

How to import Angular Material in project?

Asked 2 years, 1 month ago Active 3 months ago Viewed 58k times



I have installed Angular Material Design. Now I try to add this in `app.module.ts` file:

31

```
import { MaterialModule } from '@angular/material';
```



What I should decify in section: `imports: []` ? that to load all material entities.



I tried: `imports: ['MaterialModule']` but it is deprecated

16



edited Jul 23 '18 at 21:45



Frederik Struck-Schønning

10.2k 7 49 56

asked Jul 18 '17 at 12:29



OPV

2,016 5 20 59

material.angular.io/guide/getting-started – Jota.Toledo Jul 18 '17 at 12:31

I did that, as you can see there is an sample with `import { MaterialModule} from '@angular/material` here – OPV Jul 18 '17 at 12:33

I dont see that in the page – Jota.Toledo Jul 18 '17 at 12:34 ✎

Yes, therfore I ask how to import all modules – OPV Jul 18 '17 at 12:35

Ill add an answer, I think you might be looking at an old, cached documentation page or similar... – Jota.Toledo Jul 18 '17 at 12:37

5 Answers



`MaterialModule` was depreciated in version 2.0.0-beta.3 and it has been removed completely in version 2.0.0-beta.11. See this

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



- Angular Material now requires Angular 4.4.3 or greater
- **MaterialModule** has been removed.
- For beta.11, we've made the decision to **deprecate the "md"** prefix completely and **use "mat"** moving forward.

Please go through [CHANGELOG](#) we will get more answer!

Example shown below cmd

```
npm install --save @angular/material @angular/animations @angular/cdk
npm install --save angular/material2-builds angular/cdk-builds
```

Create file (material.module.ts) inside the 'app' folder

```
import { NgModule } from '@angular/core';

import {
  MatButtonModule,
  MatMenuModule,
  MatToolbarModule,
  MatIconModule,
  MatCardModule
} from '@angular/material';

@NgModule({
  imports: [
    MatButtonModule,
    MatMenuModule,
    MatToolbarModule,
    MatIconModule,
    MatCardModule
  ],
  exports: [
    MatButtonModule,
    MatMenuModule,
    MatToolbarModule,
    MatIconModule,
    MatCardModule
  ]
})
export class MaterialModule {}
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```
import { MaterialModule } from './material.module';
```

Your component html file

```
<div>
  <mat-toolbar color="primary">
    <span><mat-icon>mood</mat-icon></span>

    <span>Yay, Material in Angular 2!</span>

    <button mat-icon-button [mat-menu-trigger-for]="menu">
      <mat-icon>more_vert</mat-icon>
    </button>
  </mat-toolbar>
  <mat-menu x-position="before" #menu="matMenu">
    <button mat-menu-item>Option 1</button>
    <button mat-menu-item>Option 2</button>
  </mat-menu>

  <mat-card>
    <button mat-button>All</button>
    <button mat-raised-button>Of</button>
    <button mat-raised-button color="primary">The</button>
    <button mat-raised-button color="accent">Buttons</button>
  </mat-card>

  <span class="done">
    <button mat-fab>
      <mat-icon>check_circle</mat-icon>
    </button>
  </span>
</div>
```

Add global css 'style.css'

```
@import 'https://fonts.googleapis.com/icon?family=Material+Icons';
@import '~@angular/material/prebuilt-themes/indigo-pink.css';
```

Your component css

By using our site, you acknowledge that you have read and understand our Cookie Policy, Privacy Policy, and our Terms of Service.

```

}

mat-card {
  max-width: 80%;
  margin: 2em auto;
  text-align: center;
}

mat-toolbar-row {
  justify-content: space-between;
}

.done {
  position: fixed;
  bottom: 20px;
  right: 20px;
  color: white;
}

```

If any one didn't get output use below instruction

instead of above interface (material.module.ts) u can directly use below code also in the app.module.ts.

```

import { FormsModule, ReactiveFormsModule } from '@angular/forms';
import { BrowserAnimationsModule } from '@angular/platform-browser/animations';
import { MatButtonModule, MdCardModule, MdMenuModule, MdToolbarModule, MdIconModule,
MatAutocompleteModule, MatInputModule, MatFormFieldModule } from '@angular/material';

```

So this case u don't want to import

```

import { MaterialModule } from './material.module';

```

in the app.module.ts

edited Feb 18 at 16:24



Mike Gledhill

18.5k 4 107 122

answered Sep 29 '17 at 2:10



Rijo

1,230 11 26

- 1 Great Answer Rijo. & very timely indeed! Got a quick question that bugs me for some time. It's not a big deal but I'm just curious. See the `./` in the `import { MaterialModule } from './material.module';` above. I was told that `./` means the current directory. But why does it need it? Don't we assume

