

angular.json

```
1 {
2   "$schema": "./node_modules/@angular/cli/lib/config/schema.json",
3   "version": 1,
4   "newProjectRoot": "projects",
5   "projects": {
6     "ProjectYoga": {
7       "projectType": "application",
8       "schematics": {
9         "@schematics/angular:application": {
10           "strict": true
11         }
12       },
13       "root": "",
14       "sourceRoot": "src",
15       "prefix": "app",
16       "architect": {
17         "build": {
18           "builder": "@angular-devkit/build-angular:browser",
19           "options": {
20             "outputPath": "dist/ProjectYoga",
21             "index": "src/index.html",
22             "main": "src/main.ts",
23             "polyfills": "src/polyfills.ts",
24             "tsConfig": "tsconfig.app.json",
25             "assets": [
26               "src/favicon.ico",
27               "src/assets"
28             ],
29             "styles": ["node_modules/bootstrap/dist/css/bootstrap.css",
30               "src/styles.css"
31             ],
32             "scripts": ["node_modules/bootstrap/dist/js/bootstrap.js"]
33           },
34           "configurations": {
35             "production": {
36               "budgets": [
37                 {
38                   "type": "initial",
39                   "maximumWarning": "5mb",
40                   "maximumError": "5mb"
41                 },
42                 {
43                   "type": "anyComponentStyle",
44                   "maximumWarning": "2kb",
45                   "maximumError": "4kb"
46                 }
47               ],
48               "fileReplacements": [
49                 {
50                   "replace": "src/environments/environment.ts",
51                   "with": "src/environments/environment.prod.ts"
52                 }
53               ],
54               "outputHashing": "all"
55             },
56             "development": {
57               "buildOptimizer": false,
58               "optimization": false,
```

```

59         "vendorChunk": true,
60         "extractLicenses": false,
61         "sourceMap": true,
62         "namedChunks": true
63     },
64 },
65     "defaultConfiguration": "production"
66 },
67     "serve": {
68         "builder": "@angular-devkit/build-angular:dev-server",
69         "configurations": {
70             "production": {
71                 "browserTarget": "ProjectYoga:build:production"
72             },
73             "development": {
74                 "browserTarget": "ProjectYoga:build:development"
75             }
76         },
77         "defaultConfiguration": "development"
78     },
79     "extract-il8n": {
80         "builder": "@angular-devkit/build-angular:extract-il8n",
81         "options": {
82             "browserTarget": "ProjectYoga:build"
83         }
84     },
85     "test": {
86         "builder": "@angular-devkit/build-angular:karma",
87         "options": {
88             "main": "src/test.ts",
89             "polyfills": "src/polyfills.ts",
90             "tsConfig": "tsconfig.spec.json",
91             "karmaConfig": "karma.conf.js",
92             "assets": [
93                 "src/favicon.ico",
94                 "src/assets"
95             ],
96             "styles": [
97                 "src/styles.css"
98             ],
99             "scripts": []
100         }
101     }
102 },
103 },
104 },
105 "defaultProject": "ProjectYoga"
106 }

```

backend\app.js

```
1 const express = require("express");
2 const cors = require("cors");
3 const bodyparser = require("body-parser");
4 const mongodb = require("mongodb").MongoClient;
5 const app = express();
6 app.use(cors());
7 app.use(bodyparser.json());
8 var db;
9 mongodb.connect("mongodb://localhost:27017/yogausers?
retryWrites=true&w=majority", (error, result) => {
10     if (error) {
11         console.log("DB Not Connected");
12     }
13     else {
14         db = result.db("yogausers");
15         console.log("DB Connected");
16     }
17 });
18 app.use((req, res, next) => {                // middleware common for all
the paths
19     console.log("Middleware 1");
20     next();
21 });
22 app.use("/home", (req, res, next) => {        // middleware only for
the /home path
23     console.log("Middleware 2");
24     next();
25 });
26 function verifyUser(req, res, next) {
27     console.log("User Verified");
28     next();
29 }
30 app.get("/", (req, res) => {
31     console.log("Index Page");
32     res.send("<h1>Welcome to Express</h1>");
33 });
34 app.get("/home", verifyUser, (req, res) => {
35     console.log("Home Page");
36     var data = { a: "hi", b: "hello" };
37     res.json(data);
38 });
39 app.post("/register", (req, res) => {
40     req.body._id = new Date().getTime();
41     console.log(req.body);
42     db.collection("users").insertOne(req.body, (error, data) => {
43         if (error) {
44             res.status(403).json("Error in Inserting Doc");
45         }
46         else {
47             res.json("User Registered Successfully!");
48         }
49     })
50 });
51 app.post("/login", (req, res) => {
52     console.log(req.body);
53     db.collection("users").find(req.body, { projection: { _id: 1, Fname:
1, lname: 1, userage: 1, gender: 1 } }).toArray((error, data) => {
54         if (error) {
```

```

55         res.status(403).json("Error");
56     }
57     else {
58         res.json(data);
59     }
60 });
61 });
62 app.get("/allusers", (req, res) => {
63     db.collection("users").find(null, { projection: { userpass:
0 } }).toArray((error, data) => {
64         if (error) {
65             res.status(403).json("Error in Finding the Doc");
66         }
67         else {
68             res.json(data);
69         }
70     });
71 });
72 app.get("/uemailcheck/:uemail", (req, res) => {
73     console.log(req.params.uemail);
74     db.collection("users").find({ uemail: req.params.uemail },
{ projection: { _id: 1 } }).toArray((error, data) => {
75         if (error) {
76             res.status(403).json("Error in Finding the Doc");
77         }
78         else {
79             res.json(data);
80         }
81     });
82 });
83 app.get("/uemailcheck/:uemail", (req, res) => {
84     console.log(req.params.uemail);
85     db.collection("users").find({ uemail: req.params.uemail },
{ projection: { _id: 1 } }).toArray((error, data) => {
86         if (error) {
87             res.status(403).json("Error in Finding the Doc");
88         }
89         else {
90             res.json(data);
91         }
92     });
93 });
94 app.get("/getuser/:userid", (req, res) => {
95     console.log(req.params.userid);
96     db.collection("users").find({ _id:
Number(req.params.userid) }).toArray((error, data) => {
97         if (error) {
98             res.status(403).json("Error in Finding the Doc");
99         }
100         else {
101             res.json(data);
102         }
103     });
104 });
105 app.get("/profile/:userid", (req, res) => {
106     console.log(req.params.userid);
107     db.collection("users").find({ _id: Number(req.params.userid) },
{ projection: { userpass: 0 } }).toArray((error, data) => {
108         if (error) {
109             res.status(403).json("Error in Finding the Doc");
110         }

```

```

111         else {
112             res.json(data);
113         }
114     });
115 });
116 app.put("/cart", (req, res) => {
117     console.log(req.body);
118     var condition = { _id: req.body._id };
119     var newValues = { $set: { date:req.body.date, pack:req.body.pack,
level: req.body.level, isdietplan: req.body.isdietplan, istrain:
req.body.istrain, payment: req.body.payment, isbath: req.body.isbath ,
paid:req.body.paid, coupon:req.body.coupon } }
120     db.collection("users").update(condition, newValues, (error, data) => {
121         if (error) {
122             res.status(403).json("Error in Finding the Doc");
123         }
124         else {
125             res.json("User data updated successfully");
126         }
127     })
128 });
129 app.put("/updateuser", (req, res) => {
130     console.log(req.body);
131     var condition = { _id: req.body._id };
132     var newValues = { $set: { Fname: req.body.Fname, lname:
req.body.lname, usage: req.body.usage, uemail: req.body.uemail,
level:req.body.level, gender:req.body.gender,
pack:req.body.pack ,isbath:req.body.isbath, isdietplan:req.body.isdietplan,
istrain:req.body.istrain, date:req.body.date} }
133     db.collection("users").update(condition, newValues, (error, data) => {
134         if (error) {
135             res.status(403).json("Error in Finding the Doc");
136         }
137         else {
138             res.json("User data updated successfully");
139         }
140     })
141 });
142 app.delete("/deleteuser/:userid", (req, res) => {
143     console.log(req.params);
144     db.collection("users").deleteOne({ _id: Number(req.params.userid) },
(error, data) => {
145         res.json("Users deleted successfully");
146     });
147 });
148 app.get("/searchuser/:searchtxt?", (req, res) => {
149     console.log(req.params.searchtxt);
150     if (req.params.searchtxt != undefined) {
151         var search = new RegExp(req.params.searchtxt, "i");
152         var searchCond = { Fname: search };
153     }
154     else {
155         var searchCond = null;
156     }
157     db.collection("users").find(searchCond).toArray((error, data) => {
158         res.json(data);
159     });
160 });
161 //app.listen(3000, () => console.log("Server ready"))
162 module.exports = app;
163 // const express = require("express")

```

```
164 // const mongo = require("mongodb").MongoClient
165 // const app = express()
166 // app.post("/trip", (req, res) => {
167 //   /* */
168 //   })
169 //   app.get("/trips", (req, res) => {
170 //     res.status(200).json({ trips: {} })
171 //   })
172 //   app.post("/expense", (req, res) => {
173 //     /* */
174 //   })
175 //   app.get("/expenses", (req, res) => {
176 //     /* */
177 //   })
178 //   app.listen(3000, () => console.log("Server ready"))
```

```

1 // Karma configuration file, see link for more information
2 // https://karma-runner.github.io/1.0/config/configuration-file.html
3 module.exports = function (config) {
4   config.set({
5     basePath: '',
6     frameworks: ['jasmine', '@angular-devkit/build-angular'],
7     plugins: [
8       require('karma-jasmine'),
9       require('karma-chrome-launcher'),
10      require('karma-jasmine-html-reporter'),
11      require('karma-coverage'),
12      require('@angular-devkit/build-angular/plugins/karma')
13    ],
14    client: {
15      jasmine: {
16        // you can add configuration options for Jasmine here
17        // the possible options are listed at https://jasmine.github.io/
18        // for example, you can disable the random execution with `random:
19        // or set a specific seed with `seed: 4321`
20      },
21      clearContext: false // leave Jasmine Spec Runner output visible in
22      browser
23    },
24    jasmineHtmlReporter: {
25      suppressAll: true // removes the duplicated traces
26    },
27    coverageReporter: {
28      dir: require('path').join(__dirname, './coverage/ProjectYoga'),
29      subdir: '.',
30      reporters: [
31        { type: 'html' },
32        { type: 'text-summary' }
33      ]
34    },
35    reporters: ['progress', 'kjhtml'],
36    port: 9876,
37    colors: true,
38    logLevel: config.LOG_INFO,
39    autoWatch: true,
40    browsers: ['Chrome'],
41    singleRun: false,
42    restartOnFileChange: true
43  });
44 };

```

package.json

```
1 {
2   "name": "project-yoga",
3   "version": "0.0.0",
4   "scripts": {
5     "ng": "ng",
6     "start": "ng serve",
7     "build": "ng build",
8     "watch": "ng build --watch --configuration development",
9     "test": "ng test"
10  },
11  "private": true,
12  "dependencies": {
13    "@angular/animations": "~12.2.0",
14    "@angular/common": "~12.2.0",
15    "@angular/compiler": "~12.2.0",
16    "@angular/core": "~12.2.0",
17    "@angular/forms": "~12.2.0",
18    "@angular/platform-browser": "~12.2.0",
19    "@angular/platform-browser-dynamic": "~12.2.0",
20    "@angular/router": "~12.2.0",
21    "body-parser": "^1.19.0",
22    "bootstrap": "^5.1.3",
23    "cors": "^2.8.5",
24    "express": "^4.18.3",
25    "git": "^0.1.5",
26    "mangodb": "^1.0.0",
27    "mongodb": "^3.7.4",
28    "ngx-webstorage-service": "^4.1.0",
29    "nodemon": "^2.0.14",
30    "rxjs": "~6.6.0",
31    "tslib": "^2.3.0",
32    "zone.js": "~0.11.4"
33  },
34  "devDependencies": {
35    "@angular-devkit/build-angular": "~12.2.9",
36    "@angular/cli": "~12.2.9",
37    "@angular/compiler-cli": "~12.2.0",
38    "@types/jasmine": "~3.8.0",
39    "@types/node": "^12.11.1",
40    "jasmine-core": "~3.8.0",
41    "karma": "~6.3.0",
42    "karma-chrome-launcher": "~3.1.0",
43    "karma-coverage": "~2.0.3",
44    "karma-jasmine": "~4.0.0",
45    "karma-jasmine-html-reporter": "~1.7.0",
46    "typescript": "~4.3.5"
47  }
48 }
```


