Cheatography

Table Hitung Angular 6 Cheat Sheet

by amihapsari via cheatography.com/82859/cs/19675/

Angular CLI

Generating classes	ng g cl [class's name]
Serving commands	ng serve
	ng serve -o

my-app/src/app/app.componen-t.css

```
table{
    mar gin -left: auto;
    mar gin -right:
auto;
}
.tabel2 {
        bor der -co -
llapse: collapse;
   }

.tabel2, tr, th, td {
      border: 1px solid
black;
}
```

Button / Tombol

```
<button type="submit"
class="btn btn-primary"
(click)="onNew()">New</button>
```

Metode 1

Metode 1 (cont)

Retrieve Data	<td>{{ stc.kuantitas }}</td>

my-app/src/app/app.componen-t.html

```
<!--The content below
is only a placeholder
and can be replaced.-->
<div style= " tex t-a -
lig n:c ent er">
    <h1>
       CRUD {{ title }}
    </h 1>
    <h2 >Insert Data</ -
h2>
    <button type="s -
ubm it" class= "btn btn-
pr ima ry" (click )="o -
nNe w() " >Ne w</ but -
ton>
    <div *ngIf= " sho -
wNe w">
        <form (ngSubmit)
= " onS ave ()">
            <table
style= " border: 1px
solid white">
                <tr>
                    -
<td >SK U</ td>
                    -
<td ><input type="t -
ext " class= " for m-c -
ont rol " [(ngMo -
del )]= " sto ckR eg.S -
KU " name="S KU"> </t d>
                </t r>
                <tr>
                    -
<td >Na ma< /td>
```

my-app/src/app/app.componen-t.html (cont)

```
>          <td><input type="text"
class="form-control" [(ngMo-
del)]="stockReg.nama"
name="nama"></td>
        </tr>
        <tr>
          <td>Kuantitas</td>
          <td><input type="number"
class="form-control" [(ngMo-
del)]="stockReg.kuantitas"
name="kuantitas"></td>
        </tr>
        <tr>
          <td>hargaBeli</td>
          <td><input type="number"
class="form-control" [(ngMo-
del)]="stockReg.hargaBeli"
name="hargaBeli"></td>
        </tr>
        <tr>
          <td>hargaJual</td>
          <td><input type="number"
class="form-control" [(ngMo-
del)]="stockReg.hargaJual"
name="hargaJual"></td>
        </tr>


    </table>

    <button type="submit">
{{submitType}}</button>
```

my-app/src/app/app.componen-t.html (cont)

```
>      <button type="submit"
(click)="onCancel()">Cancel</-
button>
   </form>
 </div>


 <h2>Result</h2>
 <table class='tabel2'>
  <thead>
   <tr>
    <th>No.</th>
    <th>SKU</th>
    <th>Nama</th>
    <th>Kuantitas </th>
    <th>Harga beli</th>
    <th>Harga jual</th>
    <th>Profit per item</th>
    <th colspan="2">Action</-
th>
   </tr>
  </thead>
  <tbody>
   <tr *ngFor="let stc of
listStock; let i = index">
    <th scope="row">{{ i+1 }}
</th>
    <td scope="row">{{
stc.SKU }}</td>
    <td>{{ stc.nama }}</td>
    <td>{{ stc.kuantitas }}</td>
    <td>{{ stc.hargaBeli }}</td>
```

Tampilkan Data	{{ title }}
Tombol untuk Buka Form Baru	<button type="s-ubmit" class="btn btn-primary" (click)="onNew()"->New</button>
If	<div *ngIf="sho-wNew">
Form Submit	<form (ngSubmit) = "onSave()"></f-orm>
Binding	<td><input type="t-ext" class="form-c-ontrol" [(ngMo-del)]="stockReg.S-KU" name="S-KU"></td>
Tombol Submit	<button type="submit"> {{submitType}} </button>

By **amihapsari**

cheatography.com/amihapsari/

Published 22nd May, 2019.
Last updated 22nd May, 2019.

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

my-app/src/app/app.componen-t.html (cont)

```
>       <td>{{ stc.hargaJual }}
</td>
      <td>{{ stc.getProfit() }}</td>
      <td><button type="button"
(click)="onEdit(i)">Edit</button>
</td>
      <td><button type="button"
(click)="onDelete(i)">Dele-
te</button></td>
    </tr>
   </tbody>

 </table>
```

Kelas

```
class StockBarang{
    con str uctor(
        public SKU:
string = '',
        public
kuantitas: number =
null,
   ){}
    get Pro fit(){
        thi s.p rofit =
this.h arg aJu al- thi -
s.h arg aBeli;
        return this.p -
rofit;
   }
}
```

Inisiasi

```
export class
AppComponent implements
OnInit{
    lis tStock: StockB -
arang[] = [];
```

