WEB AND MOBILE PROGRAMMING

COURSE COORDINATOR: Dr.yugyung lee

Instructor Name: Yeruva, Vijaya Kumari

TA's :- Khan, Muhammad Zubair and Alanazi, Ahmed

STUDENT:DAMODAR REDDY KOTTAM

EMAILID: dkzgg@umsystem.edu

Video link: https://youtu.be/Mr7RKh7IoX4

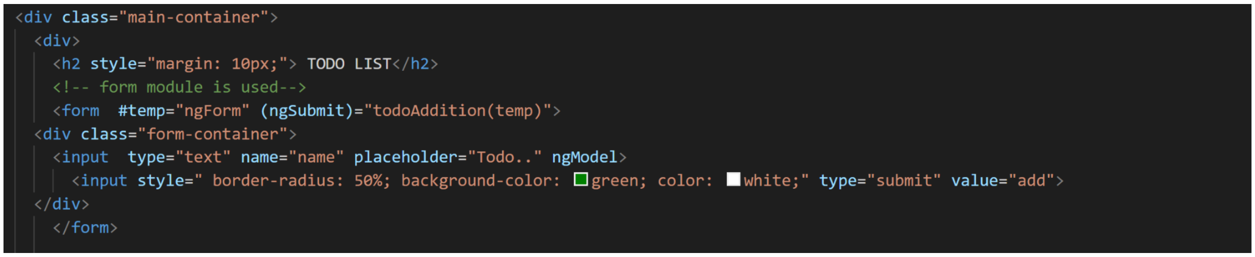
Source: https://github.com/damodarreddy3/web-ICP5

TASK - 1: To-Do Application

For the TODO application, FormModule is used to get the info from the user and two-way data binding is used to manipulate the data i.e; CRUD Operations.

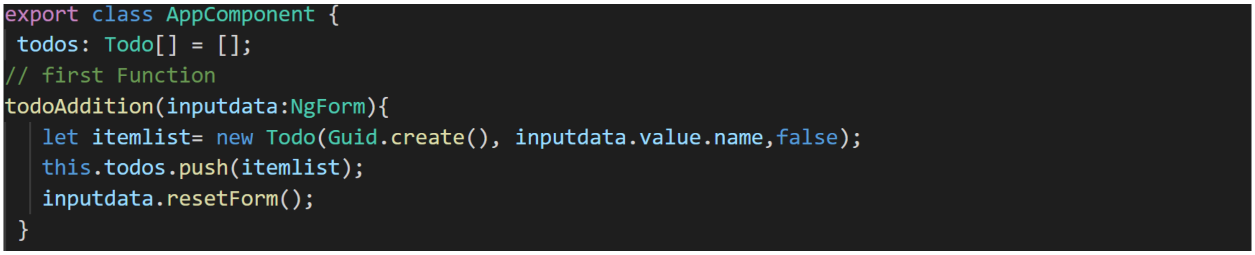
In this Application Grid is being used for the object to be created and when the user clicks on submit button the respective function is executed and stored in the todo list and here create() method is used to create the id of the grid.
After defining the list format it is pushed to the defined list and form is being reset for new entries from the user.

HTML:



```html
<div class="main-container">
  <div>
    <h2 style="margin: 10px;"> TODO LIST</h2>
    <!-- form module is used-->
    <form  #temp="ngForm" (ngSubmit)="todoAddition(temp)">
  <div class="form-container">
    <input  type="text" name="name" placeholder="Todo.." ngModel>
      <input style=" border-radius: 50%; background-color: green; color: white;" type="submit" value="add">
  </div>
    </form>
```

TS:



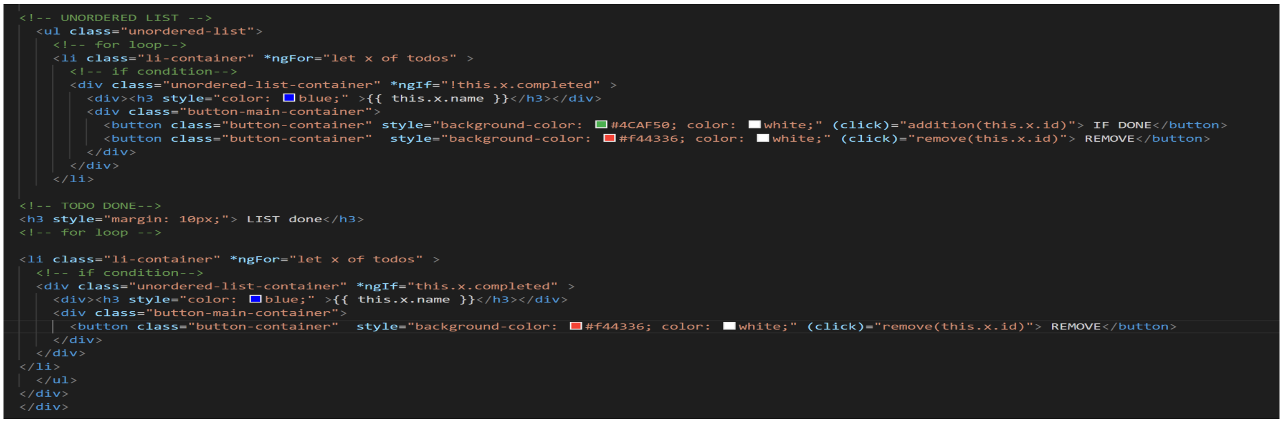
```
export class AppComponent {
 todos: Todo[] = [];
// first Function
todoAddition(inputdata:NgForm){
   let itemlist= new Todo(Guid.create(), inputdata.value.name,false);
   this.todos.push(itemlist);
   inputdata.resetForm();
 }
```

All the user entered todo lists are displayed using an unordered list and two buttons are being used to crud operations. Data is retrieved using annotation “{{}}”

Button1: it is the check button which has the functionality whether used completed the task or not.
When the user clicks on a button, the respective function in ts file is being executed and initially it checks for the id if it matches then setting the boolean function to true in html if condition is used to check the boolean. If boolean is true then the particular list of data is sent to an unordered list that satisfies the condition.

Button 2: It is used to delete the list of data, When the user clicks on the button, the respective function in ts file is being executed and initially it checks for the position and using the position splice is used to remove the data from the list.

HTML:

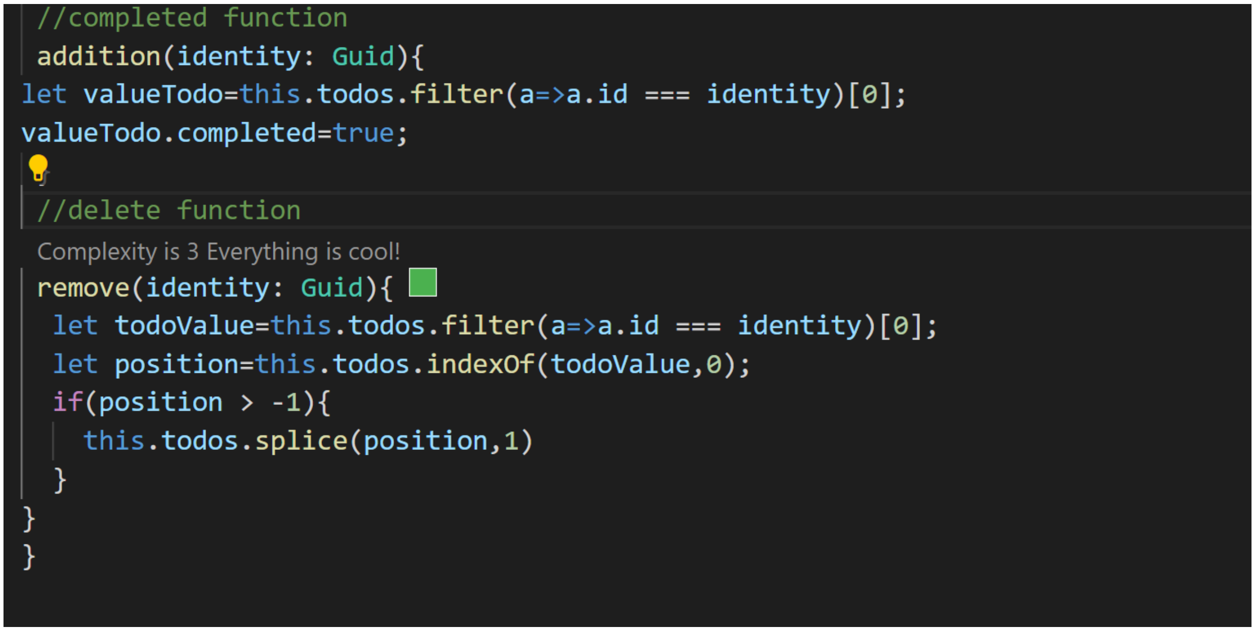


```
<!-- UNORDERED LIST -->
  <ul class="unordered-list">
    <!-- for loop-->
    <li class="li-container" *ngFor="let x of todos" >
      <!-- if condition-->
      <div class="unordered-list-container" *ngIf="!this.x.completed" >
        <div><h3 style="color: blue;" >{{ this.x.name }}</h3></div>
        <div class="button-main-container">
          <button class="button-container" style="background-color: #4CAF50; color: white;" (click)="addition(this.x.id)"> IF DONE</button>
          <button class="button-container"  style="background-color: #f44336; color: white;" (click)="remove(this.x.id)"> REMOVE</button>
        </div>
      </div>
    </li>

<!-- TODO DONE-->
<h3 style="margin: 10px;"> LIST done</h3>
<!-- for loop -->

<li class="li-container" *ngFor="let x of todos" >
  <!-- if condition-->
  <div class="unordered-list-container" *ngIf="this.x.completed" >
    <div><h3 style="color: blue;" >{{ this.x.name }}</h3></div>
    <div class="button-main-container">
      <button class="button-container"  style="background-color: #f44336; color: white;" (click)="remove(this.x.id)"> REMOVE</button>
    </div>
  </div>
</li>
  </ul>
</div>
</div>
```

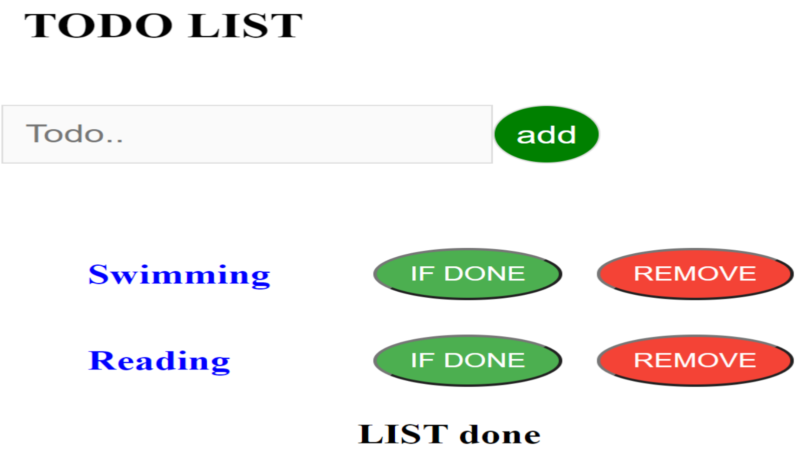
TS:

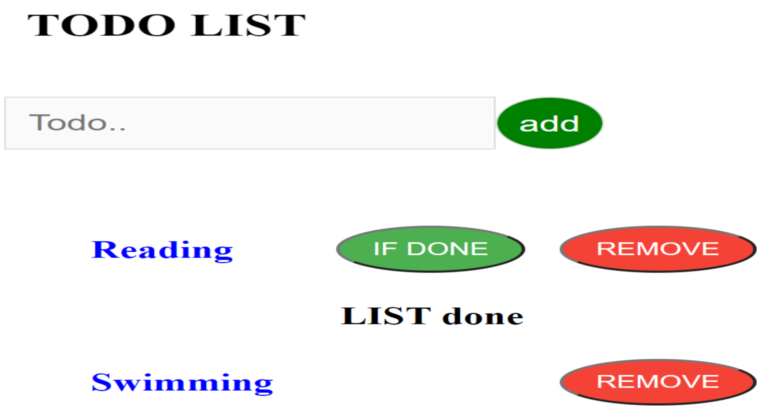


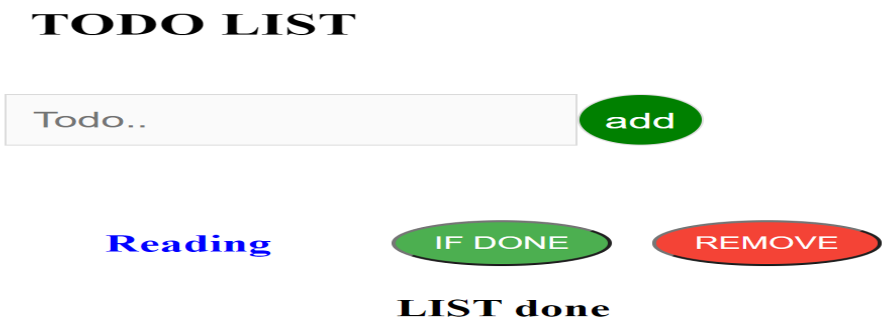
```
 //completed function
 addition(identity: Guid){
let valueTodo=this.todos.filter(a=>a.id === identity)[0];
valueTodo.completed=true;

 //delete function
 Complexity is 3 Everything is cool!
 remove(identity: Guid){
  let todoValue=this.todos.filter(a=>a.id === identity)[0];
  let position=this.todos.indexOf(todoValue,0);
  if(position > -1){
    this.todos.splice(position,1)
  }
}
}
```

OUTPUT:



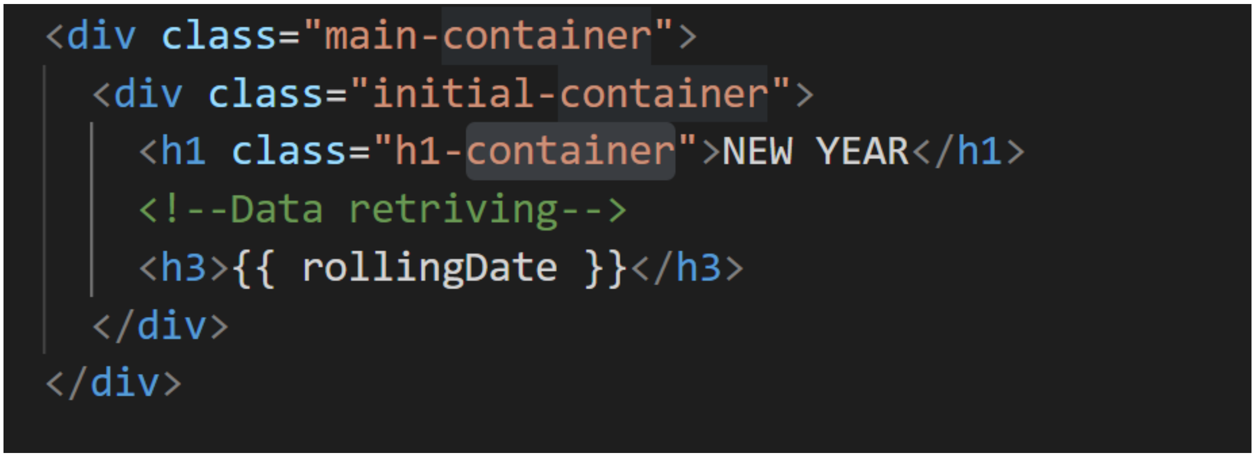

TODO LIST
Todo..
add
Reading
IF DONE
REMOVE
LIST done
Swimming
REMOVE


TODO LIST
Todo..
add
Reading
IF DONE
REMOVE
LIST done

TASK-2:

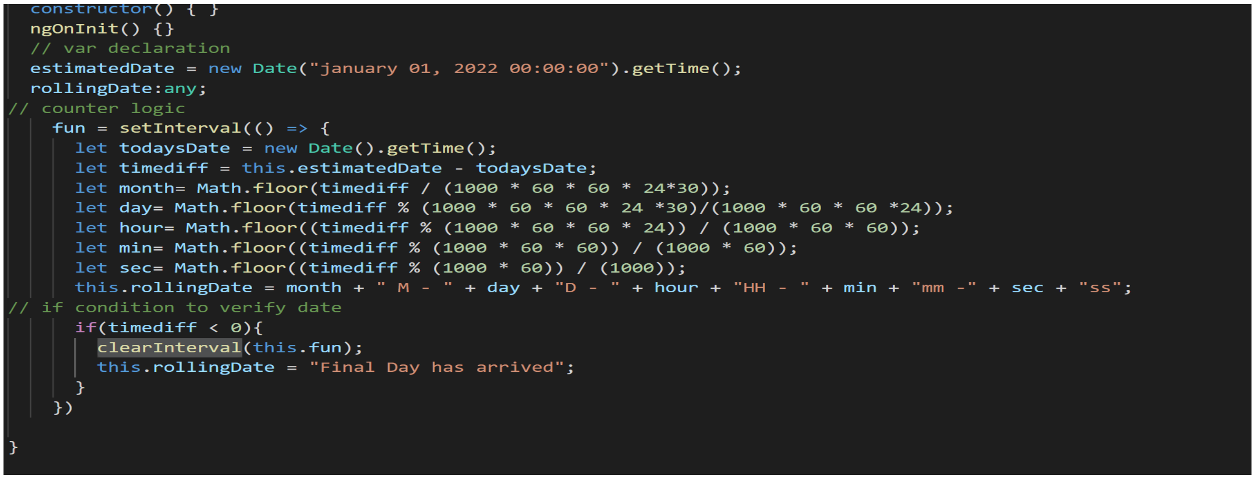
Counter is implemented by manipulating and calculating the dates using Math, here I have used new year as my final Data and calculated accordingly

HTML:



```html
<div class="main-container">
  <div class="initial-container">
    <h1 class="h1-container">NEW YEAR</h1>
    <!--Data retriving-->
    <h3>{{ rollingDate }}</h3>
  </div>
</div>
```

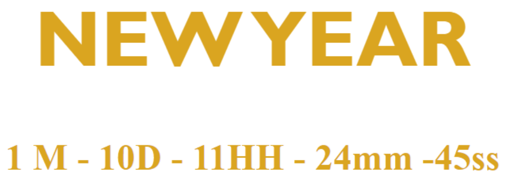
TS:



```typescript
  constructor() { }
  ngOnInit() {}
  // var declaration
  estimatedDate = new Date("january 01, 2022 00:00:00").getTime();
  rollingDate:any;
// counter logic
    fun = setInterval(() => {
      let todaysDate = new Date().getTime();
      let timediff = this.estimatedDate - todaysDate;
      let month= Math.floor(timediff / (1000 * 60 * 60 * 24*30));
      let day= Math.floor(timediff % (1000 * 60 * 60 * 24 *30)/(1000 * 60 * 60 *24));
      let hour= Math.floor((timediff % (1000 * 60 * 60 * 24)) / (1000 * 60 * 60));
      let min= Math.floor((timediff % (1000 * 60 * 60)) / (1000 * 60));
      let sec= Math.floor((timediff % (1000 * 60)) / (1000));
      this.rollingDate = month + " M - " + day + "D - " + hour + "HH - " + min + "mm -" + sec + "ss";
// if condition to verify date
      if(timediff < 0){
        clearInterval(this.fun);
        this.rollingDate = "Final Day has arrived";
      }
    })

}
```

OUTPUT:



```
NEWYEAR

1 M - 10D - 11HH - 24mm -45ss
```