README.md

```
1 # YOGA FITNESS CENTER WEB APPLICATION
2 ##### 1. Single Page Application (Angular Routing) and Mobile Responsive
(Bootstrap).
3 ##### 2. Tech Stack: MEAN Stack (MongoDB, Express.js, Angular, Node.js).
4 ##### 3. The user has to register and login to buy packages , Admin user
can do member management.
5 ##### YogaFitnessCenter
6 This project is generated with [Angular CLI](https://github.com/angular/
angular-cli) version 12.2.9.
7 ##### Development server
8 Run `ng serve` for a dev server. Navigate to `http://localhost:4200/`. The
app will automatically reload if you change any of the source files.
9 ##### Code scaffolding
10 Run `ng generate component component-name` to generate a new component.
You can also use `ng generate directive|pipe|service|class|guard|interface|
enum|module`.
11 ##### Build
12 Run `ng build` to build the project. The build artifacts will be stored in
the `dist/` directory.
13 ##### Running unit tests
14 Run `ng test` to execute the unit tests via [Karma](https://karma-
runner.github.io).
15 ##### Running end-to-end tests
16 Run `ng e2e` to execute the end-to-end tests via a platform of your
choice. To use this command, you need to first add a package that implements
end-to-end testing capabilities.
17 ##### Further help
18 To get more help on the Angular CLI use `ng help` or go check out the
[Angular CLI Overview and Command Reference](https://angular.io/cli) page.
19 # TO RUN THE YOGA FITNESS CENTER APPLICATION
20 ## 1. Prerequisites
21 Before we begin, ensure that you have the following tools and knowledge:
22 ##### 1. Node.js:
23 Install Node.js from the official website (https://nodejs.org/). Node.js
will enable you to run JavaScript on the server-side.
24 ##### 2. npm (Node Package Manager):
25 npm comes bundled with Node.js and is used to manage packages and
dependencies for your project.
26 ##### 3. MongoDB:
27 Install MongoDB from the official website (https://www.mongodb.com/).
MongoDB is a NoSQL database that stores your application's data.
28 ##### 4. Angular CLI:
29 Install Angular CLI globally using npm. You can do this by running `npm
install -g @angular/cli`. Angular is a front-end framework for building
dynamic web applications.
30 Now that you have the necessary tools in place, let's get started with
building your MEAN stack application!
31 ## 2. Install node modules
32 > npm install
33 ## 3. Run Application
34 1. open command prompt and run below command
35 > npm run start
36 2. open another command prompt and run below command
37 > node server.js
```

server.js

```
1 const http=require("http");
2 const app =require("./backend/app");
3 const server =http.createServer(app);
4 server.listen(3000, ()=>{
5     console.log("check 3000");
6 });
```

src\app\adduser\adduser.component.html

```
1 <div class="container" style="min-height: 560px; font-family: 'Big
Shoulders Stencil Display', cursive;
2 font-family: 'Londrina Solid', cursive;">
3   <div class="row justify-content-center" style="padding-top: 4%;">
4     <div class="col-lg-5">
5       <div class="card shadow-lg border-0 rounded-lg mt-5">
6         <div class="card-header " style="background-color: #c5c6c7;">
7           <h2 style="text-align: center; font-family: 'Big Shoulders
Stencil Display', cursive;
8                                     font-family: 'Londrina
Solid', cursive;">Add members</h2>
9           <h5>{{msg}}</h5>
10          <div class="card-body shadow-lg" style="background-color:
#1F2833;">
11            <form (submit)="addUser(f)" #f="ngForm">
12              <div class="row">
13                <div class="col">
14                  <div class="form-floating mb-3">
15                    <input class="form-control" id="inputfname"
#firstname="ngModel" name="Fname" ngModel type="text"
16                      placeholder="fname" required minlength="3" />
17                    <label for="fname">First Name</label>
18                  </div>
19                </div>
20                <div class="col">
21                  <div class="form-floating mb-3">
22                    <input class="form-control" id="inputlname"
#lastname="ngModel" name="lname" type="text"
23                      placeholder="lname" ngModel required />
24                    <label for="lname">Last Name</label>
25                  </div>
26                </div>
27              </div>
28              <p class="alert alert-danger"
*ngIf="firstname.errors?.required && firstname.touched">Please enter the
29                First Name</p>
30              <p class="alert alert-danger"
*ngIf="firstname.errors?.minlength && firstname.touched">Please enter Min 3
31                characters </p>
32              <div class="row">
33                <div class="col">
34                  <div class="form-floating mb-3">
35                    <input class="form-control" id="age" type="number"
placeholder="age" name="userage" ngModel
36                      #userage="ngModel" required />
37                    <label for="age">Age</label>
38                  </div>
39                  <p class="alert alert-danger"
*ngIf="userage.errors?.required && userage.touched">Please enter the age
40                    </p>
41                </div>
42                <div class="col">
43                  <div class="form-check">
44                    <input class="form-check-input" type="radio"
name="gender" value="male" id="flexRadioDefault1"
45                      ngModel #usergender="ngModel" required>
46                    <label class="form-check-label"
for="flexRadioDefault1">
```

```

47             style="background-color: whitesmoke; border-radius:
3px; width: 50%; text-align: center; ">
48             Male
49         </label>
50     </div>
51     <div class="form-check" style="margin-top: 10px; ">
52         <input class="form-check-input" type="radio"
name="gender" value="female" id="flexRadioDefault2"
53             ngModel #usergender="ngModel" required>
54         <label class="form-check-label"
for="flexRadioDefault2"
55             style="background-color: whitesmoke; border-radius:
3px; width: 50%; text-align: center; ">
56             Female
57         </label>
58     </div>
59 </div>
60     <p class="alert alert-danger"
*ngIf="usergender.errors?.required && usergender.touched">Please click the
61         gender </p>
62 </div>
63     <div class="form-floating mb-3">
64         <input class="form-control" id="inputphone" type="text"
placeholder="Phone Number" name="userphone"
65             ngModel #userphone="ngModel" required pattern="^(0|[1-9]
[0-9]*)$" maxlength="10" />
66         <label for="inputphone">Phone Number</label>
67     </div>
68     <p class="alert alert-danger"
*ngIf="userphone.errors?.required && userphone.touched">Please enter the
69         Phone number</p>
70     <p class="alert alert-danger"
*ngIf="userphone.errors?.pattern">please enter the valid phone number</p>
71     <div class="form-floating mb-3">
72         <input class="form-control"
id="inputEmail" (blur)="uemailcheck(uemail.value)" name="uemail"
73             type="email" ngModel #uemail="ngModel"
placeholder="name@example.com" required email />
74         <label for="inputEmail">Email address</label>
75     </div>
76     <p class="alert alert-danger"
*ngIf="uemail.errors?.required && uemail.touched">Please enter the Email</p>
77     <p class="alert alert-danger" *ngIf="uemail.errors?.email
&& uemail.touched">Please enter the valid Email
78 </p>
79     <div class="form-floating mb-3">
80         <input class="form-control" id="inputPassword"
type="password" placeholder="Password" name="userpass"
81             ngModel #userpass="ngModel" required password
minlength="8" />
82         <label for="inputPassword">Create Password</label>
83     </div>
84     <p class="alert alert-danger"
*ngIf="userpass.errors?.required && userpass.touched">Please Create a
85         password</p>
86     <p class="alert alert-danger"
*ngIf="userpass.errors?.minlength">Please Create a strong password (min 8
87         character) </p>
88     <h4 style="color:; margin: 1vw; display: flex; justify-
content: center; ">Package</h4>
89     <select class="form-select form-select-lg mb-3" name="pack"

```

```

ngModel aria-label=".form-select-lg example"
90         placeholder="Select Package">
91         <option selected>Select Package</option>
92         <option value="Hatha Yoga">Hatha Yoga</option>
93         <option value="Vinyasa Yoga">Vinyasa Yoga</option>
94         <option value="Iyengar Yoga">Iyengar Yoga</option>
95         <option value="Ashtanga Yoga">Ashtanga Yoga</option>
96         <option value="Bikram Yoga">Bikram Yoga</option>
97         <option value="Kundalini Yoga">Kundalini Yoga</option>
98     </select>
99     <h4 style="color: pink; margin: 1vw; display: flex; justify-
content: center;">Level</h4>
100     <div class="form-check" style="display: inline-block; color:
#c5c6c7; margin-left: 2vw; ">
101         <input class="form-check-input" type="radio" name="level"
#level="ngModel" id="gridRadios1"
102             value="Beginner" checked ngModel >
103         <label class="form-check-label" for="level">
104             Beginner
105         </label>
106     </div>
107     <div class="form-check" style="display: inline-block; color:
#c5c6c7; margin-left: 2vw; ">
108         <input class="form-check-input" type="radio" name="level"
#level="ngModel" id="gridRadios2"
109             value="Intermediate" ngModel>
110         <label class="form-check-label" for="level">
111             Intermediate
112         </label>
113     </div>
114     <div class="form-check" style="display: inline-block; color:
#c5c6c7; margin-left: 2vw; ">
115         <input class="form-check-input" type="radio" name="level"
#level="ngModel" id="gridRadios3"
116             value="Advanced" ngModel>
117         <label class="form-check-label" for="level">
118             Advanced
119         </label>
120     </div>
121     <div>
122         <h4 style="color: pink; margin: 1vw; display: flex;
justify-content: center;">Adds on</h4>
123         <div class="form-check" style="display: inline-block;
color: #c5c6c7; margin-left: 2vw; ">
124             <input class="form-check-input" type="checkbox"
id="flexCheckDefault" ngModel value="Steambath"
125                 name="isbath">
126             <label class="form-check-label" for="flexCheckDefault">
127                 Steambath
128             </label>
129         </div>
130         <div class="form-check" style="display: inline-block;
color: #c5c6c7; margin-left: 2vw; ">
131             <input class="form-check-input" type="checkbox" ngModel
value="DietPlan" name="isdietplan"
132                 id="flexCheckDefault">
133             <label class="form-check-label" for="flexCheckDefault">
134                 DietPlan
135             </label>
136         </div>
137         <div class="form-check" style="display: inline-block;

```

```

color: #c5c6c7; margin-left: 2vw; ">
138         <input class="form-check-input" type="checkbox"
id="flexCheckDefault" ngModel value="PersonalTrainer"
139         name="istrain">
140         <label class="form-check-label" for="flexCheckDefault">
141             PersonalTrainer
142         </label>
143     </div>
144 </div>
145 <div class="d-flex align-items-center justify-content-
between mt-4 mb-0">
146     <button class="btn btn-info btn-block" value="Register"
type="submit"
147         [disabled]="f.invalid || !useremailavail" style="text-
align: center; font-family: 'Big Shoulders Stencil Display', cursive;
148         font-family:
'Londrina Solid', cursive;background-color: pink!important; border-color:
pink !important;"> Register</button> <h5 style="color: pink;">{{msg}}</h5>
149     </div>
150 </form>
151 </div>
152 <div class="card-footer text-center py-3">
153     <p class="card-text"><small class="text-muted">Yoga fitness
center</small></p>
154     <button class="btn btn-info me-md-2" type="button"
routerLink="/manager" style="border:solid #1F2833; background-color:
pink;margin: 1vw;">Back
155     </button>
156 </div>
157 </div>
158 </div>
159 </div>
160 </div>
161 </div>

```

src/app/adduser/adduser.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { AdduserComponent } from './adduser.component';
3 describe('AdduserComponent', () => {
4   let component: AdduserComponent;
5   let fixture: ComponentFixture<AdduserComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ AdduserComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(AdduserComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src/app/adduser/adduser.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { NgForm } from '@angular/forms';
3 import { UsersService } from '../users.service';
4 @Component({
5   selector: 'app-adduser',
6   templateUrl: './adduser.component.html'
7 })
8 export class AdduserComponent implements OnInit {
9   msg:string;
10  useremailavail=false;
11  date = new Date();
12  constructor(public userSer :UsersService) { }
13  ngOnInit(): void {
14  }
15  addUser(form:NgForm){
16    form.value.date = this.date;
17    this.userSer.doUserRegistration(form.value).subscribe((data:string)=>{
18      console.log(data);
19      this.msg = data;
20      form.reset();
21    }, (error:any)=>{
22      console.log(error);
23      this.msg = "Something Went Wrong!!";
24    });
25  }
26  uemailcheck(uemail:string)
27  {
28    this.userSer.uemailcheckAvail(uemail).subscribe((data:any[])=>{
29      console.log(data);
30      if(data.length==0)
31      {
32        this.msg=""
33        this.useremailavail= true;
34      }
35      else{
36        this.msg="Email already registered";
37        this.useremailavail= false;
38      }
39    },(error:any)=>{
40      console.log(error);
41      this.msg="Something went wrong";
42    })
43  }
44 }
```


src\app\admin.guard.spec.ts

```
1 import { TestBed } from '@angular/core/testing';
2 import { AdminGuard } from './admin.guard';
3 describe('AdminGuard', () => {
4   let guard: AdminGuard;
5   beforeEach(() => {
6     TestBed.configureTestingModule({});
7     guard = TestBed.inject(AdminGuard);
8   });
9   it('should be created', () => {
10     expect(guard).toBeTruthy();
11   });
12 });
```

src\app\admin.guard.ts

```
1 import { Injectable } from '@angular/core';
2 import { CanActivate, Router } from '@angular/router';
3 import { UsersService } from '../users.service';
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class AdminGuard implements CanActivate {
8   useri:string=localStorage.getItem("loggeduser");
9   constructor(public userSer : UsersService, public myRouter : Router)
10  {
11  }
12  canActivate(): boolean {
13    if(!this.userSer.isAdmin())
14    {
15      this.myRouter.navigateByUrl("/");
16    }
17    return this.userSer.isAdmin();
18  }
19 }
```

src/app/app-routing.module.ts

```
1 import { NgModule } from '@angular/core';
2 import { RouterModule, Routes } from '@angular/router';
3 import { AuthGuard } from '../auth.guard';
4 import { AdminGuard } from '../admin.guard';
5 import { logGuard } from '../log.guard';
6 import { LoginComponent } from '../login/login.component';
7 import { MempackagesComponent } from '../mempackages/mempackages.component';
8 import { ProfileComponent } from '../profile/profile.component';
9 import { RegisterComponent } from '../register/register.component';
10 import { SlideComponent } from '../slide/slide.component';
11 import { NotfoundComponent } from '../notfound/notfound.component';
12 import { CartComponent } from '../cart/cart.component';
13 import { PaymentComponent } from '../payment/payment.component';
14 import { ManagerComponent } from '../manager/manager.component';
15 import { AdduserComponent } from '../adduser/adduser.component';
16 import { EdituserComponent } from '../edituser/edituser.component';
17 import { ContactComponent } from '../contact/contact.component';
18 const routes: Routes = [
19   {path: "", component: SlideComponent},
20   {path: "register", component: RegisterComponent, canActivate: [logGuard]},
21   {path: "contact", component: ContactComponent},
22   {path: "login", component: LoginComponent, canActivate: [logGuard]},
23   {path: "profile", component: ProfileComponent, canActivate: [AuthGuard]},
24   {path: "packages", component: MempackagesComponent},
25   {path: "cart", component: CartComponent, canActivate: [AuthGuard]},
26   {path: "payment", component: PaymentComponent, canActivate: [AuthGuard]},
27   {path: "manager", component: ManagerComponent, canActivate: [AdminGuard]},
28   {path: "edituser/:userid", component: EdituserComponent, canActivate:
[AdminGuard]},
29   {path: "adduser", component: AdduserComponent, canActivate: [AdminGuard]},
30   {path: "**", component: NotfoundComponent}
31 ];
32 @NgModule({
33   imports: [RouterModule.forRoot(routes)],
34   exports: [RouterModule]
35 })
36 export class AppRoutingModuleModule { }
```

src\app\app.component.css

