Desenvolvimento de Aplicações Móveis usando HTML5 e PhoneGap (Cordova) https://goo.gl/5AdgAh
Turma: 13MOB
Prof. Henrique Poyatos
Módulo 3 - Angular 4.2.5
Laboratório DespesasApp - 09/10/2017

Atom
atom-typescript
linter
linter-ui-default
ionic-preview
autocomplete-ionic2-framework
angular-2-typeScript-snippets
emmet

DespesasApp usando Angular4 e angular-cli

1. Instalar o node.js (6.9.1), inclui o npm 3.10.8;

Passo opcional: Instale o Atom.io com o plugins de typescript, angular2;

2. Instalar na linha de comando:

```
npm install -g angular-cli
```

3. Criar o projeto usando o angular-cli (use a pasta Documents):

```
ng new DespesasApp
```

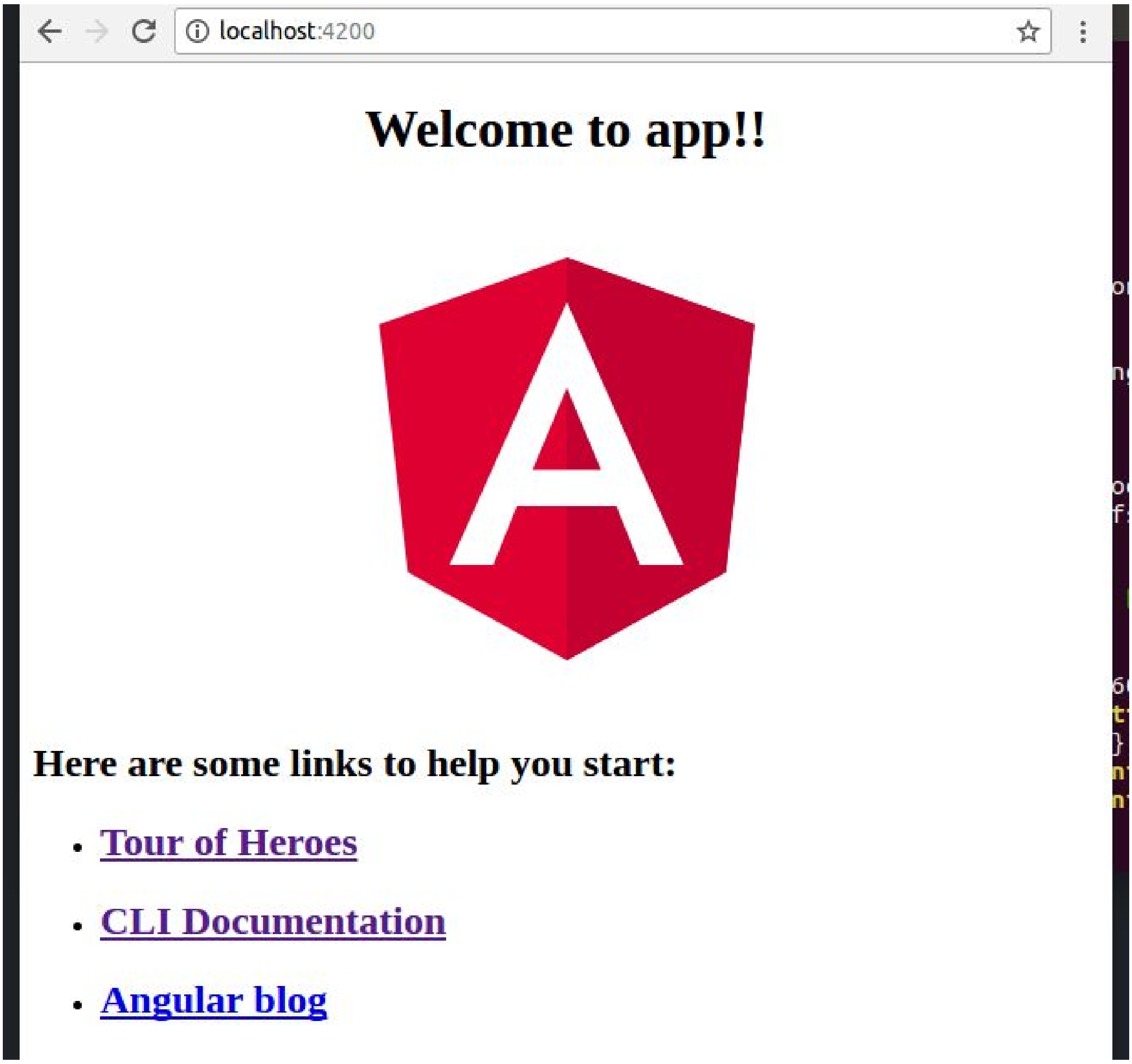
4. Entrar na pasta de projeto e instalar o angular-2-local-storage:

```
cd DespesasApp
npm install angular-2-local-storage
```

