



A screenshot of a web browser window. The address bar shows 'localhost:4200/view'. The page has two buttons: 'Add data' and 'Remove data'. Below them is a table with four columns: 'No.', 'Name', 'Weight', and 'Symbol'. The table contains 10 rows of data for elements from Hydrogen to Neon. The browser's taskbar at the bottom shows the Windows logo, a search bar, and various application icons. The system tray on the right indicates a temperature of 30°C, 'Partly cloudy' weather, and the date '06-10-2022'.

No.	Name	Weight	Symbol
1	Hydrogen	1.0079	H
2	Helium	4.0026	He
3	Lithium	6.941	Li
4	Beryllium	9.0122	Be
5	Boron	10.811	B
6	Carbon	12.0107	C
7	Nitrogen	14.0067	N
8	Oxygen	15.9994	O
9	Fluorine	18.9984	F
10	Neon	20.1797	Ne

A screenshot of a web browser window. The address bar shows 'localhost:4200'. The page title is 'My App'. The main content area displays a dark blue background with a starry night sky and silhouettes of trees at the bottom. Overlaid on this is a white registration form. The form has a progress bar at the top with six steps: '1 Fill out your name', '2 Fill out your address', '3 Your phone number', '4 Your mail', '5 Your password', and '6 Done'. The first step is active. Below the progress bar is a text input field labeled 'Name *'. At the bottom of the form is a 'Next' button. The browser's taskbar at the bottom shows the Windows logo, a search bar, and various application icons. The system tray on the right indicates a temperature of 30°C, 'Partly cloudy' weather, and the date '06-10-2022'.

WELCOME.COMPONENT.HTML

```
<mat-toolbar color="primary">

  <button mat-icon-button class="example-icon" aria-label="Example icon-button with menu
icon">

    <mat-icon>menu</mat-icon>

  </button>&nbsp;  

  <span>My App</span>

  <span class="example-spacer"></span>

  <button mat-icon-button class="example-icon favorite-icon" aria-label="Example icon-button
with heart icon" routerLink="/view">

    <mat-icon>login</mat-icon>

  </button>&nbsp;  

  <button mat-icon-button class="example-icon" aria-label="Example icon-button with share
icon" (click)="opendialog()">

    <mat-icon>mode_edit</mat-icon>

  </button>

</mat-toolbar>

<div

  class="bg-image d-flex justify-content-center align-items-center"

  style="

    background-image: url('https://mdbcdn.b-cdn.net/img/new/fluid/nature/015.webp');

    height: 100vh;

    "

>
```

HOME.COMPONENT.TS

```
import { Component, OnInit } from '@angular/core';

import { MatDialog, MatDialogConfig } from '@angular/material/dialog';

import { AddstudentComponent } from '../addstudent/addstudent.component';
```

```

export interface PeriodicElement {
    name: string;
    position: number;
    weight: number;
    symbol: string;
}

const ELEMENT_DATA: PeriodicElement[] = [
    {position: 1, name: 'Hydrogen', weight: 1.0079, symbol: 'H'},
    {position: 2, name: 'Helium', weight: 4.0026, symbol: 'He'},
    {position: 3, name: 'Lithium', weight: 6.941, symbol: 'Li'},
    {position: 4, name: 'Beryllium', weight: 9.0122, symbol: 'Be'},
    {position: 5, name: 'Boron', weight: 10.811, symbol: 'B'},
    {position: 6, name: 'Carbon', weight: 12.0107, symbol: 'C'},
    {position: 7, name: 'Nitrogen', weight: 14.0067, symbol: 'N'},
    {position: 8, name: 'Oxygen', weight: 15.9994, symbol: 'O'},
    {position: 9, name: 'Fluorine', weight: 18.9984, symbol: 'F'},
    {position: 10, name: 'Neon', weight: 20.1797, symbol: 'Ne'},
];

@Component({
    selector: 'app-welcome',
    templateUrl: './welcome.component.html',
    styleUrls: ['./welcome.component.css']
})

export class WelcomeComponent implements OnInit {
    displayedColumns: string[] = ['position', 'name', 'weight', 'symbol'];
    dataSource = ELEMENT_DATA;
    constructor(public dialog: MatDialog) { }

```

```

ngOnInit(): void {
}

opendialog(){
    const configDia=new MatDialogConfig();
    configDia.width="80%";
    configDia.autoFocus=true;
    configDia.disableClose=true;
    this.dialog.open(AddstudentComponent,configDia);
}
}

```

ADD.COMPONENT.HTML

```

<mat-stepper [linear]="isLinear" #stepper>
  <mat-step [stepControl]="firstFormGroup">
    <form [formGroup]="firstFormGroup">
      <ng-template matStepLabel>Fill out your name</ng-template>
      <mat-form-field appearance="fill">
        <mat-label>Name</mat-label>
        <input matInput placeholder="Last name, First name" formControlName="firstCtrl"
required>
      </mat-form-field>
      <div>
        <button mat-button matStepperNext>Next</button>
      </div>
    </form>
  </mat-step>
  <mat-step [stepControl]="secondFormGroup" label="Fill out your address">
    <form [formGroup]="secondFormGroup">
      <mat-form-field appearance="fill">
        <mat-label>Address</mat-label>