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

- 1 It would be really nice if there is some kind of automated generator / validator / cleanup for the material-import.module.ts. Ok, I understand how it works, and I have my "version" setup and copy it from project to project, and it has "filled up" over the time. But removing unused Modules (review by hand all templates) is not what I often do. So the new import has no advantage to the old / deprecated but complete MaterialModule ... – [mhoff](#) Dec 19 '17 at 11:15

If you want to import *all* Material modules, create your own module i.e. `material.module.ts` and do something like the following:

9

```
import { NgModule } from '@angular/core';
import * as MATERIAL_MODULES from '@angular/material';

export function mapMaterialModules() {
  return Object.keys(MATERIAL_MODULES).filter((k) => {
    let asset = MATERIAL_MODULES[k];
    return typeof asset == 'function'
      && asset.name.startsWith('Mat')
      && asset.name.includes('Module');
  }).map((k) => MATERIAL_MODULES[k]);
}
const modules = mapMaterialModules();

@NgModule({
  imports: modules,
  exports: modules
})
export class MaterialModule { }
```

[Run code snippet](#)[Expand snippet](#)

Then import the module into your `app.module.ts`

answered Feb 2 '18 at 16:04



[Taylor](#)

369 2 6

- 1 Unfortunately it doesn't work when compiling with `--prod` option – [devalone](#) Mar 30 '18 at 20:04

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).



The `MaterialModule` was deprecated in the **beta3** version with the goal that developers should only import into their applications what they are going to use and thus improve the bundle size.

6



The developers have now 2 options:

- Create a **custom** `MyMaterialModule` which imports/exports the components that your application requires and can be imported by other (feature) modules in your application.
- Import directly the individual material modules that a module requires into it.

Take the following as example (extracted from [material page](#))

First approach:

```
import {MdButtonModule, MdCheckboxModule} from '@angular/material';

@NgModule({
  imports: [MdButtonModule, MdCheckboxModule],
  exports: [MdButtonModule, MdCheckboxModule],
})
export class MyOwnCustomMaterialModule { }
```

Then you can import this module into any of yours.

Second approach:

```
import {MdButtonModule, MdCheckboxModule} from '@angular/material';

@NgModule({
  ...
  imports: [MdButtonModule, MdCheckboxModule],
  ...
})
export class PizzaPartyAppModule { }
```

Now you can use the respective material components in all the components **declared in** `PizzaPartyAppModule`.

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

- With the latest version of material, you need to import `BrowserAnimationsModule` into your main module if you want the animations to work
- With the latest version developers now need to add `@angular/cdk` to their `package.json` (material dependency)
- Import the material modules always after `BrowserModule` , as stated by the docs:

Whichever approach you use, be sure to import the Angular Material modules after Angular's BrowserModule, as the import order matters for NgModules.

edited Jul 23 '18 at 21:46



Frederik Struck-Schøning

10.2k 7 49 56

answered Jul 18 '17 at 12:46



Jota.Toledo

14.6k 7 34 53

How to install this with this webpack? github.com/AngularClass/angular-starter/blob/master/... – OPV Jul 19 '17 at 11:58

1 github.com/AngularClass/angular-starter/wiki/... – Jota.Toledo Jul 19 '17 at 12:04

Should not they have fixed this first before this breaking change? material.angular.io – Average Joe Oct 1 '17 at 15:56

Step 1

0

```
yarn add @angular/material @angular/cdk @angular/animations
```

Step 2 - Create a new file(/myApp/src/app/material.module.ts) that includes all the material UI modules (there is no shortcut, you have to include individual modules one by one)

```
import { NgModule } from '@angular/core';
```

```
import {  
    MatButtonModule,  
    MatMenuModule,  
    MatToolbarModule,  
    MatIconModule
```

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

```

@NgModule({
  imports: [
    MatButtonModule,
    MatMenuModule,
    MatToolbarModule,
    MatIconModule,
    MatCardModule
  ],
  exports: [
    MatButtonModule,
    MatMenuModule,
    MatToolbarModule,
    MatIconModule,
    MatCardModule
  ]
})
export class MaterialModule {}

```

Step 3 - Import and add that newly created module to your **app.module.ts**

```

import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { AppComponent } from './app.component';
import { MaterialModule } from './material.module'; // material module imported

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    MaterialModule // MATERIA module added
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }

```

answered Jan 17 at 21:09



sapy

3 703

3

32

41

By using our site, you acknowledge that you have read and understand our [Cookie Policy](#), [Privacy Policy](#), and our [Terms of Service](#).

[Click here to see Error message screenshot](#)

▲
-1
▼

If you people are getting this error "compiler.js:2430 Uncaught Error: Unexpected directive 'MatIcon' imported by the module 'AppModule'. Please add a @NgModule annotation"

Please do not import MatIcon from @angular/material.

Just Import below: **import { MatIconModule } from '@angular/material';**

How to import Angular Material?

You can run below command. **ng add @angular/material**

edited May 16 at 7:04

answered May 15 at 6:03



[Santhosh Raj Mudulakar](#)

15 4