5. Subir o servidor para testar:

```
ng serve
```

6. Acesse http://localhost:4200/:



Welcome to app!!

Here are some links to help you start:

- Tour of Heroes
- CLI Documentation
- Angular blog

7. Alterar em **./src/index.html**:

```
<head>
    <title>DespesasApp em Angular4</title>
    <base href="/">
```

Roteamento em Single Page Application

8. Alterar em **./src/app/app.component.html**:

```
<h1>Despesas App</h1>
<router-outlet></router-outlet>
```

9. Criar o módulo de roteamento “routing” e os componentes “list” e “add”:

```
ng g module routing
ng g component list
ng g component add
```

10. Alterar em **app/routing/routing.module.ts**:

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

import { RouterModule, Routes } from '@angular/router';
import { ListComponent }  from '../list/list.component';
import { AddComponent }   from '../add/add.component';

const routes: Routes = [
  { path: '', redirectTo: '/list', pathMatch: 'full' },
  { path: 'list',  component: ListComponent },
  { path: 'add',  component: AddComponent }
];

@NgModule({
  imports: [
    CommonModule,
  RouterModule.forRoot(routes)
  ],
  exports: [ RouterModule ],
  declarations: []
})
export class RoutingModule { }
```

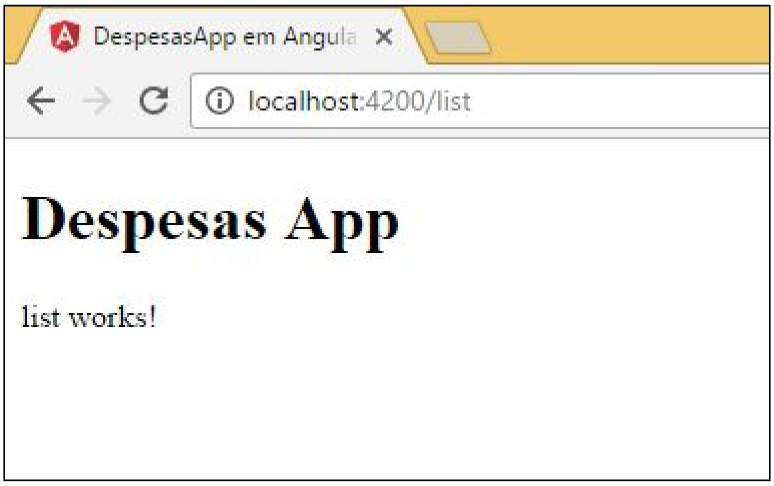
11. Alterar em **app/app.module.ts**:

```
/* Módulos */
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { HttpModule } from '@angular/http';
import { RoutingModule } from './routing/routing.module';

/* Componentes */
import { AppComponent } from './app.component';
~~import { RoutingComponent } from './routing/routing.component';~~
import { ListComponent } from './list/list.component';
import { AddComponent } from './add/add.component';

@NgModule({
  declarations: [
    AppComponent,
~~    RoutingComponent,~~
    ListComponent,
    AddComponent
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpModule,
    RoutingModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