```
1 .body{
2     font-family: 'Big Shoulders Stencil Display', cursive; font-family:
'Londrina Solid', cursive;
3 }
```

src\app\app.component.html

```
1 <app-navmain></app-navmain>
2 <router-outlet></router-outlet>
3 <app-marq></app-marq>
4 <app-footer1></app-footer1>
```

src\app\app.component.spec.ts

```
1 import { TestBed } from '@angular/core/testing';
2 import { RouterTestingModule } from '@angular/router/testing';
3 import { AppComponent } from './app.component';
4 describe('AppComponent', () => {
5   beforeEach(async () => {
6     await TestBed.configureTestingModule({
7       imports: [
8         RouterTestingModule
9       ],
10      declarations: [
11        AppComponent
12      ],
13    }).compileComponents();
14  });
15  it('should create the app', () => {
16    const fixture = TestBed.createComponent(AppComponent);
17    const app = fixture.componentInstance;
18    expect(app).toBeTruthy();
19  });
20  it(`should have as title 'ProjectYoga'`, () => {
21    const fixture = TestBed.createComponent(AppComponent);
22    const app = fixture.componentInstance;
23    expect(app.title).toEqual('ProjectYoga');
24  });
25  it('should render title', () => {
26    const fixture = TestBed.createComponent(AppComponent);
27    fixture.detectChanges();
28    const compiled = fixture.nativeElement as HTMLElement;
29    expect(compiled.querySelector('.content
span')?.textContent).toContain('ProjectYoga app is running!');
30  });
31 });
```

src\app\app.component.ts

```
1 import { Component } from '@angular/core';
2 @Component({
3   selector: 'app-root',
4   templateUrl: './app.component.html',
5   styleUrls: ['./app.component.css']
6 })
7 export class AppComponent {
8   title = 'ProjectYoga';
9 }
```

src\app\app.module.ts

```
1 import { NgModule } from '@angular/core';
2 import { BrowserModule } from '@angular/platform-browser';
3 import { FormsModule } from '@angular/forms';
4 import { AppRoutingModule } from './app-routing.module';
5 import { HttpClientModule } from '@angular/common/http';
6 import { StorageServiceModule } from 'ngx-webstorage-service';
7 import { AppComponent } from './app.component';
8 import { NavmainComponent } from './navmain/navmain.component';
9 import { SlideComponent } from './slide/slide.component';
10 import { Footer1Component } from './footer1/footer1.component';
11 import { MarqComponent } from './marq/marq.component';
12 import { LoginComponent } from './login/login.component';
13 import { RegisterComponent } from './register/register.component';
14 import { ProfileComponent } from './profile/profile.component';
15 import { NotFoundComponent } from './notfound/notfound.component';
16 import { MempackagesComponent } from './mempackages/mempackages.component';
17 import { CartComponent } from './cart/cart.component';
18 import { PaymentComponent } from './payment/payment.component';
19 import { ManagerComponent } from './manager/manager.component';
20 import { AdduserComponent } from './adduser/adduser.component';
21 import { EdituserComponent } from './edituser/edituser.component';
22 import { ContactComponent } from './contact/contact.component';
23 @NgModule({
24   declarations: [
25     AppComponent,
26     NavmainComponent,
27     SlideComponent,
28     Footer1Component,
29     MarqComponent,
30     LoginComponent,
31     RegisterComponent,
32     ProfileComponent,
33     NotFoundComponent,
34     MempackagesComponent,
35     CartComponent,
36     PaymentComponent,
37     ManagerComponent,
38     AdduserComponent,
39     EdituserComponent,
40     ContactComponent
41   ],
42   imports: [
43     BrowserModule,
44     AppRoutingModule,
45     FormsModule,
46     HttpClientModule,
47     StorageServiceModule
48   ],
49   providers: [],
50   bootstrap: [AppComponent]
51 })
52 export class AppModule { }
```


src\app\auth.guard.spec.ts

```
1 import { TestBed } from '@angular/core/testing';
2 import { AuthGuard } from '../auth.guard';
3 describe('AuthGuard', () => {
4   let guard: AuthGuard;
5   beforeEach(() => {
6     TestBed.configureTestingModule({});
7     guard = TestBed.inject(AuthGuard);
8   });
9   it('should be created', () => {
10     expect(guard).toBeTruthy();
11   });
12 });
```

src\app\auth.guard.ts

```
1 import { Injectable } from '@angular/core';
2 import { CanActivate, Router } from '@angular/router';
3 import { UsersService } from '../users.service';
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class AuthGuard implements CanActivate {
8   constructor(public userSer : UsersService, public myRouter : Router)
9   {
10  }
11   canActivate(): boolean {
12     if(!this.userSer.isLoggedIn())
13     {
14       this.myRouter.navigateByUrl("/login");
15     }
16     return this.userSer.isLoggedIn();
17   }
18 }
```

src/app/cart/cart.component.html

```
1 <div style="font-family: 'Big Shoulders Stencil Display', cursive;
2 font-family: 'Londrina Solid', cursive; background-color: #c5c6c7 !
important; max-height:200vh; ">
3     <div class="d-grid gap-2 d-md-flex justify-content-md-center">
4         <button class="btn btn me-md-2" style=" margin: 1vw; color:
pink !important; background-color:#1F2833 ;">
5             <h3>CART</h3>
6         </button>
7         <button class="btn btn-info me-md-2" type="button" routerLink="/
profile"
8             style="border:solid #1F2833; background-color: pink;margin:
1vw;">Profile</button>
9         <button class="btn btn-info me-md-2" type="button" routerLink="/
packages"
10             style="border:solid #1F2833; background-color: pink;margin:
1vw;">Back
11     </div> <!-- nav bar-->
12 <!-- Cart-->
13 <hr style="height: 2px; margin: 1%;padding: 0;">
14 <div class="row justify-content-center">
15     <div class="col-lg-5">
16         <div class="card shadow-lg border-0 rounded-lg" style="
border-radius: 12px;">
17             {{msg}}
18             <div class="card-header" style="background-color:
#c5c6c7; border-radius: 12px;">
19                 <h2 style="text-align: center; font-family: 'Big
Shoulders Stencil Display', cursive;
20                                     font-family:
'Londrina Solid', cursive;">Check Out</h2>
21                 <div class="card-body" style="background-color:
whitesmoke; border-radius:12px ;">
22                     <div *ngIf="isPackbuy">
23                         <h3 class="bg-info"
24                             style="display: flex; justify-content:
center; border-radius: 12px; padding:4px; background-color:pink !important ;">
25                             {{packname}}</h3>
26                     </div>
27                     <h5> {{packdesc}} </h5>
28                     <form (submit)="packCb(fo)" #fo="ngForm"
29 *ngIf="userdata">
30                         <fieldset class="form-group">
31                             <div class="row">
32                                 <legend class="col-form-label col-
sm-2 pt-0">
33                                     <h5 class="bg-info"
34                                         style="display: flex; justify-
content: center; border-radius: 12px; padding:4px;background-color:pink !
important ;">
35                                         Level</h5>
36                                 </legend>
37                                 <div class="col-sm-10">
38                                     <div class="form-check">
39                                         <input class="form-check-
input" type="radio" name="level" #level="ngModel"
40                                             id="gridRadios1"
value="Beginner" checked ngModel required>
```

```

41                                     <label class="form-check-
label" for="level">
42                                     Beginner
43                                     </label>
44                                 </div>
45                                 <div class="form-check">
46                                     <input class="form-check-
input" type="radio" name="level" #level="ngModel"
47                                     id="gridRadios2"
value="Intermediate" ngModel>
48                                     <label class="form-check-
label" for="level">
49                                     Intermediate
50                                     </label>
51                                 </div>
52                                 <div class="form-check">
53                                     <input class="form-check-
input" type="radio" name="level" #level="ngModel"
54                                     id="gridRadios3"
value="Advanced" ngModel>
55                                     <label class="form-check-
label" for="level">
56                                     Advanced
57                                     </label>
58                                 </div>
59                             </div>
60                         </div>
61                         <hr style="margin: 1%; height: 2px;">
62                     </fieldset>
63                     <fieldset class="form-group">
64                         <div class="row">
65                             <legend class="col-form-label col-
sm-2 pt-0">
66                                     <h5 class="bg-info"
67                                     style="display: flex; justify-
content: center; border-radius: 12px; padding: 4px; background-color: pink !
important ;">
68                                     Add ons</h5>
69                             </legend>
70                             <div class="col-sm-10">
71                                 <div class="form-check">
72                                     <input class="form-check-
input" type="checkbox" value="DietPlan"
73                                     name="isdietplan" ngModel
#addson1 (change)="diet($event)" ngModel>
74                                     <label class="form-check-
label" for="defaultCheck2">
75                                     Diet Plan (&#8377; 500)
76                                     </label>
77                                 </div>
78                                 <div class="form-check">
79                                     <input class="form-check-
input" type="checkbox" value="PersonalTrainer"
80                                     name="istrain" (change)="train($event)" ngModel>
81                                     <label class="form-check-
label" for="defaultCheck2">
82                                     Personal Trainer (&#8377;
2000)
83                                     </label>
84                                 </div>

```

```

85         <div class="form-check">
86             <input class="form-check-
input" type="checkbox" value="Steambath"
87
name="isbath" (change)="bath($event)" ngModel>
88             <label class="form-check-
label" for="defaultCheck2">
89                 steambath (&#8377; 1000)
90             </label>
91         </div>
92     </div>
93 </div>
94     <hr style="margin: 1%; height: 2px;">
95 </fieldset>
96 <fieldset class="form-group">
97     <div class="row">
98         <legend class="col-form-label col-
sm-2 pt-0">
99             <h5 class="bg-info"
100                 style="display: flex; justify-
content: center; border-radius: 12px; padding: 4px; background-color: pink !
important ;">
101                 Payment</h5>
102             </legend>
103             <div class="col-sm-10">
104                 <div class="form-check">
105                     <input class="form-check-
input" type="radio" name="payment" id="gridRadios1"
106                         value="Gpay" required
ngModel #payment="ngModel" [(ngModel)]= "val" >
107                     <label class="form-check-
label" for="gridRadios1">
108                         Gpay
109                     </label>
110                 </div>
111                 <div class="form-check">
112                     <input class="form-check-
input" type="radio" name="payment" id="gridRadios2"
113                         value="Upi" ngModel
#payment="ngModel" ngModel="val" >
114                     <label class="form-check-
label" for="gridRadios2">
115                         Upi
116                     </label>
117                 </div>
118                 <div class="form-check">
119                     <input class="form-check-
input" type="radio" name="payment" id="gridRadios3"
120                         value="Network Banking"
ngModel #payment="ngModel" [(ngModel)]= "val" >
121                     <label class="form-check-
label" for="gridRadios3">
122                         Network Banking
123                     </label>
124                 </div>
125             </div>
126         </div>
127     </fieldset>
128 <div class="row">
129     <div class="col">
130         <input type="text" class="form-

```

