No. of times visited: "+userCount);
  next();
}
app.use(myLogger);
app.listen(3000);
```

10b.

Index.html

```
<html>
<body>
<form action="process_get" method="GET">
ID: <input type="text" name="id"> <br>
Title: <input type="text" name="title"><br>
Name: <input type="text" name="name"><br>
Branch: <input type="text" name="branch"><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
  res.sendFile(__dirname+"/"++"index.html");
})
app.get('/process_get',function(req,res){
  response = {
  id : req.query.id,
  branch : req.query.branch,
  name : req.query.name,
  title : req.query.title
  };
  console.log(response);
  MongoClient.connect('mongodb://127.0.0.1:27017',function(err,client){
    if(err) throw err;
    var db = client.db('mydb');
    var collection = db.collection('student10b');
```

```

collection.insert(response);
collection.find({$and:[{"title" : "Professor"}, {"branch" :
"CSE"}]}).toArray(function(err,results){
//console.dir(results);
if(err) throw err
res.render('disp.ejs',{students:results})
client.close();
});
});
});
var server = app.listen(5000,function(){
var host = server.address().address;
var port = server.address().port;
console.log("Example app listening at http://%s:%s",host,port);
});

```

Disp.ejs

Faculty Information

```

<% for(var i=0;i<students.length;i++){%>
<p><%= "ID: "+students[i].id %></p>
<p><%= "Title: "+students[i].title %></p>
<p><%= "Name: "+students[i].name %></p>
<p><%= "Branch: "+students[i].branch %></p>

<% } %>

```

11a.

```
<!DOCTYPE html>
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <title></title>
    <style>
      h1 {
        border : 1px solid red;
        background-color : #D18C1D;
        padding : 10px;
      }
      ul {
        margin : 15px;
        background-color : #C0A9DB;
      }
      p {
        height : 600px;
        width : 400px;
        border : 2px dotted black;
        background-color: #D1D631;
      }
    </style>
  </head>
  <body>
    <h1>This is a H1 heading</h1>
    Branches offered
    <ul>
      <li>CSE
      <li> ISE
      <li> ECE
    </ul>
    <p> This is a paragraph element </p>
  </body>
</html>
```

11b.

Index.html

```
<html>
<body>
<form action="process_get" method="GET">
Name: <input type="text" name="name"><br>
Attendance: <input type="text" name="att"><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
res.sendFile(__dirname+"/"+"index.html");
})
app.get('/process_get',function(req,res){
response = {
name : req.query.name,
att : parseInt(req.query.att)
};
console.log(response);
MongoClient.connect('mongodb://127.0.0.1:27017',function(err,client){
if(err) throw err;
var db = client.db('mydb');
var collection = db.collection('student11b');
collection.insert(response);
collection.find({ "att" : {$lt : 75} }).toArray(function(err,results){
//console.dir(results);
if(err) throw err
res.render('disp.ejs',{students:results})
client.close();
});
});
});
var server = app.listen(5000,function(){
var host = server.address().address;
var port = server.address().port;
console.log("Example app listening at http://%s:%s",host,port);
});
```

Disp.ejs

Student Information

```
<% for(var i=0;i<students.length;i++){%>
<p><%= "Name: "+students[i].name %></p>
<p><%= "Attendance: "+students[i].att %></p>
```

```
<% } %>
```

12a.

Stack.js

```
function push(arr,num){
  arr.push(num);
  return arr;
}
function pop(arr){
  arr.pop();
  return arr;
}
module.exports = {
  push : push,
  pop : pop
}
```

Queue.js

```
function enqueue(arr,num){
  arr.push(num);
  return arr;
}
function dequeue(arr){
  arr.shift();
  return arr;
}
module.exports = {
  enqueue:enqueue,
  dequeue:dequeue
}
```

Main.js

```
var stack = require('./stack');
var queue = require('./queue');
var st = [];
var qu = [];
st = stack.push(st,10);
st = stack.push(st,20);
st = stack.push(st,30);
st = stack.pop(st);
qu = queue.enqueue(qu,12);
qu = queue.enqueue(qu,13);
qu = queue.enqueue(qu,14);
qu = queue.dequeue(qu);
console.log("The operations performed are : stack.push(10), stack.push(20), stack.push(30) and stack.pop()");
console.log("The operations performed on queue are : queue.enqueue(12),queue.enqueue(13),queue.enqueue(14) and queue.dequeue.");
```



```
console.log(st);
console.log(qu);
```

12b.

Index.html

```
<html>
<body>
<form action="process_get" method="GET">
Name: <input type="text" name="name"> <br>
USN: <input type="text" name="usn"><br>
Marks: <input type="text" name="marks"><br>
<input type="submit" value="Submit">
</form>
</body>
</html>
```

Index.js

```
var express = require('express');
var app = express();
var MongoClient = require('mongodb').MongoClient;
app.get('/',function(req,res){
res.sendFile(__dirname+"/"+"index.html");
})
app.get('/process_get',function(req,res){
response = {
usn : req.query.usn,
name : req.query.name,
marks : parseInt(req.query.marks)
};
console.log(response);
MongoClient.connect('mongodb://127.0.0.1:27017',function(err,client){
if(err) throw err;
var db = client.db('mydb');
var collection = db.collection('student12b');
collection.insert(response);
collection.find({"marks" : {$lt : 20}}).toArray(function(err,results){
//console.dir(results);
if(err) throw err
res.render('disp.ejs',{students:results})
client.close();
});
});
});
var server = app.listen(5000,function(){
var host = server.address().address;
var port = server.address().port;
console.log("Example app listening at http://%s:%s",host,port);
```

```
});
```

Disp.ejs

Not Eligible Student List

```
<% for(var i=0;i<students.length;i++){%>
<p><%= "USN: "+students[i].usn %></p>
<p><%= "Name: "+students[i].name %></p>
<p><%= "Marks: "+students[i].marks %></p>

<% } %>
```