12. Subir o servidor para testar:

```
ng serve
```

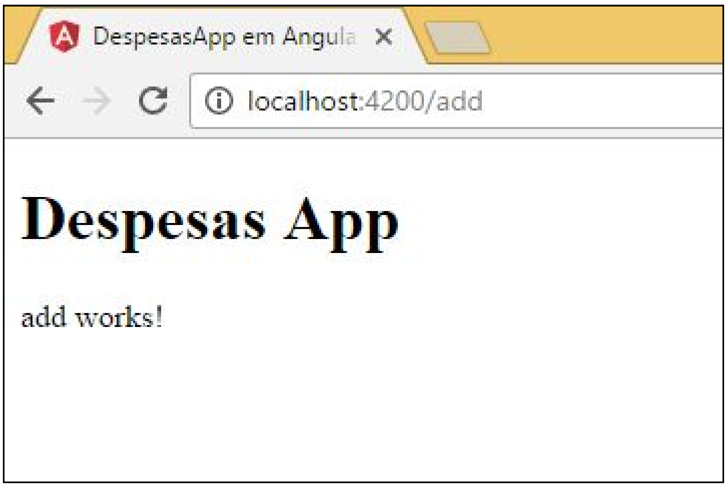
13. Acesse **http://localhost:4200/**:



Despesas App

list works!

14. Acesse **http://localhost:4200/add**:



Despesas App

add works!

Instalação do ng-bootstrap

npm install --save @ng-bootstrap/ng-bootstrap

15. Baixar o Bootstrap 3.3.7, descompacte-o, renomeie sua pasta para ‘boostrap’ e mova-a para **./src**.

16. Alterar **index.html**:

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>DespesasApp em Angular 4</title>
  <base href="/">
  <link
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstra
p.min.css" rel="stylesheet">

  <meta name="viewport" content="width=device-width,
initial-scale=1">
  <link rel="icon" type="image/x-icon" href="favicon.ico">
</head>
<body>
  <app-root></app-root>
</body>
</html>
```

17. Instalar o ng-bootstrap usando o npm:

```
npm install --save @ng-bootstrap/ng-bootstrap
```

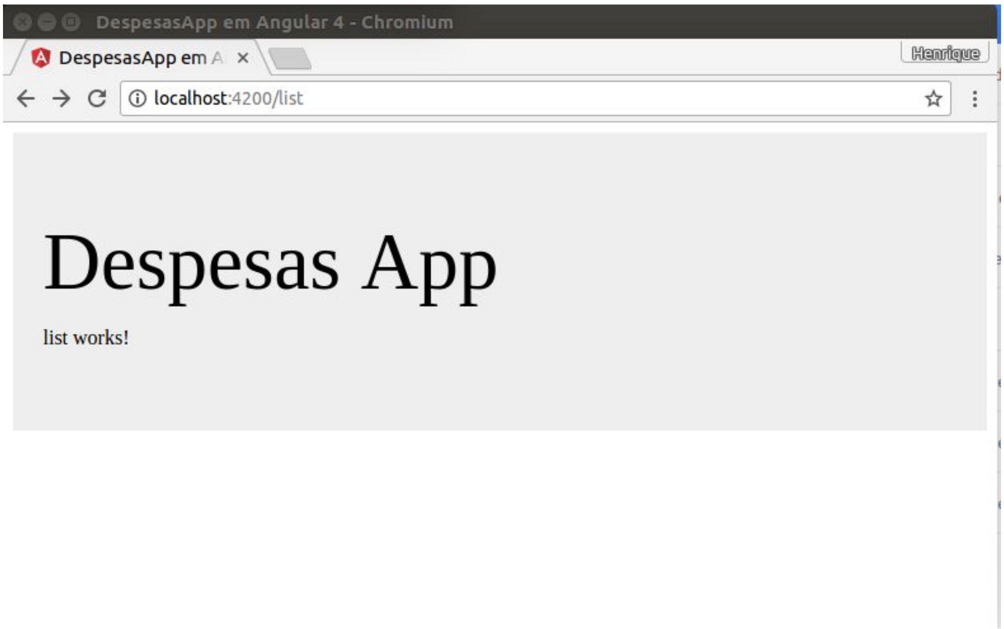
18. Alterar em **app/app.module.ts**:

```
/* Módulos */
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { HttpModule } from '@angular/http';
import { RoutingModule } from './routing/routing.module';
import { NgbModule } from '@ng-bootstrap/ng-bootstrap';

/* Componentes */
import { AppComponent } from './app.component';

import { ListComponent } from './list/list.component';
import { AddComponent } from './add/add.component';

@NgModule({
  declarations: [
    AppComponent,
    ListComponent,
    AddComponent
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpModule,
    RoutingModule,
    NgbModule.forRoot()
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

19. Alterar **app/app.component.html**:

```
<div class="jumbotron">
  <div class="container">
    <h1>Despesas App</h1>
    <router-outlet></router-outlet>
  </div>
</div>
```

20. Acesse **http://localhost:4200/**:



Despesas App

list works!

21. Remover de **app/app.component.html**:

```
~~<div class="jumbotron">~~
~~  <div class="container">~~
    <h1>Despesas App</h1>
    <router-outlet></router-outlet>
```

```
  ~~</div>~~
~~</div>~~
```

Classe de Despesa e serviço

22. Criar uma classe despesa e um serviço de despesa:

```
ng g class despesa
ng g service despesa
```

23. Alterar em **./src/app/despesa.ts**:

```
export class Despesa {
  id: number;
  tipo: string;
  icone: string;
  nome: string;
  data: Date;
  valor: number;

  constructor(
    id: number,
  tipo: string,
  icone: string,
  nome: string,
  data: Date,
  valor: number
  )
  {
    this.id = id;
    this.tipo = tipo;
    this.icone = icone;
    this.nome = nome;
    this.data = data;
    this.valor = valor;
  }
}
```

24. Alterar em **./src/app/despesa.service.ts**:

```
import { Injectable } from '@angular/core';
import { Despesa } from './despesa';

@Injectable()
export class DespesaService {
  constructor() { }

  getAll(): Despesa[] {
      let despesas: Despesa[] = [
        new Despesa(1, "Mercado", "glyphicon-shopping-cart",
"Extra", new Date("2016-10-01T03:00:00Z"), 200.5),
        new Despesa(2, "Lazer", "glyphicon-knight", "Cinema", new
Date("2016-10-20T03:00:00Z"), 59.9)

                         ];
      return despesas;
    }

}
```

25. Alterar em **./src/app/app.module.ts**:

```
/* Módulos */
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { HttpModule } from '@angular/http';
import { RoutingModule } from './routing/routing.module';
import { NgbModule } from '@ng-bootstrap/ng-bootstrap';

/* Serviços */
import { DespesaService } from './despesa.service';

/* Componentes */
import { AppComponent } from './app.component';

import { ListComponent } from './list/list.component';
import { AddComponent } from './add/add.component';

@NgModule({
  declarations: [
    AppComponent,
    ListComponent,
    AddComponent
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpModule,
    RoutingModule,
    NgbModule.forRoot()
  ],
  providers: [DespesaService],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

Lista de Despesas (dinâmica a partir do Typescript)

26. Alterar em **./src/app/list/list.component.ts**:

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

/* Serviços */
import { DespesaService }        from '../despesa.service';

/* Classes */
import { Despesa }               from '../despesa';

@Component({
  selector: 'app-list',
  templateUrl: './list.component.html',
  styleUrls: ['./list.component.css']
})
export class ListComponent implements OnInit {
  despesas: Despesa[];

  constructor(private despesaService: DespesaService) { }

  ngOnInit() {
    this.despesas = this.despesaService.getAll();
  }
}
```