```

```

        <input matInput formControlName="secondCtrl" placeholder="Ex. 1 Main St, New York, NY"
              required>
    </mat-form-field>

    <div>

        <button mat-button matStepperPrevious>Back</button>

        <button mat-button matStepperNext>Next</button>

    </div>

</form>

</mat-step>

<mat-step [stepControl]="thirdFormGroup" label="Your phone number">

    <form [formGroup]="thirdFormGroup">

        <mat-form-field appearance="fill">

            <mat-label>phone number</mat-label>

            <input matInput formControlName="thirdCtrl" placeholder="Ex. 9098877647"
                  required>

        </mat-form-field>

        <div>

            <button mat-button matStepperPrevious>Back</button>

            <button mat-button matStepperNext>Next</button>

        </div>

    </form>

</mat-step>

<mat-step [stepControl]="fourthFormGroup" label="Your mail">

    <form [formGroup]="fourthFormGroup">

        <mat-form-field appearance="fill">

            <mat-label>mail id</mat-label>

            <input matInput formControlName="fourthCtrl" placeholder="Ex. abc@mail.com"
                  required>

        </mat-form-field>

```

```

    <div>

        <button mat-button matStepperPrevious>Back</button>

        <button mat-button matStepperNext>Next</button>

    </div>

</form>
</mat-step>
<mat-step [stepControl]="fifthFormGroup" label="Your password">
    <form [formGroup]="fifthFormGroup">
        <mat-form-field appearance="fill">
            <mat-label>password</mat-label>

            <input matInput formControlName="fifthCtrl" placeholder="Ex. Abc@123"
                required>

        </mat-form-field>

        <div>

            <button mat-button matStepperPrevious>Back</button>

            <button mat-button matStepperNext>Next</button>

        </div>

    </form>
</mat-step>
<mat-step>
    <ng-template matStepLabel>Done</ng-template>

    <p>You are now done.</p>

    <div>

        <button mat-button matStepperPrevious>Back</button>

        <button mat-button (click)="stepper.reset()">Reset</button>

    </div>

</mat-step>
</mat-stepper>

```

VIEW.COMPONENT.HTML

```
<div class="demo-button-container">

  <button mat-raised-button (click)="addData()" class="demo-button">

    Add data

  </button>

  <button

    mat-raised-button

    [disabled]="!dataSource.length"

    (click)="removeData()"

    class="demo-button">

    Remove data

  </button>

</div>


<table mat-table [dataSource]="dataSource" class="mat-elevation-z8 demo-table">

  <!-- Position Column -->

  <ng-container matColumnDef="position">

    <th mat-header-cell *matHeaderCellDef>No.</th>

    <td mat-cell *matCellDef="let element">{{element.position}}</td>

  </ng-container>


  <!-- Name Column -->

  <ng-container matColumnDef="name">

    <th mat-header-cell *matHeaderCellDef>Name</th>

    <td mat-cell *matCellDef="let element">{{element.name}}</td>

  </ng-container>


  <!-- Weight Column -->
```

```

<ng-container matColumnDef="weight">

  <th mat-header-cell *matHeaderCellDef>Weight</th>

  <td mat-cell *matCellDef="let element">{{element.weight}}</td>

</ng-container>

<!-- Symbol Column -->

<ng-container matColumnDef="symbol">

  <th mat-header-cell *matHeaderCellDef>Symbol</th>

  <td mat-cell *matCellDef="let element">{{element.symbol}}</td>

</ng-container>

<tr mat-header-row *matHeaderRowDef="displayedColumns"></tr>

<tr mat-row *matRowDef="let row; columns: displayedColumns;"></tr>

</table>

```

VIEW.COMPONENT.TS

```

import { Component, OnInit, ViewChild } from '@angular/core';

import { MatTable } from '@angular/material/table';

export interface PeriodicElement {

  name: string;

  position: number;

  weight: number;

  symbol: string;

}

const ELEMENT_DATA: PeriodicElement[] = [

  {position: 1, name: 'Hydrogen', weight: 1.0079, symbol: 'H'},

  {position: 2, name: 'Helium', weight: 4.0026, symbol: 'He'},

```



```

    {position: 3, name: 'Lithium', weight: 6.941, symbol: 'Li'},
    {position: 4, name: 'Beryllium', weight: 9.0122, symbol: 'Be'},
    {position: 5, name: 'Boron', weight: 10.811, symbol: 'B'},
    {position: 6, name: 'Carbon', weight: 12.0107, symbol: 'C'},
    {position: 7, name: 'Nitrogen', weight: 14.0067, symbol: 'N'},
    {position: 8, name: 'Oxygen', weight: 15.9994, symbol: 'O'},
    {position: 9, name: 'Fluorine', weight: 18.9984, symbol: 'F'},
    {position: 10, name: 'Neon', weight: 20.1797, symbol: 'Ne'},
  ];

  @Component({
    selector: 'app-viewstudent',
    templateUrl: './viewstudent.component.html',
    styleUrls: ['./viewstudent.component.css']
  })
  export class ViewstudentComponent implements OnInit {
    displayedColumns: string[] = ['position', 'name', 'weight', 'symbol'];
    dataSource = [...ELEMENT_DATA];

    @ViewChild(MatTable)
    table!: MatTable<PeriodicElement>;

    addData() {
      const randomElementIndex = Math.floor(Math.random() * ELEMENT_DATA.length);
      this.dataSource.push(ELEMENT_DATA[randomElementIndex]);
      this.table.renderRows();
    }

    removeData() {
      this.dataSource.pop();
      this.table.renderRows();
    }

    constructor() { }
  }

```

```
ngOnInit(): void {  
  }  
}
```