Explaining the project how to implemented step by step in short

1. First I will check the node is installed or not the command is node –version
2. Then I will check the angular is installed or not the command is ng –version
3. I was created a folder student record project under that project I was created one folder CRUD & I hit the command cd CRUD
4. Give the command on command prompt is to open the visual studio code is code.
5. Then I was added the angular material from the website https://material.angular.io/
6. To add the angular material command is ng add @angular/material
7. Going on that website angular material components
8. Click on toolbar & check the code & copy the code into the app.module.ts
9. Added the imports in app.module.ts file
10. Check the code is working or not
11. Go to the app.module.ts & added Imports MatButtonModule
12. Go to the toolbar overview & copy the css & paste it in the app.componnt.scss file
13. Go to the google & search the mat icon list in material Icons
14. Copy the name & paste it in the app.component.html under the <mat-icon></mat-icon> tag
15. Go to the dialog copy the code & paste it in the app.module.ts
16. Go to the typescript file & copy the dialog box code .
17. Create the new component give the command ng generate component dialog
18. Copy & paste the code app.component.ts
19. Import the API in app.module.ts file
20. Write a constructor in typescript file
21. Go to the Forms & Copy the code of Form Field appearance variants
22. Import the API to app.module.ts
23. Inside @NgModule add the declarations
24. Go to the Input import MatInputModule
25. Give the css style in dialog.component.scss file
26. Go to the Select copy the API & paste it in the app.module.ts file import MatSelectModule
27. Go to the Datepicker & copy the code & API & import the module in app.module.ts file

Explaining the project how to implemented step by step in short

28. Also import the MatNativeDateModule it is from @angular/core
29. Insert MatInput tittle
30. Edit the html code dialog.component.html file
31. Adding the Reactive Form
32. Create the form in the dialog.component.ts studentForm! :FormGroup
33. Write constructor in dialog.component.ts file
34. Bind the form in the dialog.component.html file
35. Import the ReactiveFormsModule
36. Add the inputs FormControlName="Same name as in the add student dialog form"
37. In dialog.component.html added the click event button (click)="addStudent"
38. Create the method in dialog.component.ts
39. For the check the code is working or not console.log(this.studentForm.value);
40. Right Click on UI Inspect Check the console window data was coming or not.
41. Create the Json Server
42. Install the Json Server using cmd on Terminal npm install –g json-server
43. Json-server –watch db.json
44. Visit the localhost
45. Db.json created
46. Change in the db.json file is "Posts" to "studentList"
47. Create the API service
48. Type the Command on terminal ng generate service services/api
49. Inject the HttpClient in api.service.ts & add the constructor(private http: Httpclient){}
50. Add import in app.module.ts is HttpClientModule
51. Add the method postStudent(data:any) in api.service.ts file
52. Also add the method getStudent(){} in api.service.ts file
53. Go to the dialog.component.ts file & add method addStudent(){}
54. Import MatDialogRef also add in the constructor
55. Inject the API in app.component.ts in constructor getAllStudents(){}
56. Search the examples add example of Data Table with Sorting, Pagination & Filtering copy the html code & paste it in the app.component.html

Explaining the project how to implemented step by step in short

57. Imports the MatPaginator, MatSort, MatTable, Datasource paste the code above the constructor displayColumns , dataSource, @viewChild
58. Added the Exclamatory mark to solve the errors of dataSource, Pagination & sort
59. Match the API names with Forms.
60. Added in the app.component.ts
61. ApplyFilter code in ts file
62. Import Paginator Module
63. Change the names in app.component.html
64. Databinding using string interpolation concept {{row.date}} for table data
65. Copy the css & paste it in app.component.css
66. Add the .container in css
67. Added the pipe for date
68. Added the action buttons means edit & delete
69. Import the MAT_DIALOG_DATA & inject it
70. Changes in dialog.component.ts
71. Create the API putStudent(){} data also add the API deleteStudent(){} in api.service.ts file
72. Add the method updateStudent(){} in dialog.component.ts
73. Create the method deleteStudent(){} in app.component.ts file
74. Go to html write click event
75. Hence Successfully Run the Project