27. Alterar em **./src/app/list/list.component.html**:

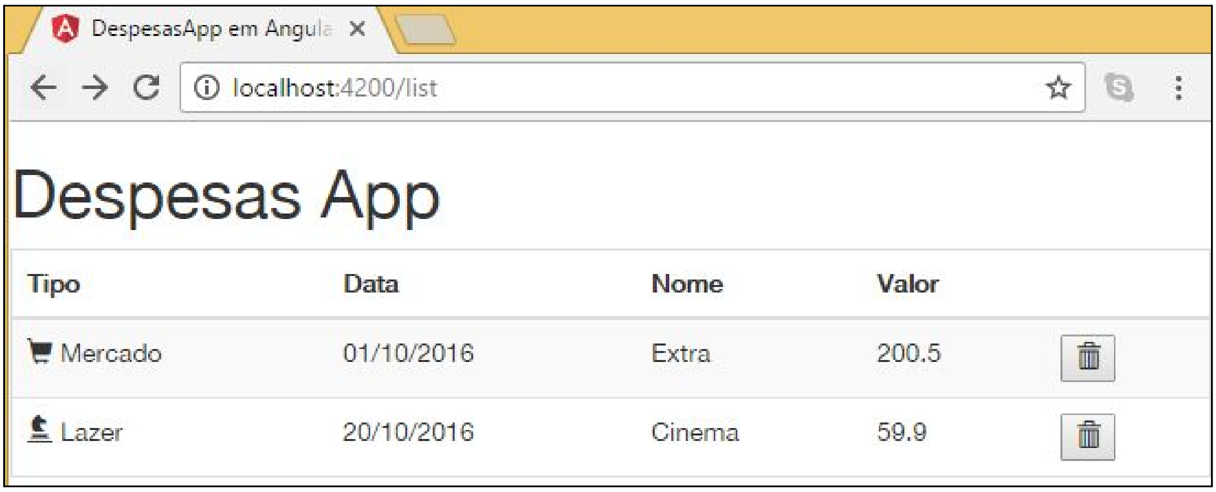
```
<div class="table-responsive">
            <table class="table table-striped">
                <thead>
                    <tr>
                        <th>Tipo</th>
                        <th>Data</th>
                        <th>Nome</th>
                        <th>Valor</th>
                        <th></th>
                    </tr>
                </thead>
                <tbody>
```

```
                    <tr *ngFor="let despesa of despesas">
                        <td><div class="glyphicon
{{despesa.icone}}"></div> {{despesa.tipo}}</td>
                        <td>{{despesa.data | date: 'dd/MM/yyyy
HH:mm:ss' }}</td>
                        <td>{{despesa.nome}}</td>
                        <td>{{despesa.valor}}</td>
                        <td>
                    <button class="delete"><div class="glyphicon
glyphicon-trash"></div></button>
                   </tr>
                </tbody>
              </table>
</div>
```

28. Subir o servidor para testar:

```
ng serve
```

29. Acesse **http://localhost:4200/**:



Despesas App

Tipo	Data	Nome	Valor	
Mercado	01/10/2016	Extra	200.5	🗑
Lazer	20/10/2016	Cinema	59.9	🗑