```

131         [ (ngModel) ]="couponcode">
132     </div>
133     <div class="col">
134         <input type="button" class="btn btn-
info" style="background-color: pink !important ;"
Value="Apply" (click)="coupon($event)">
135         <p style="color: red; display: inline-
block; margin-left: 1vw;">{{codemsg}} {{offer}}</p><p style="color: green;
display: inline-block; margin-left: 1vw; background-color: pink !
important ;">{{appliedmsg}} </p>
136     </div>
137 </div>
138 <div style="margin-top: 1vh; display: flex;
justify-content: flex-end;">
139     <h3 style="margin: 1vh;"> <span
class="text-info" style="color: pink !important ;">Adds ons + package &nbsp; </
span> Total: {{pay}} </h3>
140     <button type="submit" class="btn btn-info
btn-lg" style="background-color: pink; border-color: pink;" value="Buy"
141         [disabled]="fo.invalid">Buy</button>
142 </div>
143 </form>
144 </div>
145 <div class="card-footer text-center py-3">
146     <p class="card-text"><small class="text-
muted">Yoga fitness center</small></p>
147     <div class="small"><a routerLink="/contact"
style="color: #1F2833;">Need Help?</a>
148     </div>
149 </div>
150 </div>
151 </div>
152 </div>
153 </div>
154 </div>

```

src/app/cart/cart.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { CartComponent } from './cart.component';
3 describe('CartComponent', () => {
4   let component: CartComponent;
5   let fixture: ComponentFixture<CartComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ CartComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(CartComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src/app/cart/cart.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { PackService } from '../pack.service';
3 import { NgForm } from '@angular/forms';
4 import { Router } from '@angular/router';
5 import { UsersService } from '../users.service';
6 import { ActivatedRoute, Params } from '@angular/router';
7 @Component({
8   selector: 'app-cart',
9   templateUrl: './cart.component.html'
10 })
11 export class CartComponent implements OnInit {
12   isPackbuy = true;
13   code: string = "452dfg";
14   pay = 2000;
15   userdataa: { _id: number, Fname: string, lname: string, uemail: string,
16   userage: Number, usergender: Number, gender: String, userpass: string,
17   userphone: number }
18   codemsg: string;
19   couponcode: string = "";
20   packn: string;
21   isdiet = false;
22   msg: string;
23   userdata: any = localStorage.getItem("loggeduser");
24   date = new Date();
25   offer: string;
26   appliedmsg: string;
27   taxy: number;
28   istax: boolean;
29   isnottax: boolean;
30   val:string;
31   coup:string;
32   packname = localStorage.getItem("packna");
33   packdesc = localStorage.getItem("packds");
34   constructor(public activeRoute: ActivatedRoute, public pac:
35   PackService, public userSer: UsersService, public rooter: Router)
36   { this.packn = pac.packname }
37   ngOnInit(): void {
38     this.activeRoute.params.subscribe((param: Params) => {
39       console.log(param.userid);
40     });
41   }
42   bath(event: any) {
43     if (event.target.checked == true)
44       if (this.couponcode == this.code) {
45         this.pay += 500;
46       }
47     else {
48       this.pay += 1000;
49     }
50   }
51   else {
52     if (this.couponcode == this.code) {
53       this.pay -= 500
54     }
55     else {
56       this.pay -= 1000;
57     }
58   }
59 }
```



```

55 train(event: any) {
56     if (event.target.checked == true)
57         if (this.couponcode == this.code) {
58             this.pay += 1000;
59         }
60     else {
61         this.pay += 2000;
62     }
63     else {
64         if (this.couponcode == this.code) {
65             this.pay -= 1000;
66         }
67     else {
68         this.pay -= 2000;
69     }
70 }
71 }
72 diet(event: any) {
73     if (event.target.checked == true)
74         if (this.couponcode == this.code) {
75             this.pay += 250;
76         }
77     else {
78         this.pay += 500;
79     }
80     else {
81         if (this.couponcode == this.code) {
82             this.pay -= 250;
83         }
84     else {
85         this.pay -= 500;
86     }
87 }
88 }
89 coupon(event: any) {
90     if (this.couponcode == this.code) {
91         event.target.disabled = true;
92         this.pay *= 0.5;
93         this.coup="Applied";
94         this.appliedmsg = "Coupon Applied";
95         this.codemsg = "50%";
96     }
97     else {
98         this.codemsg = "Please Enter valid Code";
99     }
100 }
101 iscoupon() {
102     if (this.couponcode == this.code) {
103         return true;
104     }
105     else {
106         return false;
107     }
108 }
109 packCb(form: NgForm) {
110     console.log("Package Registered");
111     console.log(form.value);
112     var stringToConvert = localStorage.getItem("loggeduser");
113     var numberValue = Number(stringToConvert);
114     console.log(numberValue);
115     form.value._id = numberValue;

```

```
116     form.value.paid= this.pay;
117     form.value.coupon=this.coup;
118     form.value.date = this.date;
119     form.value.pack = this.packn;
120     this.userSer.addpackages(form.value).subscribe((data: string) => {
121         this.msg = data;
122         form.reset();
123     }, (error: any) => {
124         this.msg = "something went wrong";
125     })
126     this.router.navigateByUrl("/payment")
127 }
128 }
```

src\app\contact\contact.component.html

```
1 <div style="font-family: 'Big Shoulders Stencil Display', cursive;
2 font-family: 'Londrina Solid', cursive; background-color: #c5c6c7; min-
height:80vh; border-radius: 12px; padding-bottom: 2vh; ">
3     <div class="d-grid gap-2 d-md-flex justify-content-md-center">
4         <button class="btn btn-info me-md-2" type="button" routerLink="/"
5             style="border:solid #1F2833; background-color: pink;margin:
1vw;">Home</button>
6         <button class="btn btn-info me-md-2" type="button" routerLink="/"
packages"
7             style="border:solid #1F2833; background-color: pink;margin:
1vw;">Package
8         </button>
9     </div>
10    <div class="container-fluid">
11        <div class="card shadow-lg" style="background-color: #c5c6c7;">
12            <br>
13            <div class="card-body">
14                <h5 class="card-title " style="color: #1F2833;">Address</h5>
15                <p style="color: #1F2833;" class="card-text">Powai,Andhrer East, Maharashtra.</
p><hr>
16                <h5 class="card-title " style="color: #1F2833;">Contact Us</h5>
17                <a class="card-text" >  YogaFitnessCenter@gmail.com <br>  9999999999
18            </a>
19            <p class="card-text"><small class="text-muted">Last updated on
06/11/2021</small></p>
20        </div>
21    </div>
22 </div>
23 </div>
```

src/app/contact/contact.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { ContactComponent } from './contact.component';
3 describe('ContactComponent', () => {
4   let component: ContactComponent;
5   let fixture: ComponentFixture<ContactComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ ContactComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(ContactComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\contact\contact.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 @Component({
3   selector: 'app-contact',
4   templateUrl: './contact.component.html'
5 })
6 export class ContactComponent implements OnInit {
7   constructor() { }
8   ngOnInit(): void {
9   }
10 }
```

src\app\edituser\edituser.component.html

```
1 <div class="container" style="min-height: 560px; font-family: 'Big
Shoulders Stencil Display', cursive;
2 font-family: 'Londrina Solid', cursive;">
3   <div class="row justify-content-center" style="padding-top: 4%;">
4     <div class="col-lg-5">
5       <div class="card shadow-lg border-0 rounded-lg mt-5">
6         <div class="card-header " style="background-color: #c5c6c7;">
7           <h2 style="text-align: center; font-family: 'Big Shoulders
Stencil Display', cursive;
8                                     font-family: 'Londrina
Solid', cursive;">Edit Member</h2>
9         <div class="card-body shadow-lg" style="background-color:
#1F2833;">
10           <form #f="ngForm" (submit)="editUser(f)" *ngIf="userdata">
11             <div class="row">
12               <div class="col">
13                 <div class="form-floating mb-3">
14                   <input class="form-control" id="inputfname"
#firstname="ngModel" name="Fname" [ngModel]= "userdata.Fname" type="text"
15                     placeholder="fname" required minlength="3" />
16                   <label for="fname">First Name</label>
17                 </div>
18               </div>
19               <div class="col">
20                 <div class="form-floating mb-3">
21                   <input class="form-control" id="inputlname"
#lastname="ngModel" name="lname" type="text" placeholder="lname" [ngModel]=
"userdata.lname" required />
22                   <label for="lname">Last Name</label>
23                 </div>
24               </div>
25             </div>
26             <p class="alert alert-danger"
*ngIf="firstname.errors?.required && firstname.touched">Please enter the
First Name</p>
27             <p class="alert alert-danger"
*ngIf="firstname.errors?.minlength && firstname.touched">Please enter Min 3
characters </p>
28             <div class="row">
29               <div class="col">
30                 <div class="form-floating mb-3">
31                   <input class="form-control" id="age" type="number"
placeholder="age" name="userage" [ngModel]= "userdata.userage"
#userage="ngModel" required />
32                   <label for="age">Age</label>
33                 </div>
34               <p class="alert alert-danger"
*ngIf="userage.errors?.required && userage.touched" >Please enter the age </p>
35               </div>
36               <div class="col">
37                 <div class="form-check">
38                   <input class="form-check-input" type="radio"
name="gender" value="male" id="flexRadioDefault1" [ngModel]=
"userdata.gender" #usergender="ngModel" required >
39                   <label class="form-check-label"
for="flexRadioDefault1"
40                     style="background-color: whitesmoke; border-radius:
3px; width: 50%; text-align: center;">
```

```

41             Male
42         </label>
43     </div>
44     <div class="form-check" style="margin-top: 10px;">
45         <input class="form-check-input" type="radio"
name="gender" value="female" id="flexRadioDefault2" [ngModel]=
"userdata.gender" #usergender="ngModel" required >
46         <label class="form-check-label"
for="flexRadioDefault2"
47             style="background-color: whitesmoke; border-radius:
3px; width: 50%; text-align: center;">
48             Female
49         </label>
50     </div>
51 </div>
52 <p class="alert alert-danger"
*ngIf="usergender.errors?.required && usergender.touched" >Please click the
gender </p>
53     </div>
54     <div class="form-floating mb-3">
55         <input class="form-control" id="inputphone" type="text"
placeholder="Phone Number" name="userphone" [ngModel]= "userdata.userphone"
#userphone="ngModel" required pattern="^(0|[1-9][0-9]*)$" maxlength="10"/>
56         <label for="inputphone">Phone Number</label>
57     </div>
58     <p class="alert alert-danger"
*ngIf="userphone.errors?.required && userphone.touched">Please enter the
Phone number</p>
59     <p class="alert alert-danger"
*ngIf="userphone.errors?.pattern">please enter the valid phone number</p>
60     <div class="form-floating mb-3">
61         <input class="form-control" id="inputEmail" name="uemail"
type="email" #uemail="ngModel" [ngModel]=
"userdata.uemail" placeholder="name@example.com" required email />
62         <label for="inputEmail">Email address</label>
63     </div>
64     <p class="alert alert-danger"
*ngIf="uemail.errors?.required && uemail.touched">Please enter the Email</p>
65     <p class="alert alert-danger" *ngIf="uemail.errors?.email
&& uemail.touched">Please enter the valid Email
66     </p>
67     <div class="form-floating mb-3">
68         <input class="form-control" id="inputPassword"
type="password" placeholder="Password" name="userpass"
69         [ngModel]= "userdata.userpass" #userpass="ngModel"
required password minlength="8" />
70         <label for="inputPassword">Create Password</label>
71     </div>
72     <p class="alert alert-danger"
*ngIf="userpass.errors?.required && userpass.touched">Please Create a
73     password</p>
74     <p class="alert alert-danger"
*ngIf="userpass.errors?.minlength">Please Create a strong password (min 8
75     character) </p>
76     <h4 style="color: pink; margin: 1vw; display: flex; justify-
content: center;">Package</h4>
77     <select class="form-select form-select-lg mb-3"
name="pack" [ngModel]= "userdata.pack" #pack="ngModel" aria-label=".form-
select-lg example"
78     placeholder="Select Package">
79     <option selected>Select Package</option>
80

```

```

81         <option value="Hatha Yoga">Hatha Yoga</option>
82         <option value="Vinyasa Yoga">Vinyasa Yoga</option>
83         <option value="Iyengar Yoga">Iyengar Yoga</option>
84         <option value="Ashtanga Yoga">Ashtanga Yoga</option>
85         <option value="Bikram Yoga">Bikram Yoga</option>
86         <option value="Kundalini Yoga">Kundalini Yoga</option>
87     </select>
88     <h4 style="color: pink; margin: 1vw; display: flex; justify-
content: center;">Level</h4>
89     <div class="form-check" style="display: inline-block; color:
#c5c6c7; margin-left: 2vw; ">
90         <input class="form-check-input" type="radio"
name="level" [ngModel]= "userdata.level" #level="ngModel" id="gridRadios1"
91             value="Beginner" checked ngModel >
92         <label class="form-check-label" for="level">
93             Beginner
94         </label>
95     </div>
96     <div class="form-check" style="display: inline-block; color:
#c5c6c7; margin-left: 2vw; ">
97         <input class="form-check-input" type="radio"
name="level" [ngModel]= "userdata.level" #level="ngModel" id="gridRadios2"
98             value="Intermediate" ngModel>
99         <label class="form-check-label" for="level">
100             Intermediate
101         </label>
102     </div>
103     <div class="form-check" style="display: inline-block; color:
#c5c6c7; margin-left: 2vw; ">
104         <input class="form-check-input" type="radio"
name="level" [ngModel]= "userdata.level" #level="ngModel" id="gridRadios3"
105             value="Advanced" ngModel>
106         <label class="form-check-label" for="level">
107             Advanced
108         </label>
109     </div>
110     <div>
111         <h4 style="color: pink; margin: 1vw; display: flex;
justify-content: center;">Adds on</h4>
112         <div class="form-check" style="display: inline-block;
color: #c5c6c7; margin-left: 2vw; ">
113             <input class="form-check-input" type="checkbox"
id="flexCheckDefault" [ngModel]="userdata.isbath" value="Steambath"
114                 name="isbath">
115             <label class="form-check-label" for="flexCheckDefault">
116                 Steambath
117             </label>
118         </div>
119         <div class="form-check" style="display: inline-block;
color: #c5c6c7; margin-left: 2vw; ">
120             <input class="form-check-input"
type="checkbox" [ngModel]="userdata.isdietplan" value="DietPlan"
name="isdietplan"
121                 id="flexCheckDefault">
122             <label class="form-check-label" for="flexCheckDefault">
123                 DietPlan
124             </label>
125         </div>
126         <div class="form-check" style="display: inline-block;
color: #c5c6c7; margin-left: 2vw; ">
127             <input class="form-check-input" type="checkbox"

```