Inisiasi (cont)

```
>  stockReg: StockBarang;
  showNew: Boolean = false;
  selectedRow: number;
  submitType: string = 'Save';
  constructor(){
    this.listStock.push(new
StockBarang('AIR001', 'Aqua',
10, 3000, 6000, null));
  ngOnInit(){}
}
```

create

```
onNew(){
        thi s.s tockReg
= new StockB ara ng();
        thi s.s ubm -
itType = 'Save';
        thi s.s howNew =
true;
   }
```

save

```
onSave(){
        if( thi s.s -
ubm itType === 'Save'){
            thi s.l -
ist Sto ck.p us h(t -
his.st ock Reg);
       }
        else{
            //U pdate
```

save (cont)

```
>     this.listStock[this.selecte-
dRow].SKU = this.stockRe-
g.SKU;
   }
   //Hide
   this.showNew = false;
 }
```

edit

```
onEdit(index: number){
            thi s.s -
ubm itType = 'Update';
            thi s.s -
ele ctedRow = index;
            thi s.s -
tockReg = new StockB -
ara ng();
            thi s.s -
tockReg = Object.as -
sig n({},
                thi -
s.l ist Sto ck[ thi -
s.s ele cte dRow]);
            thi s.s -
howNew = true;
   }
```

delete

```
onDelete(index: number){
        thi s.l ist -
Sto ck.s pl ice (index,
1);
   }
```

Cancel

```
onCancel(){
        thi s.s howNew =
false;
   }
```

my-app/src/app/app.compo-nent.ts

```
import { Component,
OnInit } from
'@angular/core';
class StockB arang{
    con str uctor(
        public SKU:
string = '',
        public nama:
string = '',
        public
kuantitas: number =
null,
        public
hargaBeli: number =
null,
        public
hargaJual: number =
null,
        public profit:
number = null,
   ){}
    get Pro fit(){
        thi s.p rofit =
this.h arg aJu al- thi -
s.h arg aBeli;
        return this.p -
rofit;
   }
}
@Compo nent({
    sel ector: 'app-r -
oot',
    tem pla teUrl:
'./app.co mpo nen t.h -
tml',
    sty leUrls: ['./ap -
p.c omp one nt.c ss']
})
export class AppCom -
ponent implements
OnInit{
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

my-app/src/app/app.component.ts (cont)

```
>   listStock: StockBarang[] = [];
  stockReg: StockBarang;
  showNew: Boolean = false;
  selectedRow: number;
  submitType: string = 'Save';
  constructor(){
    this.listStock.push(new
StockBarang('AIR001', 'Aqua',
10, 3000, 6000, null));
    this.listStock.push(new
StockBarang('SOF001', 'Coca
Cola', 20, 5000, 4500, null));
    this.listStock.push(new
StockBarang('SNK001', 'Chitato',
15, 4000, 5000, null));
  }
  ngOnInit(){}
  onNew(){
    this.stockReg = new StockB-
arang();
    this.submitType = 'Save';
    this.showNew = true;
  }
  onSave(){
```

my-app/src/app/app.component.ts (cont)

```
>     if(this.submitType ===
'Save'){
      this.listStock.push(this.st-
ockReg);
    }
    else{
      //Update
      this.listStock[this.selecte-
dRow].SKU = this.stockRe-
g.SKU;
      this.listStock[this.selecte-
dRow].nama = this.stockRe-
g.nama;
      this.listStock[this.selecte-
dRow].kuantitas = this.stockRe-
g.kuantitas;
      this.listStock[this.selecte-
dRow].hargaBeli = this.stockRe-
g.hargaBeli;
      this.listStock[this.selecte-
dRow].hargaJual = this.stockRe-
g.hargaJual;
      this.listStock[this.selecte-
dRow].profit = this.stockReg.p-
rofit;
    }
    //Hide
    this.showNew = false;
  }
  onEdit(index: number){
      this.submitType = 'Update';
```

my-app/src/app/app.component.ts (cont)

```
>       this.selectedRow = index;
      this.stockReg = new StockB-
arang();
      this.stockReg = Object.as-
sign({},
        this.listStock[this.selecte-
dRow]);
      this.showNew = true;
  }
  onDelete(index: number){
    this.listStock.splice(index, 1);
  }

  onCancel(){
    this.showNew = false;
  }
}
```

my-app/src/app/app.module.ts

```
import { BrowserModule }
from '@angular/platform-
browser';
import { NgModule } from
'@angu lar /core';
import { FormsM odule }
from '@angu lar /fo -
rms';
import { AppCom ponent }
from './app.co mpo -
nent';
@NgMod ule({
    dec lar ations: [
        App Com ponent
```

my-app/src/app/app.module.ts (cont)

```
>   ],
  imports: [
    BrowserModule,
    FormsModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com