Linkando a listagem com o formulário de adicionar despesa

30. Alterar em **app/list/list.component.ts**:

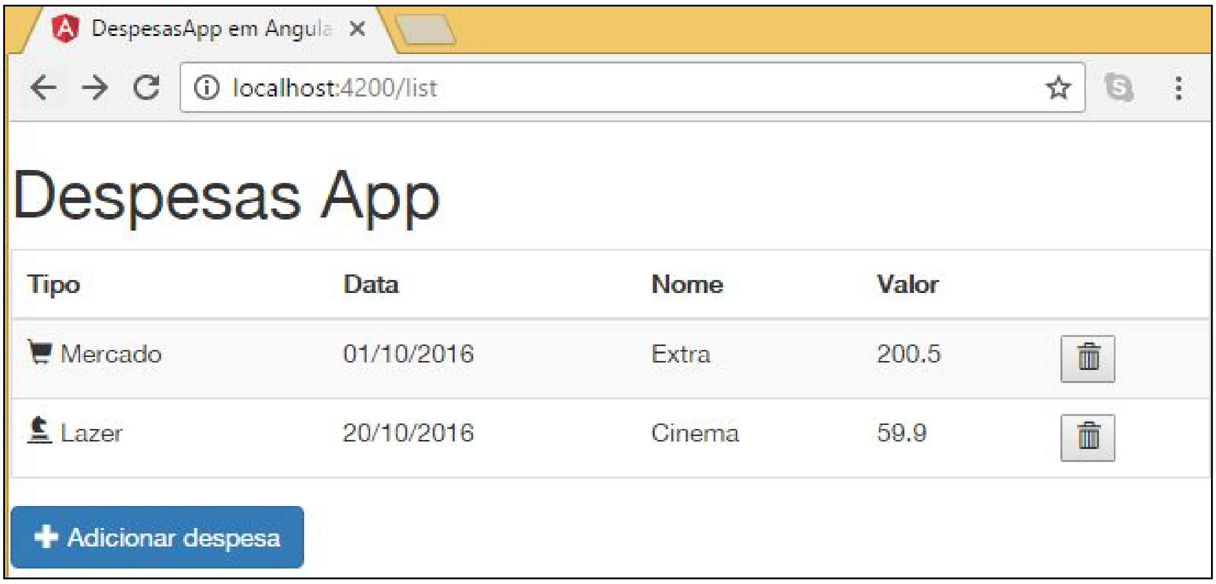
```
import { Component, OnInit } from '@angular/core';
import { Router }            from '@angular/router';

/* Serviços */
import { DespesaService }         from '../despesa.service';

/* Classes */
import { Despesa }                from '../despesa';

@Component({
  selector: 'app-list',
  templateUrl: './list.component.html',
  styleUrls: ['./list.component.css']
})
export class ListComponent implements OnInit {
  despesas: Despesa[];

  constructor(private despesaService: DespesaService,
              private router: Router) { }

  ngOnInit() {
    this.despesas = this.despesaService.getAll();
    console.log(this.despesas);
  }

  add(): void {
    this.router.navigate(['/add']);
  }
}
```

31. Alterar em **app/list/list.component.html**:

```
<div class="table-responsive">
            <table class="table table-striped">
                <thead>
                    <tr>
                        <th>Tipo</th>
                        <th>Data</th>
                        <th>Nome</th>
                        <th>Valor</th>
                        <th></th>
                    </tr>
                </thead>
                <tbody>
                    <tr *ngFor="let despesa of despesas">
                        <td><div class="glyphicon
{{despesa.icone}}"></div> {{despesa.tipo}}</td>
                        <td>{{despesa.data | date: 'dd/MM/yyyy'
}}</td>
                        <td>{{despesa.nome}}</td>
                        <td>{{despesa.valor}}</td>
                        <td>
                    <button class="delete"><div class="glyphicon
glyphicon-trash"></div></button>
                   </tr>
                </tbody>
              </table>
</div>
<button class="btn btn-primary" (click)="add()"><div
```

```
class="glyphicon glyphicon-plus"></div> Adicionar despesa</button>
```

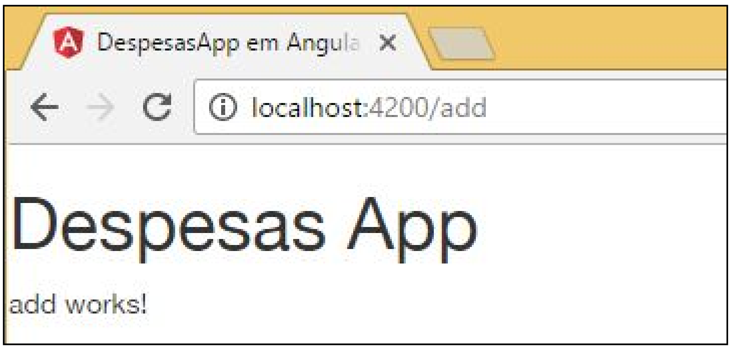
32. Acesse **http://localhost:4200/**:



Despesas App

Tipo	Data	Nome	Valor
Mercado	01/10/2016	Extra	200.5
Lazer	20/10/2016	Cinema	59.9

33. Clicar no botão “Adicionar despesa”:



Despesas App

add works!