```

id="flexCheckDefault" [ngModel]="userdata.istrain" value="PersonalTrainer"
128         name="istrain">
129         <label class="form-check-label" for="flexCheckDefault">
130             PersonalTrainer
131         </label>
132     </div>
133 </div>
134 <div class="d-flex align-items-center justify-content-
between mt-4 mb-0">
135     <button class="btn btn-info btn-block" value="Register"
type="submit" [disabled]="f.invalid" style="background-color: pink !important;
center; font-family: 'Big Shoulders Stencil Display', cursive;
136                                     font-family:
'Londrina Solid', cursive;"> Register</button><h5 style="color: pink;">{{msg}}
</h5>
137     </div>
138 </form>
139 </div>
140 <div class="card-footer text-center py-3">
141     <p class="card-text"><small class="text-muted">Yoga fitness
center</small></p>
142     <button class="btn btn-info me-md-2" type="button"
routerLink="/manager" style="border: solid #1F2833; margin: 1vw; background-
color: pink !important;">Back
143     </button>
144 </div>
145 </div>
146 </div>
147 </div>
148 </div>
149 </div>

```

src\app\edituser\edituser.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { EdituserComponent } from './edituser.component';
3 describe('EdituserComponent', () => {
4   let component: EdituserComponent;
5   let fixture: ComponentFixture<EdituserComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ EdituserComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(EdituserComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\edituser\edituser.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { NgForm } from '@angular/forms';
3 import { ActivatedRoute, Params } from '@angular/router';
4 import { UsersService } from '../users.service';
5 @Component({
6   selector: 'app-edituser',
7   templateUrl: './edituser.component.html'
8 })
9 export class EdituserComponent implements OnInit {
10   msg: string;
11   date = new Date();
12   userdata: { _id: number, Fname: string, lname: string, uemail:string,
13   userage:Number, usergender:Number, gender:String, userpass:string,
14   userphone:number ,level:string;
15   isbath:boolean,istrain:boolean,isdietplan:boolean,pack:string }
16   constructor(public activeRoute: ActivatedRoute, public userSer:
17   UsersService) { }
18   ngOnInit(): void {
19     this.activeRoute.params.subscribe((param: Params) => {
20       console.log(param);
21       if (param.userid) {
22         this.userSer.getSingleUserData(param.userid).subscribe((data:
23         any[]) => {
24           console.log(data);
25           this.userdata = data[0];
26         }, (error: any) => {
27           console.log(error);
28         });
29       }
30     });
31   }
32   editUser(form: NgForm) {
33     form.value._id = this.userdata._id
34     form.value.date=this.date;
35     console.log(form.value);
36     this.userSer.editsingleUserdata(form.value).subscribe((data: string)
37     => {
38       this.msg = data;
39     }, (error: any) => {
40       this.msg = "something went wrong";
41     });
42   }
43 }
```

src\app\footer1\footer1.component.html

```
1 <div class="container-fluid" style="background-color:#1F2833;">
2   <footer class="d-flex flex-wrap justify-content-between align-items-
center py-3 border-top" style="background-color:#1F2833;">
3     <p class="col-md-4 mb-0 text-muted">&copy; 2024 YogaFitnessCenter,
GPM College</p>
4     <a routerLink="/" class="col-md-4 d-flex align-items-center justify-
content-center mb-3 mb-md-0 me-md-auto link-dark text-decoration-none">
5       
6     </a>
7     <ul class="nav col-md-4 justify-content-end" >
8       <li class="nav-item"><a routerLink="/" class="nav-link px-2 text-
muted" style="color: pink !important; font-size: 100%; font-family: 'Big
Shoulders Stencil Display', cursive; font-family: 'Londrina Solid',
cursive;">Home</a></li>
9       <li class="nav-item"><a routerLink="packages/" class="nav-link
px-2 text-muted" style="color: pink !important; font-size: 100%; font-family:
'Big Shoulders Stencil Display', cursive; font-family: 'Londrina Solid',
cursive;">Package</a></li>
10      <li class="nav-item"><a routerLink="contact/" class="nav-link px-2
text-muted" style="color: pink !important; font-size: 100%; font-family: 'Big
Shoulders Stencil Display', cursive; font-family: 'Londrina Solid',
cursive;">Contact Us</a></li>
11    </ul>
12  </footer>
13 </div>
```

src\app\footer1\footer1.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { Footer1Component } from './footer1.component';
3 describe('Footer1Component', () => {
4   let component: Footer1Component;
5   let fixture: ComponentFixture<Footer1Component>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ Footer1Component ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(Footer1Component);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\footer1\footer1.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 @Component({
3   selector: 'app-footer1',
4   templateUrl: './footer1.component.html'
5 })
6 export class Footer1Component implements OnInit {
7   constructor() { }
8   ngOnInit(): void {
9   }
10 }
```

src\app\log.guard.spec.ts

```
1 import { TestBed } from '@angular/core/testing';
2 import { LogGuard } from './log.guard';
3 describe('LogGuard', () => {
4   let guard: LogGuard;
5   beforeEach(() => {
6     TestBed.configureTestingModule({});
7     guard = TestBed.inject(LogGuard);
8   });
9   it('should be created', () => {
10     expect(guard).toBeTruthy();
11   });
12 });
```

src\app\log.guard.ts

```
1 import { Injectable } from '@angular/core';
2 import { CanActivate, Router } from '@angular/router';
3 import { UsersService } from '../users.service';
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class logGuard implements CanActivate {
8   constructor(public userSer : UsersService, public myRouter : Router)
9   {
10  }
11  canActivate(): boolean {
12    if(!this.userSer.isLoggedIn())
13    {
14      this.myRouter.navigateByUrl("/profile");
15    }
16    return !this.userSer.isLoggedIn();
17  }
18 }
```


src\app\login\login.component.html

```
1 <div class="container" style="min-height: 560px; font-family: 'Big
Shoulders Stencil Display', cursive;
2 font-family: 'Londrina Solid', cursive;">
3     <div class="row justify-content-center" style="padding-top: 4%;">
4         <div class="col-lg-5">
5             <div class="card shadow-lg border-0 rounded-lg mt-5">
6                 <div class="card-header" style="background-color:
#c5c6c7;">
7                     <h2 style="text-align: center;">Login</h2>
8                     <div class="card-body" style="background-color:
#1F2833;"><p class="alert alert-danger" *ngIf="msg" style="align-self:
center;">{{msg}}</p>
9                         <form #f="ngForm" (submit)="doLogin(f)">
10                             <div class="form-floating mb-3">
11                                 <input class="form-control"
id="inputEmail" type="email" name="uemail"
12                                     placeholder="name@example.com" ngModel/
>
13                                 <label for="inputEmail">Email address</
label>
14                             </div>
15                             <div class="form-floating mb-3">
16                                 <input class="form-control"
id="inputPassword" type="password" placeholder="Password" name="userpass"
ngModel/>
17                                 <label for="inputPassword">Password</label>
18                             </div>
19                             <div class="d-flex align-items-center justify-
content-between mt-4 mb-0">
20                                 <button type="submit" class="btn btn-info
btn-block" name="upassword" style="text-align: center;background-color:
pink!important; border-color: pink !important;" >Login</button>
21                                 <!-- routerLink="/profile"-->
22                             </div>
23                         </form>
24                     </div>
25                     <div class="card-footer text-center py-3">
26                         <div class="small" ><a routerLink="/register"
style="color: #1F2833;">Need an account? Sign up!</a></div>
27                     </div>
28                 </div>
29             </div>
30         </div>
31     </div>
32 </div>
```

src\app\login\login.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { LoginComponent } from '../login.component';
3 describe('LoginComponent', () => {
4   let component: LoginComponent;
5   let fixture: ComponentFixture<LoginComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ LoginComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(LoginComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\login\login.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { NgForm } from '@angular/forms';
3 import { ActivatedRoute, Router } from '@angular/router';
4 import { UsersService } from '../users.service';
5 @Component({
6   selector: 'app-login',
7   templateUrl: './login.component.html'
8 })
9 export class LoginComponent implements OnInit {
10   msg : string;
11   constructor(public userSer : UsersService, public myRouter :
Router,public activeRoute: ActivatedRoute) { }
12   ngOnInit(): void {
13   }
14   doLogin(form:NgForm)
15   {
16     console.log(form.value);
17     this.userSer.doUserLogin(form.value).subscribe((data:any[])=>{
18       console.log(data);
19       if(data.length==0)
20       {
21         this.msg = "Invalid Login";
22       }
23       else {
24         localStorage.setItem("loggeduser", data[0]._id);
25         localStorage.setItem("loggedusername", data[0].Fname);
26         localStorage.setItem("loggedusernamee", data[0].lname);
27         localStorage.setItem("loggedusergen", data[0].gender);
28         localStorage.setItem("loggeduserage", data[0].usage);
29         this.myRouter.navigateByUrl("/profile");
30       }
31       if(data[0]._id==1711611304560)
32       {
33         localStorage.setItem("loggedadmin", data[0]._id);
34       }
35     }, (error:any)=>{
36       console.log(error);
37       this.msg = "Something went wrong";
38     });
39   }
40 }
```



```

52             <td>{{user.pack}}</td>
53             <td>{{user.level}}</td>
54             <td>{{user.isbath}}</td>
55             <td>{{user.isdietplan}}</td>
56             <td>{{user.istrain}}</td>
57             <td>{{user.date|slice :0:10}}</td>
58             <td>{{user.paid}}</td>
59             <td>{{user.coupon}}</td>
60             <td style="display: flex; margin: 1vh;
background-color:#c5c6c7">
61                 &nbsp;
66             </td>
67         </tr>
68     </tbody>
69 </table>
70 <p *ngIf="users.length==0"> No Records Found</p>
71 </div>
72 <div class="col"></div>
73 </div>
74 </div>
75 </div>

```

src\app\manager\manager.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { ManagerComponent } from './manager.component';
3 describe('ManagerComponent', () => {
4   let component: ManagerComponent;
5   let fixture: ComponentFixture<ManagerComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ ManagerComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(ManagerComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\manager\manager.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { Router } from '@angular/router';
3 import { UsersService } from '../users.service';
4 @Component({
5   selector: 'app-manager',
6   templateUrl: './manager.component.html'
7 })
8 export class ManagerComponent implements OnInit {
9   msg: string;
10  users: any[] = [];
11  constructor(public userSer: UsersService) { }
12  ngOnInit(): void {
13    this.userSer.getAllUsers().subscribe((data: any[]) => {
14      console.log(data);
15      this.users = data;
16    }, (error: any) => {
17      this.msg = "Something went wrong"
18    });
19  }
20  deleteUser(userId: number) {
21    if (confirm("Do you want to delete this user?")) {
22      //console.log("User deleted with id"+userId);
23      this.userSer.deleteUserData(userId).subscribe((data: string) => {
24        this.msg = data;
25        var index = this.users.findIndex((obj) => {
26          return obj._id == userId;
27        });
28        this.users.splice(index, 1);
29      }, (error: any) => {
30        this.msg = "something went wrong";
31      });
32    }
33  }
34  doSearch(search: string) {
35    this.userSer.searchUsers(search).subscribe((data: any[]) => {
36      console.log(data);
37      this.users=data;
38    }, (error: any) => {
39      console.log(error);
40    });
41  }
42 }
```

src\app\marq\marq.component.html

```
1 <div style="z-index:0;">
2   <marquee direction="left" height="10%" style="background-color:
#1F2833 ;" >
3     <p style="color:#daa520; border-radius: 5px;"><strong> NOW GET
Yoga fitness center PACKAGES AT 50% OFF !!! Coupon code: 452dfg </strong></p>
4   </marquee>
5 </div>
```


src\app\marq\marq.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { MarqComponent } from './marq.component';
3 describe('MarqComponent', () => {
4   let component: MarqComponent;
5   let fixture: ComponentFixture<MarqComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ MarqComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(MarqComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\marq\marq.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 @Component({
3   selector: 'app-marq',
4   templateUrl: './marq.component.html'
5 })
6 export class MarqComponent implements OnInit {
7   constructor() { }
8   ngOnInit(): void {
9   }
10 }
```

src/app/mempackages/mempackages.component.html

```
1 <div
2   style="font-family: 'Big Shoulders Stencil Display', cursive;
3   font-family: 'Londrina Solid', cursive; background-color: #c5c6c7; border-
radius: 12px; padding-left: 2vh; padding-right: 2vh;">
4   <div class="d-grid gap-2 d-md-flex justify-content-md-center">
5     <button class="btn btn-info me-md-2" type="button" routerLink="/"
6     style="border:solid #1F2833;background-color: pink; margin:
1vw;">Home
7   </button>
8   <button class="btn btn-info me-md-2" type="button" routerLink="/"
profile"
9     style="border:solid #1F2833;background-color: pink; margin:
1vw;">Profile
10  </button>
11  <button class="btn btn  me-md-2" style=" margin: 1vw; color: pink;
background-color:#1F2833 ;">
12    <h3>Package</h3>
13  </button>
14 </div>
15 <hr style="height: 2px; margin: 1%;padding: 0;">
16 <div class="row row-cols-1 row-cols-md-3 g-1">
17   <div class="col">
18     <div class="card shadow-lg" style="margin: 2vh;">
19       
21       <div class="card-body">
22         <div style="background-color: #1F2833; color: pink; border-
radius: 5px;">
23           <h4 class="card-title" style=" text-align:
center;">{{pac.HathaYogaThen }}</h4><!-- Hatha Yoga-Card 1-->
24         </div>
25         <p class="card-text">{{pac.HathaYogaS}}</p>
26         <div style=" color: #1F2833; display: flex; justify-content:
flex-end; border-radius: 5px; ">
27           <h5 class="card-title bg-info" style=" text-align: center;
padding: .4vh; border-radius: 4px;background-color: pink !important;">
Price :&#8377;2000</h5>
28         </div>
29       </div>
30       <button type="button" class="btn btn-info" style="background-
color: pink; border-color: pink;"
31       (click)="pac.HathaYogaSBuy()">BUY</button>
32     </div>
33   </div>
34   <div class="col">
35     <div class="card shadow-lg" style="margin: 2vh;">
36       
38       <div class="card-body">
39         <div style="background-color: #1F2833; color: pink; border-
radius: 4px;">
40           <h4 class="card-title" style=" text-align:
center;">{{pac.VinyasaYogaM}}</h4> <!--Weigthlifting-card 2-->
41         </div>
42         <p class="card-text">{{pac.VinyasaYogaNa}}</p>
```

```

43         <div style=" color: #1F2833; display: flex; justify-content:
flex-end; border-radius: 5px; ">
44             <h5 class="card-title bg-info" style=" text-align: center;
padding: .4vh; border-radius: 4px;background-color: pink !important;">
Price :&#8377;2000</h5>
45         </div>
46     </div>
47     <button type="button" class="btn btn-info"
48         (click)="pac.VinyasaYogaMBuy()" style="background-color: pink;
border-color: pink;">BUY</button>
49 </div>
50 </div>
51 <div class="col">
52     <div class="card shadow-lg" style="margin: 2vh;">
53         
55         <div class="card-body">
56             <div style="background-color: #1F2833; color: pink; border-
radius: 5px;">
57                 <h4 class="card-title" style=" text-align:
center;">{{pac.IyengarYoga}}</h4>
58                 <!--Iyengar Yoga --card3 -->
59             </div>
60             <p class="card-text">{{pac.IyengarYo}}</p>
61             <div style=" color: #1F2833; display: flex; justify-content:
flex-end; border-radius: 5px; ">
62                 <h5 class="card-title bg-info" style=" text-align: center;
padding: .4vh; border-radius: 4px;background-color: pink !important;">
Price :&#8377;2000</h5>
63             </div>
64             </div><button type="button" class="btn btn-info"
65                 (click)="pac.IyengarYogaBuy()" style="background-color: pink;
border-color: pink;">BUY</button>
66         </div>
67     </div>
68 <div class="col">
69     <div class="card shadow-lg" style="margin: 2vh;">
70         
72         <div class="card-body">
73             <div style="background-color: #1F2833; color: pink; border-
radius: 5px;">
74                 <h4 class="card-title" style=" text-align:
center;">{{pac.AshtangaYoga}}</h4>
75                 <!--Kick boxing -- card4-->
76             </div>
77             <p class="card-text">{{pac.AshtangaYo}}</p>
78             <div style=" color: #1F2833; display: flex; justify-content:
flex-end; border-radius: 5px; ">
79                 <h5 class="card-title bg-info" style=" text-align: center;
padding: .4vh; border-radius: 4px;background-color: pink !important;">
Price :&#8377;2000</h5>
80             </div>
81             </div><button type="button" class="btn btn-info"
82                 (click)="pac.AshtangaYogaBuy()" style="background-color: pink;
border-color: pink;">BUY</button>
83         </div>
84     </div>

