**1 – Way**

Site Link : <https://medium.com/@madhavmahesh/exporting-an-excel-file-in-angular-927756ac9857>

npm install --save file-saver

npm install --save xlsx

First Create a Service : **ng g service excel**

Paste the following code

**excel.service.ts**

import { Injectable } from '@angular/core';

import \* as FileSaver from 'file-saver';

import \* as XLSX from 'xlsx';

const EXCEL\_TYPE = 'application/vnd.openxmlformats-officedocument.spreadsheetml.sheet;charset=UTF-8';

const EXCEL\_EXTENSION = '.xlsx';

@Injectable({

providedIn: 'root'

})

export class ExcelService {

constructor() { }

public exportAsExcelFile(json: any[], excelFileName: string): void {

const worksheet: XLSX.WorkSheet = XLSX.utils.json\_to\_sheet(json);

const workbook: XLSX.WorkBook = { Sheets: { 'data': worksheet }, SheetNames: ['data'] };

const excelBuffer: any = XLSX.write(workbook, { bookType: 'xlsx', type: 'array' });

this.saveAsExcelFile(excelBuffer, excelFileName);

}

private saveAsExcelFile(buffer: any, fileName: string): void {

const data: Blob = new Blob([buffer], { type: EXCEL\_TYPE });

FileSaver.saveAs(data, fileName + '\_export\_' + new Date().getTime() + EXCEL\_EXTENSION);

}

}

**excel.component.ts**

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

import { ExcelService } from '../services/excel.service';

@Component({

selector: 'app-excel’,

templateUrl: './excel.component.html',

styleUrls: ['./ excel.component.css']

})

export class ExcelComponent implements OnInit {

constructor(private excelService: ExcelService) { }

data: any = [

{ eid: 'e101', ename: 'ravi', esal: 1000 },

{ eid: 'e102', ename: 'ram', esal: 2000 },

{ eid: 'e103', ename: 'rajesh', esal: 3000 }

];

exportAsXLSX(): void {

this.excelService.exportAsExcelFile(this.data, 'sample');

}

ngOnInit() {

}

}

**excel.component.html**

<p>Exporting an excel file of type .xslx !!!</p>

<p>Export as XLSX by clicking the below button</p>

<button (click)="exportAsXLSX()">

Export to Excel

</button>

<table>

<tr>

<td>Eid</td>

<td>Ename</td>

<td>Esal</td>

</tr>

<tr \*ngFor="let item of data">

<td>{{item.eid}}</td>

<td>{{item.ename}}</td>

<td>{{item.esal}}</td>

</tr>

</table>

**2 – Way**

Site Link : <https://www.npmjs.com/package/angular5-csv>

npm install --save angular5-csv

**excel.component.ts**

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

import { Angular5Csv } from 'angular5-csv/Angular5-csv';

@Component({

selector: 'app-excel’,

templateUrl: './excel.component.html',

styleUrls: ['./ excel.component.css']

})

export class ExcelComponent implements OnInit {

constructor() { }

data = [

{ name: "Test 1", age: 13, average: 8.2, approved: true, description: "using 'Content here, content here' " },

{ name: 'Test 2', age: 11, average: 8.2, approved: true, description: "using 'Content here, content here' " },

{ name: 'Test 4', age: 10, average: 8.2, approved: true, description: "using 'Content here, content here' " },

];

exportExcel() {

var options = {

headers: ["Name", "Age", "Average", "Approved", "Description"]

};

let cc = new Angular5Csv(this.data, 'My Report', options);

}

ngOnInit() {

this.exportExcel();

}

}

## API | ****Angular5Csv(data, filename, options)****

| **Option** | **Default** | **Description** |
| --- | --- | --- |
| **fieldSeparator** | , | Defines the field separator character |
| **quoteStrings** | " | If provided, will use this characters to "escape" fields, otherwise will use double quotes as deafult |
| **decimalseparator** | . | Defines the decimal separator character (default is .). If set to "locale", it uses the [language sensitive representation of the number](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Number/toLocaleString). |
| **showLabels** | false | If provided, would use this attribute to create a header row |
| **showTitle** | false |  |
| **useBom** | true | If true, adds a BOM character at the start of the CSV |
| **noDownload** | false | If true, disables automatic download and returns only formatted CSV |

## Options Example

  var options **=** {

    fieldSeparator**:** ',',

    quoteStrings**:** '"',

    decimalseparator**:** '.',

    showLabels**:** true,

    showTitle**:** true,

    useBom**:** true,

    noDownload**:** true,

    headers**:** ["First Name", "Last Name", "ID"]

  };

  Angular5Csv(data, filename, options);