Formulário de adicionar despesa

34. Alterar em **./src/app/app.module.ts**:

```
/* Módulos */
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { RoutingModule } from './routing/routing.module';
import { NgbModule } from '@ng-bootstrap/ng-bootstrap';
import { FormsModule, ReactiveFormsModule } from '@angular/forms';

/* Componentes */
import { AppComponent } from './app.component';
import { ListComponent } from './list/list.component';
import { AddComponent } from './add/add.component';
```

```
/* Serviços */
import { DespesaService } from './despesa.service';

@NgModule({
  declarations: [
    AppComponent,
    ListComponent,
    AddComponent
  ],
  imports: [
    BrowserModule,
    RoutingModule,
    NgbModule.forRoot(),
    FormsModule
  ],
  providers: [DespesaService],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

35. Alterar em **./src/app/add/add.component.ts**:

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

/* Serviços */
import { DespesaService }        from '../despesa.service';

/* Classes */
import { Despesa }               from '../despesa';

@Component({
  selector: 'app-add',
  templateUrl: './add.component.html',
  styleUrls: ['./add.component.css']
})
export class AddComponent implements OnInit {
  despesa: Despesa;

  constructor(private despesaService: DespesaService) { }

  ngOnInit() {
    this.despesa = new Despesa(new Date().getTime(), "", "", "",
new Date(), 0);
  }

  save(): void {
    this.despesaService.save(this.despesa);
  }
}
```

36. Alterar em **app/despesa.service.ts**:

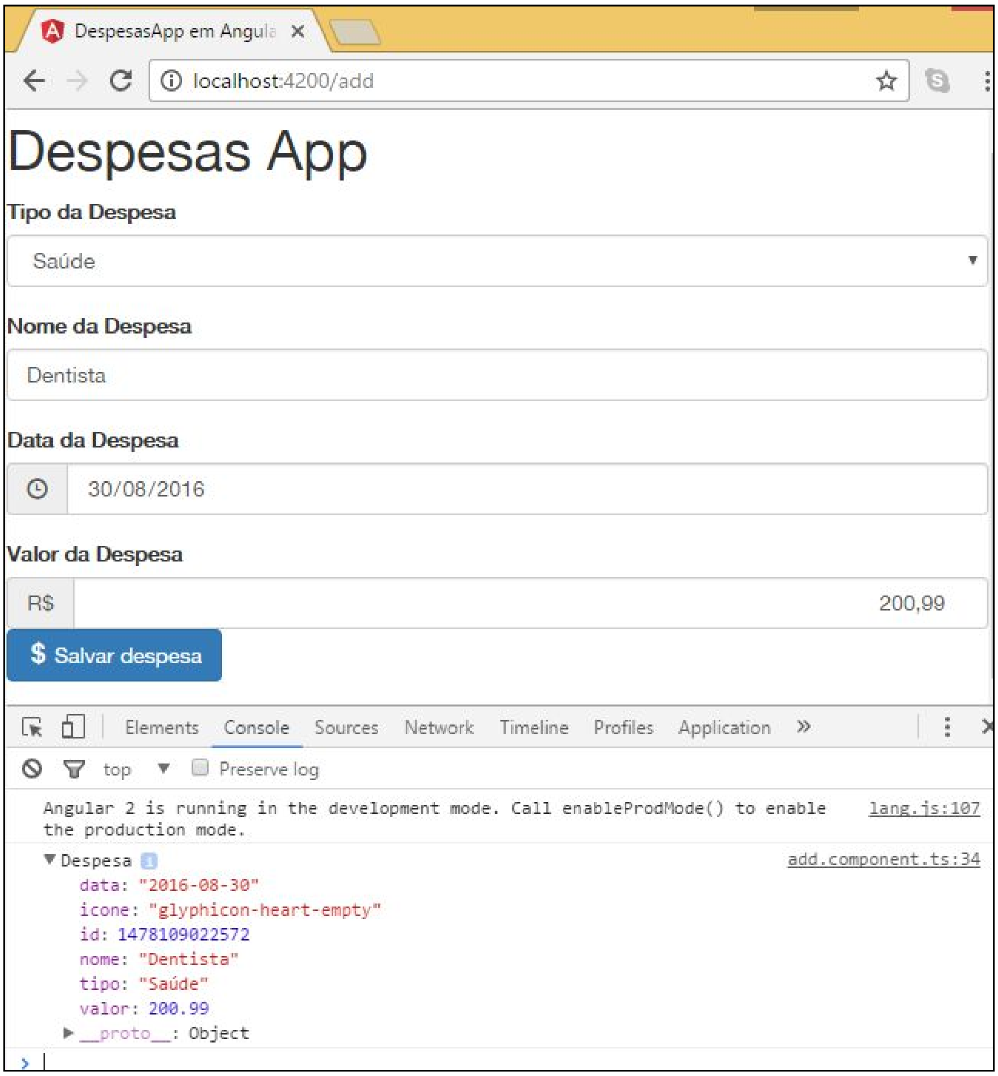
```
import { Injectable } from '@angular/core';

/* Classes */
import { Despesa }                from './despesa';

@Injectable()
export class DespesaService {

  constructor() { }

  getDespesas(): Despesa[] {
      let despesas: Despesa[] = [
        new Despesa(1, "Mercado", "glyphicon-shopping-cart",
"Extra", new Date("2016-10-01T03:00:00Z"), 200.5),
        new Despesa(2, "Lazer", "glyphicon-knight", "Cinema", new
Date("2016-10-20T03:00:00Z"), 59.9)
      ];
      return despesas;
    }

  save(despesa: Despesa): void {
    switch(despesa.tipo)
    {
      case "Mercado":
        despesa.icone = "glyphicon-shopping-cart";
      break;
      case "Lazer":
        despesa.icone = "glyphicon-knight";
      break;
      case "Saúde":
        despesa.icone = "glyphicon-heart-empty";
      break;
    }

    console.log(despesa);
  }
}
```