```

```

85     <div class="col">
86         <div class="card shadow-lg" style="margin: 2vh;">
87             
89             <div class="card-body">
90                 <div style="background-color: #1F2833; color: pink; border-
radius: 5px;">
91                     <h4 class="card-title" style=" text-align:
center;">{{pac.BikramYoga}}</h4><!-- Bikram Yoga -- card5-->
92                 </div>
93                 <p class="card-text">{{pac.BikramYo}}</p>
94                 <div style=" color: #1F2833; display: flex; justify-content:
flex-end; border-radius: 5px; ">
95                     <h5 class="card-title bg-info" style=" text-align: center;
padding: .4vh; border-radius: 4px;background-color: pink !important;">
Price :&#8377;2000</h5>
96                 </div>
97             </div><button type="button" class="btn btn-info"
98                 (click)="pac.BikramYogaBuy()" style="background-color: pink;
border-color: pink;">BUY</button>
99         </div>
100     </div>
101     <div class="col">
102         <div class="card shadow-lg" style="margin: 2vh;">
103             
104             <div class="card-body">
105                 <div style="background-color: #1F2833; color: pink; border-
radius: 5px;">
106                     <h4 class="card-title" style=" text-align:
center;">{{pac.KundaliniYoga}}</h4>
107                     <!--sport--card5-->
108                 </div>
109                 <p class="card-text">
{{pac.KundaliniYo}} </p>
110                 <div style=" color: #1F2833; display: flex; justify-content:
flex-end; border-radius: 5px; ">
111                     <h5 class="card-title bg-info" style=" text-align: center;
padding: .4vh; border-radius: 4px;background-color: pink !important;">
Price :&#8377;2000</h5>
112                 </div>
113             </div><button type="button" class="btn btn-info"
114                 (click)="pac.KundaliniYogaBuy()" style="background-color: pink;
border-color: pink; border-color: pink;">BUY</button>
115             </div>
116         </div>
117     </div>
118 </div>
119 </div>

```

src/app/mempackages/mempackages.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { MempackagesComponent } from './mempackages.component';
3 describe('MempackagesComponent', () => {
4   let component: MempackagesComponent;
5   let fixture: ComponentFixture<MempackagesComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ MempackagesComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(MempackagesComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\mempackages\mempackages.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { PackService } from '../pack.service';
3 @Component({
4   selector: 'app-mempackages',
5   templateUrl: './mempackages.component.html'
6 })
7 export class MempackagesComponent implements OnInit {
8   useriid:any= localStorage.getItem("loggeduser");
9   constructor(public pac:PackService) {
10   }
11   ngOnInit(): void {console.log(this.useriid);
12   }
13 }
```

src\app\navmain\navmain.component.css

```
1 .nav-item :hover {
2     background-color: black;
3 }
4 .nav-item button {
5     border-bottom-color: pink !important;
6     border-right-color: pink !important;
7     border-top-color: pink !important;
8     border-left-color: pink !important;
9 }
10 .nav-item {
11     border-bottom-color: pink !important;
12 }
```


src\app\navmain\navmain.component.html

```
1 <div style=" font-family: 'Big Shoulders Stencil Display', cursive;
2 font-family: 'Londrina Solid', cursive;">
3     <nav class="navbar navbar-expand-lg navbar-light "
4         style="background-color: #1F2833; border-top: 4px solid pink; z-
index: 1;">
5         <div class="container-fluid">
6             <a class="navbar-brand" routerLink="">
7                 
8             </a>
9             <a class="navbar-brand" routerLink="" style="color: pink;">
10                 <h1 style="font-family: 'Big Shoulders Stencil Display',
cursive;">Yoga Fitness center</h1>
11             </a>
12             <button class="navbar-toggler " type="button" style="color:
pink;" data-bs-toggle="collapse"
13                 data-bs-target="#navbarSupportedContent" aria-
controls="navbarSupportedContent" aria-expanded="false"
14                 aria-label="Toggle navigation" style="background-color:
pink;">
15                 <span class="navbar-toggler-icon "></span>
16             </button>
17             <div class="collapse navbar-collapse"
id="navbarSupportedContent" style="margin-left: 40%;">
18                 <ul class="navbar-nav me-auto mb-lg-0">
19                     <li class="nav-item" style="margin-right: 1vh;">
20                         <h3 style="color: pink"
*ngIf="userSer.isLoggedIn()"> Welcome! &nbsp; {{userSer.name}} </h3>
21                     </li>
22                     <li class="nav-item" style="margin-right: 1vh;">
23                         <button class="btn btn-outline-info"
routerLink="/" style="color: pink; font-size: 100%; font-family: 'Big
Shoulders Stencil Display', cursive; font-family: 'Londrina Solid',
cursive;">Home</button>
24                     </li>
25                     <li class="nav-item" style="margin-right: 1vh;">
26                         <button class="btn btn-outline-info" routerLink="/
packages" style="color: pink; font-size: 100%; font-family: 'Big Shoulders
Stencil Display', cursive; font-family: 'Londrina Solid', cursive;">Packages</
button>
27                     </li>
28                     <li class="nav-item" style="margin-right: 1vh;">
29                         <button class="btn btn-outline-info" routerLink="/
register" style="color: pink; font-size: 100%; font-family: 'Big Shoulders
Stencil Display', cursive; font-family: 'Londrina Solid', cursive;" *ngIf="!
userSer.isLoggedIn()">Register</button>
30                     </li>
31                     <li class="nav-item" style="margin-right: 1vh;">
32                         <button class="btn btn-outline-info" routerLink="/
login" style="color: pink; font-size: 100%; font-family: 'Big Shoulders
Stencil Display', cursive; font-family: 'Londrina Solid', cursive;" *ngIf="!
userSer.isLoggedIn()">Login</button>
33                     </li>
34                     <li class="nav-item" style="margin-right: 1vh;">
35                         <button class="btn btn-outline-info" routerLink="/
profile" style="color: pink; font-size: 100%; font-family: 'Big Shoulders
Stencil Display', cursive; font-family: 'Londrina Solid', cursive;"
*ngIf="userSer.isLoggedIn()">Profile</button>
```

```
36             </li>
37             <li class="nav-item" style="margin-right: 1vh;">
38                 <button class="btn btn-outline-info" routerLink="/
manager" style="color: pink; font-size: 100%; font-family: 'Big Shoulders
Stencil Display', cursive; font-family: 'Londrina Solid', cursive;"
*ngIf="userSer.isAdmin()">manager</button>
39             </li>
40             <li class="nav-item" style="margin-right: 1vh;">
41                 <button class="btn btn-outline-info" routerLink="/
register" style="color: pink; font-size: 100%; font-family: 'Big Shoulders
Stencil Display', cursive; font-family: 'Londrina Solid', cursive;"
*ngIf="userSer.isLoggedIn()"
42                     (click)="doLogout()">logout</button>
43             </li>
44         </ul>
45     </div>
46 </div>
47 </nav>
48 </div>
```

src\app\navmain\navmain.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { NavmainComponent } from './navmain.component';
3 describe('NavmainComponent', () => {
4   let component: NavmainComponent;
5   let fixture: ComponentFixture<NavmainComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ NavmainComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(NavmainComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src/app/navmain/navmain.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { Router } from '@angular/router';
3 import { UsersService } from '../users.service';
4 import { ActivatedRoute } from '@angular/router';
5 @Component({
6   selector: 'app-navmain',
7   templateUrl: './navmain.component.html',
8   styleUrls: ['./navmain.component.css']
9 })
10 export class NavmainComponent implements OnInit {
11   name: string;
12   constructor(public userSer: UsersService, public activeRoute:
ActivatedRoute, public myRouter: Router) { }
13   ngOnInit(): void {
14   }
15   doLogout() {
16     localStorage.clear();
17     this.myRouter.navigateByUrl("/");
18   }
19 }
```

src\app\notfound\notfound.component.html

```
1 <div class="cen" style="height: 50vh; " >
2   <div class="alert alert-info" role="alert" style="margin-top: 20vh;">
3     <h5 style="text-align: center;"> Page NOT FOUND!</h5>
4   </div>
5 </div>
```

src\app\notfound\notfound.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { NotfoundComponent } from './notfound.component';
3 describe('NotfoundComponent', () => {
4   let component: NotfoundComponent;
5   let fixture: ComponentFixture<NotfoundComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ NotfoundComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(NotfoundComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\notfound\notfound.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 @Component({
3   selector: 'app-notfound',
4   templateUrl: './notfound.component.html'
5 })
6 export class NotfoundComponent implements OnInit {
7   constructor() { }
8   ngOnInit(): void {
9   }
10 }
```

src/app/pack.service.spec.ts

```
1 import { TestBed } from '@angular/core/testing';
2 import { PackService } from './pack.service';
3 describe('PackService', () => {
4   let service: PackService;
5   beforeEach(() => {
6     TestBed.configureTestingModule({});
7     service = TestBed.inject(PackService);
8   });
9   it('should be created', () => {
10     expect(service).toBeTruthy();
11   });
12 });
```


src/app/pack.service.ts

```
1 import { Injectable } from '@angular/core';
2 import { HttpClient, HttpClientModule } from '@angular/common/http';
3 import { Router } from '@angular/router';
4 @Injectable({
5   providedIn: 'root'
6 })
7 export class PackService {
8   packname="";
9   packdesc="";
10  accountStatus="Inactive";
11  HathaYogaThen ="Hatha Yoga";
12  HathaYogaS ="Hatha Yoga, considered a gentler form of Yoga, comprises of
basic introductory poses in which you're required to hold each pose for a few
breaths.It's important to be empty stomach to practice yoga, trying remaining
quite and silent while practicing it.";
13  VinyasaYogaM="Vinyasa Yoga";
14  VinyasaYogaNa="Vinyasa Yoga helps in developing a flow in your body
while practicing Yoga by linking movement with breath. It can lead to rise in
your heart rate as the pace is bound to increase with each pose. It focuses
on synchronizing the breath with a continuous flow of postures improving our
blood circulation.";
15  IyengarYoga="Iyengar Yoga";
16  IyengarYo="The poses are for longer periods as they help you align your
body posture in the right way. It's therapeutic for several conditions and
helps in cultivating strength and flexibility. Toned muscles, pain relief
from different joints of the body."
17  AshtangaYoga="Ashtanga Yoga";
18  AshtangaYo="Ashtanga Yoga is only for those looking for a challenging
and orderly approach to yoga.It means - eight limbs or branches of yoga which
are - Yama - moral codes, Niyama -self-purification and study, Asana -
posture, Pranayama - breath control. "
19  KundaliniYoga="Kundalini Yoga";
20  KundaliniYo=" Kundalini Yoga includes performing repetitive physical
exercises coupled with intense breath work while chanting and meditating. It
releases the untapped energy which resides within you, providing a sense of
self-awareness. It's also a blend of bhakti yoga.";
21  BikramYo="Bikram Yoga helps in increasing blood flow, and helps in
improving posture, balance, and mobility.Practice this series of 26 poses and
2 breathing exercises in a room heated to around 105 degrees and 40% humidity
and get drenched in sweat.";
22  BikramYoga="Bikram Yoga";
23  stbath=1000;
24  pt=2000;
25  dplan=500;
26  name=localStorage.getItem("loggedusername");
27  namee=localStorage.getItem("loggedusernamee");
28  gender=localStorage.getItem("loggedusergen");
29  age=localStorage.getItem("loggeduserage");
30  constructor(public http:HttpClient,public roter: Router) {
31    console.log("service created");
32  }
33  HathaYogaSBuy(){
34    this.packname=this.HathaYogaThen ;
35    localStorage.setItem("packna", this.packname);
36    this.packdesc=this.HathaYogaS;
37    localStorage.setItem("packds",this.packdesc);
38    this.roter.navigateByUrl("/cart")
39  }
```

