

NAME:YOGESHWARAN

PROJECT TITLE : Library Management System

Reg No:731621104063

Library Management System

Form creation:

```
<html>
<head>
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css" />
</head>

<body>
  <h1>Library Management System</h1>
  <div class="container">
    <div class="row justify-content-center">
      <div class="col-md-10">
        <form #myForm="ngForm">
          <div class="form-group">
            <label for="Student Id">
              Student Id
            </label>
            <input type="text"
              class="form-control" name="stuId" #stuId="ngModel" ngModel/>
            <div *ngIf="stuId.touched">
              <div *ngIf="stuId.errors?.['required']">
                <b style="color: red;">Student Id Required</b>
              </div>
              <div *ngIf="stuId.errors?.['pattern']">
                <font color="red"><b>Invalid Student Id</b></font>
              </div>
            </div>
          </div>
        </form>
      </div>
    </div>
  </div>
</body>
</html>
```

Library Management System

Student Id

Student Name

Book Id

Book Name

Student Contact

Insert

Update

Delete

Find

Find All

DATABASE:

```
[
  {
    "id": "111",
    "stuName": "yogi",
    "bookId": "9251",
    "bookname": "Gentelman",
    "stuContact": "92586875",
  },
  {
    "id": "113",
    "stuName": "Yuva",
    "bookId": 9253,
    "bookname": "Kadhal",
    "stuContact": 994433552,
  }
]
```

MODEL:

```
TS Library.ts  X  TS app.component.ts  TS employee
src > app > model > TS Library.ts > Library
1  export class Library {
2      id : number = 0;
3      stuName : string = "";
4      bookId : number = 0;
5      bookname :string = "";
6      stuContact :number = 0;
7  }
```

INSERT OPERATION:

```
<button class="btn btn-warning" (click)="findEmployee(myForm.value)">
|   Find
</button>
<br><br>
<button class="btn btn-info" (click)="findAllEmployee()">
|   Find All
</button>
```

Library Management System

Student Id

Student Name

Book Id

Book Name

Student Contact

Employee Details Added

```
[
  {
    "id": "111",
    "stuName": "yogi",
    "bookId": "9251",
    "bookname": "Gentelman",
    "stuContact": "9258687521"
  },
  {
    "id": "113",
    "stuName": "Yuva",
    "bookId": 9253,
    "bookname": "Kadhal",
    "stuContact": 9944335522
  },
  {
    "id": "213",
    "stuName": "vijay",
    "bookId": 83459,
    "bookname": "wonderfull",
    "stuContact": 2348238947
  }
]
```

UPDATE OPERATION:

Library Management System

Student Id

Student Name

Book Id

Book Name

Student Contact

Insert

Update

Delete

Find

Find All

Employee Details Updated

```
[
  {
    "id": "111",
    "stuName": "yogi",
    "bookId": "9251",
    "bookname": "Gentelman",
    "stuContact": "9258687521"
  },
  {
    "id": "113",
    "stuName": "Yuva",
    "bookId": 9253,
    "bookname": "Kadhal",
    "stuContact": 9944335522
  },
  {
    "id": "213",
    "stuName": "yuvan",
    "bookId": 83459,
    "bookname": "wonderfull",
    "stuContact": 2348238947
  }
]
```

DELETE OPERATION:

Deleting user with id 0

Library Management System

Student Id

Student Name

Book Id

Book Name

Student Contact

Employee Details Deleted

```
[
  {
    "id": "111",
    "stuName": "yogi",
    "bookId": "9251",
    "bookname": "Gentelman",
    "stuContact": "9258687521"
  },
  {
    "id": "113",
    "stuName": "Yuva",
    "bookId": 9253,
    "bookname": "Kadhal",
    "stuContact": 9944335522
  }
]
```

APP.COMPONENT TS:

```

export class AppComponent {
  customer: Customer;
  result: string;
  flag: boolean;
  customerArr: Customer[]

  constructor(private service: CustomerService) {
    this.customer = new Customer();
    this.result = '';
    this.flag = false;
    this.customerArr = []
  }

  insertCustomer(data: any) {
    this.customer.CustomerName = data.customername;
    this.customer.Roomno = data.roomnumber;
    this.customer.AadharNumber = data.aadharnumber;
    this.customer.id = data.cusId;

    this.result = this.service.insertCustomer(this.customer);
  }

  updatecustomer(data: any) {

    this.customer.CustomerName = data.customername;
    this.customer.Roomno = data.roomnumber;
    this.customer.AadharNumber = data.aadharnumber;
    this.customer.id = data.cusId;

    this.result = this.service.updatecustomer(this.customer);

  }

  deletecustomer(data: any) {
    this.result = this.service.deletecustomer(data.cusId);
  }

  findcustomer(data: any) {
    this.customer = this.service.findcustomer(data.cusId);
    this.result = this.customer.CustomerName + " " + this.customer.Roomno + " " + this.customer.AadharNumber;
  }

  findAllcustomer() {
    this.customerArr = this.service.findAllcustomer();
    this.flag = true;
  }
}

```

SERVICE.TS:


```
updatecustomer(customer:Customer){
    this.http.put(this.url+"/"+customer.id,customer).subscribe();
    return "Customer Details Updated";
}
deletecustomer(cusId:number){
    this.http.delete(this.url+"/"+cusId).subscribe();
    return "Customer Details Deleted";
}

findcustomer(cusId:number){
    this.http.get<Customer>(this.url+"/"+cusId).subscribe(data =>this.customer=data);
    return this.customer;
}

findAllcustomer(){
    this.http.get<Customer[]>(this.url).subscribe(data =>this.customerArr=data);
    return this.customerArr;
}
}
```