```

40  VinyasaYogaMBuy(){
41      this.packname=this.VinyasaYogaM;
42      localStorage.setItem("packna", this.packname);
43      this.packdesc=this.VinyasaYogaNa;
44      localStorage.setItem("packds",this.packdesc);
45      this.rooter.navigateByUrl("/cart")
46  }
47  IyengarYogaBuy(){
48      this.packname=this.IyengarYo;
49      localStorage.setItem("packna", this.packname);
50      this.packdesc=this.IyengarYoga;
51      localStorage.setItem("packds",this.packdesc);
52      this.rooter.navigateByUrl("/cart")
53  }
54  AshtangaYogaBuy(){
55      this.packname=this.AshtangaYo;
56      localStorage.setItem("packna", this.packname);
57      this.packdesc=this.AshtangaYoga;
58      localStorage.setItem("packds",this.packdesc);
59      this.rooter.navigateByUrl("/cart")
60  }
61  BikramYogaBuy(){
62      this.packname=this.BikramYoga;
63      localStorage.setItem("packna", this.packname);
64      this.packdesc=this.BikramYo;
65      localStorage.setItem("packds",this.packdesc);
66      this.rooter.navigateByUrl("/cart")
67  }
68  KundaliniYogaBuy(){
69      this.packname=this.KundaliniYoga;
70      localStorage.setItem("packna", this.packname);
71      this.packdesc=this.KundaliniYo;
72      localStorage.setItem("packds",this.packdesc);
73      this.rooter.navigateByUrl("/cart")
74  }
75  }

```

src/app/payment/payment.component.html

```
1 <div class="container" style="font-family: 'Big Shoulders Stencil
Display', cursive;
2 font-family: 'Londrina Solid', cursive; background-color: #c5c6c7; max-
height:200vh; padding-bottom: 20vh;">
3 <div class="d-grid gap-2 d-md-flex justify-content-md-center">
4   <button class="btn btn-info me-md-2" type="button" routerLink="/
profile"
5     style="border:solid #1F2833;background-color: pink; margin:
1vw;">Profile</button>
6   <button class="btn btn-info me-md-2" type="button" routerLink="/
packages"
7     style="border:solid #1F2833;background-color: pink; margin:
1vw;">Packages
8   </button>
9 </div> <!-- nav bar-->
10 <hr style="height: 2px; margin: 1%;padding: 0;">
11   <div class="card text-center shadow-lg" style="margin: 4vw;
background-color: #c5c6c7;">
12     <div class="card-header" >
13       {{packname}} has been purchased
14     </div>
15     <div class="card-body" style="background-color:whitesmoke;">
16       <h2 class="card-title text-success">Payment Successfull</h2>
17       <p class="card-text">Thanks for choosing  Yoga fitness center</
p>
18       <a routerLink="/profile" class="btn btn-info"
style="background-color: pink;
19         border-bottom-color: pink !important;
20         border-right-color: pink !important;
21         border-top-color: pink !important;
22         border-left-color: pink !important;
23       ">Lets Go  &raquo;</a>
24     </div>
25     <div class="card-footer text-muted">
26       Yoga fitness center
27     </div>
28   </div>
29 </div>
```

src\app\payment\payment.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { PaymentComponent } from '../payment.component';
3 describe('PaymentComponent', () => {
4   let component: PaymentComponent;
5   let fixture: ComponentFixture<PaymentComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ PaymentComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(PaymentComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\payment\payment.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { PackService } from '../pack.service';
3 @Component({
4   selector: 'app-payment',
5   templateUrl: './payment.component.html'
6 })
7 export class PaymentComponent implements OnInit {
8   packname=localStorage.getItem("packna");
9   constructor( public pac: PackService) { }
10  ngOnInit(): void {
11  }
12 }
```

src\app\profile\profile.component.html

```
1 <div class="container" style="font-family: 'Big Shoulders Stencil
Display', cursive;
2 font-family: 'Londrina Solid', cursive; background-color: #c5c6c7; max-
height:300vh; border-radius: 12px;">
3     <div class="d-grid gap-2 d-md-flex justify-content-md-center">
4         <button class="btn btn-info me-md-2" type="button" routerLink="/"
style="border:solid #1F2833;background-color:pink; margin: 1vw;">Home
5     </button>
6         <button class="btn btn me-md-2" style=" margin: 1vw; color: pink;
background-color:#1F2833 ;">
7             <h3>Profile</h3>
8         </button>
9         <button class="btn btn-info me-md-2" type="button" routerLink="/"
packages"
10             style="border:solid #1F2833;background-color: pink; margin:
1vw;">Package
11     </button>
12 </div>
13 <hr style="height: 2px; margin: 1%;padding: 0; ">
14 <div style="padding-bottom: 2vh;">
15     <div style="display: flex; flex-flow: row wrap;">
16         <div class="card mb-3 shadow-lg"
17             style="width: 18rem; margin: 0vw 1vw 1vw 1vw; background-
color: #c5c6c7; border-radius: 12px;">
18             <div style="background-color:rgb(255, 255, 255); margin:
1vh; border-radius: 12px;">
19                 
21             </div>
22             <div class="card-body " *ngIf="userdata">
23                 <h5 class="card-title">Name : {{userdata.Fname}}
{{userdata.lname}}</h5>
24                 <hr>
25                 <p class="card-text">Age: {{userdata.userage}}<br>
26                     Gender: {{userdata.gender}} <br>
27                     Expertise level : {{userdata.level}}</p>
28                 <div class="d-grid gap-2 col-6 mx-auto" style="margin:
1vh;">
29                     <button class="btn btn-info" type="button"
routerLink="/packages"
30                         style="border:solid #1F2833;background-
color:pink">Buy
31                     Package</button>
32                 </div>
33             </div>
34             <div class="small"><a routerLink="/contact" style="color:
#1F2833; display: flex; justify-content: center; padding-bottom: 1vh;">Need
Help?</a>
35             </div>
36         </div>
37     <div style="display: flex; flex-direction: column;" >
38         <div class="card " style="background-color: #c5c6c7; max-
width: 120vh; border-radius: 12px;">
39             <div>
40                 
43         </div>
44         <div class="card-body shadow-lg" style="background-
color: #c5c6c7; border-radius: 12px;">
45             <h4 class="card-title"
*ngIf="userdata.pack">Package: {{userdata.pack}}-{{userdata.level}}</h4> <h4
class="card-title" *ngIf="!userdata.pack">Package: 7days Free Trial</h4>
46             <hr>
47             <div *ngIf="userdata">
48                 <span>Adds on:</span>&nbsp;<span
*ngIf="addbath()"></span> &nbsp;<span
50                     <span *ngIf="addtrain()"></span> &nbsp;<span
52                     <span *ngIf="adddiet()"></span>
54                     <hr>
55                     <p class="card-text">Purchased on: {{userdata.date
|slice :0:10 }} <br></p>
56                     </div>
57                     <p class="card-text"><small class="text-
muted">Yoga fitness center</small></p>
58                     </div>
59                 </div>
60                 <div class="container-fluid"
61                     style="display: flex; flex-wrap: wrap; justify-
content: center; background-color: #c5c6c7; border-radius: 12px; padding:
0%; margin-top: 0.5vw; ">
62                     <iframe max-width="1vh" style="margin:0.1vh; border-
radius: 12px;"
63                         src="https://www.youtube.com/embed/FdyhENXyIQ4"
title="YouTube video player" frameborder="0"
64                         allow="accelerometer; autoplay; clipboard-write;
encrypted-media; gyroscope; picture-in-picture"
65                         allowfullscreen></iframe>
66                     <iframe max-width="1vh" style="margin:0.1vh; border-
radius: 12px;"
67                         src="https://www.youtube.com/embed/brjAjq4zEIE"
title="YouTube video player" frameborder="0"
68                         allow="accelerometer; ?autoplay=1; clipboard-
write; encrypted-media; gyroscope; picture-in-picture"
69                         allowfullscreen></iframe>
70                     <iframe max-width="1vh" style="margin:0.1vh; border-
radius: 12px;"
71                         src="https://www.youtube.com/embed/cMfChJLqma4"
title="YouTube video player" frameborder="0"
72                         allow="accelerometer; autoplay; clipboard-write;
encrypted-media; gyroscope; picture-in-picture"
73                         allowfullscreen></iframe>
74                     </div>
75                 </div>
76             </div>
77         </div>
78 </div>

```

src\app\profile\profile.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { ProfileComponent } from '../profile.component';
3 describe('ProfileComponent', () => {
4   let component: ProfileComponent;
5   let fixture: ComponentFixture<ProfileComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ ProfileComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(ProfileComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```


src\app\profile\profile.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { Router } from '@angular/router';
3 import { PackService } from '../pack.service';
4 import { UsersService } from '../users.service';
5 import { ActivatedRoute, Params } from '@angular/router';
6 @Component({
7   selector: 'app-profile',
8   templateUrl: './profile.component.html'
9 })
10 export class ProfileComponent implements OnInit {
11   msg: string;
12   userdata: { _id: number, Fname: string, lname: string, uemail: string,
13   usage: Number, usergender: Number, gender: String, userpass: string,
14   userphone: number, level: string, pack: string, istrain: boolean, isdietplan:
15   boolean, isbath: boolean; date: string; }
16   useriid: any = localStorage.getItem("loggeduser");
17   constructor(public userSer: UsersService, public activeRoute:
18   ActivatedRoute, public pac: PackService, public myRouter: Router) {
19   }
20   ngOnInit(): void {
21     console.log(this.useriid);
22     this.userSer.getloggedUserData(this.useriid).subscribe((data: any[])
23     => {
24       console.log(data);
25       this.userdata = data[0];
26     }, (error: any) => {
27       console.log(error);
28     }));
29   }
30   addbath()
31   {
32     if(this.userdata.isbath==true){
33       return true
34     }
35     else{
36       return false
37     }
38   }
39   addtrain()
40   {
41     if(this.userdata.istrain==true){
42       return true
43     }
44     else{
45       return false
46     }
47   }
48   adddiet()
49   {
50     if(this.userdata.isdietplan==true){
51       return true
52     }
53     else{
54       return false
55     }
56   }
57 }
```

src\app\slide\slide.component.html

```
1 <div id="carouselExampleCaptions" class="carousel slide" data-bs-  
ride="carousel">  
2     <div class="carousel-indicators">  
3         <button type="button" data-bs-target="#carouselExampleCaptions" data-  
bs-slide-to="0" class="active" aria-current="true" aria-label="Slide 1"></  
button>  
4         <button type="button" data-bs-target="#carouselExampleCaptions" data-  
bs-slide-to="1" aria-label="Slide 2"></button>  
5         <button type="button" data-bs-target="#carouselExampleCaptions" data-  
bs-slide-to="2" aria-label="Slide 3"></button>  
6     </div>  
7     <div class="carousel-inner">  
8         <div class="carousel-item active">  
9               
10             <div class="carousel-caption d-none d-md-block">  
11                 <h1 style="color: rgb(255, 243, 246); font-family: 'Big  
Shoulders Stencil Display', cursive;  
12                     font-family: 'Londrina Solid', cursive; margin-bottom: 20%;">  
Welcome to <br> <span style="font-family: 'Big Shoulders Stencil Display',  
cursive;">Yoga fitness center</span></h1>  
13             </div>  
14         </div>  
15         <div class="carousel-item">  
16               
17             <div class="carousel-caption d-none d-md-block">  
18                 <h2 style="color: pink; background-color: #1F2833; margin-left:  
14vw; border-radius: 5px; margin-right: 14vw; font-family: 'Big Shoulders  
Stencil Display', cursive;  
19                     font-family: 'Londrina Solid', cursive;">THE BEST FITNESS CENTER  
ON THE PLANET</h2>  
20                 <p style="color: #1F2833;background-color: pink; border-radius:  
5px;"><strong>  
21                     We believe in providing a high-quality fitness experience at  
an affordable cost. Read more about our membership options to decide which  
one works best for you.</strong></p>  
22             </div>  
23         </div>  
24         <div class="carousel-item">  
25               
26             <div class="carousel-caption d-none d-md-block">  
27                 <h2 style="color: pink; font-family: 'Big Shoulders Stencil  
Display', cursive;  
28                     font-family: 'Londrina Solid', cursive; background-color:  
#1F2833; margin-left: 14vw; border-radius: 5px; margin-right: 14vw;">YOUR  
FITNESS IS ESSENTIAL ™</h2>  
29                 <p style="color: #1F2833;background-color: pink; border-radius:  
5px;"><strong>Making your overall physical and mental wellness a priority!  
will melt away stress, decrease anxiety, and increase your overall positive  
energy levels, and you'll feel healthy and strong throughout the year!</  
strong></p>  
30             </div>  
31         </div>  
32     </div>  
33     <button class="carousel-control-prev" type="button" data-bs-  
target="#carouselExampleCaptions" data-bs-slide="prev">  
34         <span class="carousel-control-prev-icon" aria-hidden="true"></span>  
35         <span class="visually-hidden">Previous</span>
```

```

36     </button>
37     <button class="carousel-control-next" type="button" data-bs-
target="#carouselExampleCaptions" data-bs-slide="next">
38         <span class="carousel-control-next-icon" aria-hidden="true"></span>
39         <span class="visually-hidden">Next</span>
40     </button>
41 </div>
42 <div style="z-index:0;" *ngIf="!userSer.isLoggedIn()">
43     <marquee direction="left" style="background-color: #1F2833 ; margin:
0%; padding: 0%;" >
44         <p style="color:goldenrod; border-radius: 5px;"><strong>REGISTER
NOW AND GET 7 DAYS FREE TRIAL !</strong></p>
45     </marquee>
46 </div>
47 <div class=" p-5 rounded" style="background-color: #1F2833;">
48     <div class="col-sm-8 mx-auto" style="background-color: #1F2833">
49         <h1 style="color: pink; font-family: 'Big Shoulders Stencil
Display', cursive;
50         font-family: 'Londrina Solid', cursive;">ABOUT Yoga fitness center</
h1>
51         <p style="color: #c5c6c7;">
52             Whether you're a first-time user or a veteran member, we're here
to provide a environment in which anyone - and everyone - can be comfortable.
53             No matter what you're looking for in a fitness center, we've got
a membership option made for you. All Planet Fitness members enjoy unlimited
access to their home club and the support of our friendly, knowledgeable
staff anytime you need it.</p>
54         <p>
55             <a class="btn btn-info" routerLink="/login" role="button"
style="background-color: pink;
56             border-bottom-color: pink !important;
57             border-right-color: pink !important;
58             border-top-color: pink !important;
59             border-left-color: pink !important;">Lets Go &raquo;</a>
60         </p>
61     </div>
62 </div>
63 <div style="background-color: #1F2833;" class="abus">
64     <!--image-->
65     <div class="card-group" style="background-color: #1F2833; margin:
1vw;" >
66         <div class="card" style="background-color: #1F2833; margin: 1vh;
border-radius: 12px;">
67             
68             <div class="card-body" style="background-color: #1F2833;">
69                 <h5 class="card-title" style="font-family: 'Big Shoulders
Stencil Display', cursive;
70                 font-family: 'Londrina Solid', cursive; color: pink; display:
flex; justify-content: center;"> Perfect for women </h5>
71                 <p class="card-text" style="color: #c5c6c7; display: flex;
justify-content: center;">We have both unisex and women only facilities with
female coaches.</p>
72             </div>
73         </div>
74         <div class="card" style="background-color: #1F2833; margin: 1vh;
border-radius: 12px;">
75             
76             <div class="card-body" style="background-color: #1F2833;">
77                 <h5 class="card-title" style="font-family: 'Big Shoulders

```

```

Stencil Display', cursive;
78         font-family: 'Londrina Solid', cursive; color: pink; display:
flex; justify-content: center;">Outdoor Programs</h5>
79         <p class="card-text" style="color: #c5c6c7; display: flex;
justify-content: center;">We conduct outdoor programs every week , at free of
cost.</p>
80     </div>
81 </div>
82     <div class="card" style="background-color: #1F2833; margin: 1vh;
border-radius: 12px; ">
83         
84         <div class="card-body" style="background-color: #1F2833; border-
radius: 12px;">
85             <h5 class="card-title" style="font-family: 'Big Shoulders
Stencil Display', cursive;
86                 font-family: 'Londrina Solid', cursive; color: pink; display:
flex; justify-content: center;">Personal Training Program</h5>
87                 <p class="card-text" style="color: #c5c6c7; display: flex;
justify-content: center;">We have best coaches from each field to train you
the best .</p>
88             </div>
89         </div>
90     </div>
91     <!-->
92 </div>

```

src\app\slide\slide.component.spec.ts

```
1 import { ComponentFixture, TestBed } from '@angular/core/testing';
2 import { SlideComponent } from './slide.component';
3 describe('SlideComponent', () => {
4   let component: SlideComponent;
5   let fixture: ComponentFixture<SlideComponent>;
6   beforeEach(async () => {
7     await TestBed.configureTestingModule({
8       declarations: [ SlideComponent ]
9     })
10    .compileComponents();
11  });
12  beforeEach(() => {
13    fixture = TestBed.createComponent(SlideComponent);
14    component = fixture.componentInstance;
15    fixture.detectChanges();
16  });
17  it('should create', () => {
18    expect(component).toBeTruthy();
19  });
20 });
```

src\app\slide\slide.component.ts

```
1 import { Component, OnInit } from '@angular/core';
2 import { UsersService } from '../users.service';
3 @Component({
4   selector: 'app-slide',
5   templateUrl: './slide.component.html'
6 })
7 export class SlideComponent implements OnInit {
8   constructor(public userSer : UsersService) { }
9   ngOnInit(): void {
10  }
11 }
```

src\app\users.service.spec.ts

```
1 import { TestBed } from '@angular/core/testing';
2 import { UsersService } from './users.service';
3 describe('UsersService', () => {
4   let service: UsersService;
5   beforeEach(() => {
6     TestBed.configureTestingModule({});
7     service = TestBed.inject(UsersService);
8   });
9   it('should be created', () => {
10     expect(service).toBeTruthy();
11   });
12 });
```

src\app\users.service.ts

```
1 import { HttpClient } from '@angular/common/http';
2 import { Injectable } from '@angular/core';
3 @Injectable({
4   providedIn: 'root'
5 })
6 export class UsersService {
7   ad:string= '1635362908540';
8   name:string="";
9   constructor(public http : HttpClient,) {
10   }
11   doUserRegistration(data:any)
12   {
13     return this.http.post<string>("http://localhost:3000/register", data);
14   }
15   doUserLogin(data:any)
16   {
17     return this.http.post<any[]>("http://localhost:3000/login", data);
18   }
19   isAdmin(){
20     return !!localStorage.getItem("loggedadmin");
21   }
22   isLoggedIn()
23   {
24     this.name=localStorage.getItem("loggedusername");
25     return !!localStorage.getItem("loggeduser");
26   }
27   getAllUsers()
28   {
29     return this.http.get<any[]>("http://localhost:3000/allusers");
30   }
31   getloggedUserData(userid:string)
32   {
33     return this.http.get<any[]>("http://localhost:3000/profile/"+userid);
34   }
35   uemailcheckAvail(uemail:string)
36   {
37     return this.http.get<any[]>("http://localhost:3000/
uemailcheck/"+uemail);
38   }
39   uuemailcheckAvail(uemail:string)
40   {
41     return this.http.get<any[]>("http://localhost:3000/
uuemailcheck/"+uemail);
42   }
43   getSingleUserData(userid:string)
44   {
45     return this.http.get<any[]>("http://localhost:3000/getuser/"+userid);
46   }
47   editsingleUserdata(data:any)
48   {
49     return this.http.put<string>("http://localhost:3000/updateuser/",data);
50   }
51   addpackages(data:any)
52   {
53     return this.http.put<string>("http://localhost:3000/cart/",data);
54   }
55   deleteUserData(userid:number)
56   {
```



```
57     return this.http.delete<string>("http://localhost:3000/
deleteuser/"+userid);
58   }
59   searchUsers(searchtxt:string)
60   {
61     return this.http.get<any[]>("http://localhost:3000/
searchuser/"+searchtxt);
62   }
63 }
```

src\environments\environment.prod.ts

```
1 export const environment = {  
2   production: true  
3 };
```

src\environments\environment.ts

```
1 // This file can be replaced during build by using the `fileReplacements`
array.
2 // `ng build` replaces `environment.ts` with `environment.prod.ts`.
3 // The list of file replacements can be found in `angular.json`.
4 export const environment = {
5   production: false
6 };
7 /*
8  * For easier debugging in development mode, you can import the following
file
9  * to ignore zone related error stack frames such as `zone.run`,
`zoneDelegate.invokeTask`.
10  *
11  * This import should be commented out in production mode because it will
have a negative impact
12  * on performance if an error is thrown.
13  */
14 // import 'zone.js/plugins/zone-error'; // Included with Angular CLI.
```

src\index.html

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <meta charset="utf-8">
5   <title>YogaFitnessCenter</title>
6   <base href="/">
7   <meta name="viewport" content="width=device-width, initial-scale=1">
8   <link rel="icon" type="image/x-icon" href="favicon.ico">
9   <link rel="preconnect" href="https://fonts.googleapis.com">
10  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
11  <link href="https://fonts.googleapis.com/css2?family=Big+Shoulders+Stenci
12  l+Display:wght@900&family=Londrina+Solid:wght@300;400&display=swap"
13  rel="stylesheet">
14 </head>
15 <body style="background-color: #1F2833;">
16   <app-root></app-root>
17 </body>
18 </html>
```

src/main.ts

```
1 import { enableProdMode } from '@angular/core';
2 import { platformBrowserDynamic } from '@angular/platform-browser-dynamic';
3 import { AppModule } from './app/app.module';
4 import { environment } from './environments/environment';
5 if (environment.production) {
6   enableProdMode();
7 }
8 platformBrowserDynamic().bootstrapModule(AppModule)
9   .catch(err => console.error(err));
```

src\polyfills.ts

```
1 /**
2  * This file includes polyfills needed by Angular and is loaded before the
app.
3  * You can add your own extra polyfills to this file.
4  *
5  * This file is divided into 2 sections:
6  * 1. Browser polyfills. These are applied before loading ZoneJS and are
sorted by browsers.
7  * 2. Application imports. Files imported after ZoneJS that should be
loaded before your main
8  *    file.
9  *
10 * The current setup is for so-called "evergreen" browsers; the last
versions of browsers that
11 * automatically update themselves. This includes Safari >= 10, Chrome >=
55 (including Opera),
12 * Edge >= 13 on the desktop, and iOS 10 and Chrome on mobile.
13 *
14 * Learn more in https://angular.io/guide/browser-support
15 */
16 /*****
*****
17 * BROWSER POLYFILLS
18 */
19 /**
20 * IE11 requires the following for NgClass support on SVG elements
21 */
22 // import 'classlist.js'; // Run `npm install --save classlist.js`.
23 /**
24 * Web Animations `@angular/platform-browser/animations`
25 * Only required if AnimationBuilder is used within the application and
using IE/Edge or Safari.
26 * Standard animation support in Angular DOES NOT require any polyfills
(as of Angular 6.0).
27 */
28 // import 'web-animations-js'; // Run `npm install --save web-animations-
js`.
29 /**
30 * By default, zone.js will patch all possible macroTask and DomEvents
31 * user can disable parts of macroTask/DomEvents patch by setting
following flags
32 * because those flags need to be set before `zone.js` being loaded, and
webpack
33 * will put import in the top of bundle, so user need to create a separate
file
34 * in this directory (for example: zone-flags.ts), and put the following
flags
35 * into that file, and then add the following code before importing
zone.js.
36 * import './zone-flags';
37 *
38 * The flags allowed in zone-flags.ts are listed here.
39 *
40 * The following flags will work for all browsers.
41 *
42 * (window as any).__Zone_disable_requestAnimationFrame = true; // disable
patch requestAnimationFrame
43 * (window as any).__Zone_disable_on_property = true; // disable patch
```

```
onProperty such as onclick
44 * (window as any).__zone_symbol__UNPATCHED_EVENTS = ['scroll',
'mousemove']; // disable patch specified eventNames
45 *
46 * in IE/Edge developer tools, the addEventListener will also be wrapped
by zone.js
47 * with the following flag, it will bypass `zone.js` patch for IE/Edge
48 *
49 * (window as any).__Zone_enable_cross_context_check = true;
50 *
51 */
52 /*****
*****
53 * Zone JS is required by default for Angular itself.
54 */
55 import 'zone.js'; // Included with Angular CLI.
56 /*****
*****
57 * APPLICATION IMPORTS
58 */
```

src\styles.css

```
1 /* You can add global styles to this file, and also import other style
files */
2 .html{
3     background-color: #1F2833;
4     font-family: 'Big Shoulders Stencil Display', cursive;
5 font-family: 'Londrina Solid', cursive;
6 }
```


src/test.ts

```
1 // This file is required by karma.conf.js and loads recursively all
the .spec and framework files
2 import 'zone.js/testing';
3 import { getTestBed } from '@angular/core/testing';
4 import {
5   BrowserDynamicTestingModule,
6   platformBrowserDynamicTesting
7 } from '@angular/platform-browser-dynamic/testing';
8 declare const require: {
9   context(path: string, deep?: boolean, filter?: RegExp): {
10     keys(): string[];
11     <T>(id: string): T;
12   };
13 };
14 // First, initialize the Angular testing environment.
15 getTestBed().initTestEnvironment(
16   BrowserDynamicTestingModule,
17   platformBrowserDynamicTesting(),
18   { teardown: { destroyAfterEach: true } },
19 );
20 // Then we find all the tests.
21 const context = require.context('./', true, /\.spec\.ts$/);
22 // And load the modules.
23 context.keys().map(context);
```

tsconfig.app.json

```
1 /* To learn more about this file see: https://angular.io/config/tsconfig.  
*/  
2 {  
3   "extends": "../tsconfig.json",  
4   "compilerOptions": {  
5     "outDir": "../out-tsc/app",  
6     "types": []  
7   },  
8   "files": [  
9     "src/main.ts",  
10    "src/polyfills.ts"  
11  ],  
12  "include": [  
13    "src/**/*.d.ts"  
14  ]  
15 }
```

tsconfig.json

```
1 /* To learn more about this file see: https://angular.io/config/tsconfig.
*/
2 {
3   "compileOnSave": false,
4   "compilerOptions": {
5     "baseUrl": "./",
6     "outDir": "./dist/out-tsc",
7     "forceConsistentCasingInFileNames": true,
8     "strict": true,
9     "strictNullChecks": false,
10    "strictPropertyInitialization": false,
11    "noImplicitReturns": true,
12    "noFallthroughCasesInSwitch": true,
13    "sourceMap": true,
14    "declaration": false,
15    "downlevelIteration": true,
16    "experimentalDecorators": true,
17    "moduleResolution": "node",
18    "importHelpers": true,
19    "target": "es2017",
20    "module": "es2020",
21    "lib": [
22      "es2018",
23      "dom"
24    ]
25  },
26  "angularCompilerOptions": {
27    "enableI18nLegacyMessageIdFormat": false,
28    "strictInjectionParameters": true,
29    "strictInputAccessModifiers": true,
30    "strictTemplates": true
31  }
32 }
```

tsconfig.spec.json

```
1 /* To learn more about this file see: https://angular.io/config/tsconfig.  
*/  
2 {  
3   "extends": "../tsconfig.json",  
4   "compilerOptions": {  
5     "outDir": "../out-tsc/spec",  
6     "types": [  
7       "jasmine"  
8     ]  
9   },  
10  "files": [  
11    "src/test.ts",  
12    "src/polyfills.ts"  
13  ],  
14  "include": [  
15    "src/**/*.spec.ts",  
16    "src/**/*.d.ts"  
17  ]  
18 }
```