37. Alterar em **app/add/add.component.html**:

```
<div>
    <div class="form-group">
      <label for="tipo">Tipo da Despesa</label>
      <select [(ngModel)]="despesa.tipo" id="tipo"
class="form-control">
```

```
        <option value="Mercado">Mercado</option>
        <option value="Lazer">Lazer</option>
        <option value="Saúde">Saúde</option>
      </select>
    </div>
    <div class="form-group">
      <label for="nome">Nome da Despesa</label>
      <input type="text" [(ngModel)]="despesa.nome" id="nome"
required placeholder="Digite o nome..." class="form-control">
           </div>
    <div class="form-group">
      <label for="data">Data da Despesa</label>
      <div class="input-group">
        <div class="input-group-addon"><span class="glyphicon
glyphicon-time"></span></div>
        <input type="date" [(ngModel)]="despesa.data" id="data"
class="form-control" required>
      </div>
    </div>
           <div class="form-group">
      <label for="valor">Valor da Despesa</label>
      <div class="input-group">
        <div class="input-group-addon">R$</div>
        <input type="number" [(ngModel)]="despesa.valor"
id="valor" class="form-control text-right" required
placeholder="Digite o valor...">
                </div>
      <div class="form-group">
          <button class="btn btn-primary" (click)="save()"><div
class="glyphicon glyphicon-usd"></div> Salvar despesa</button>
      </div>
    </div>
</div>
```

38. Testar a adição de uma despesa:



Despesas App

Tipo da Despesa: Saúde

Nome da Despesa: Dentista

Data da Despesa: 30/08/2016

Valor da Despesa: R$ 200,99

$ Salvar despesa

```
Angular 2 is running in the development mode. Call enableProdMode() to enable    lang.js:107
the production mode.
▼Despesa                                                                   add.component.ts:34
    data: "2016-08-30"
    icone: "glyphicon-heart-empty"
    id: 1478109022572
    nome: "Dentista"
    tipo: "Saúde"
    valor: 200.99
  ▶__proto__: Object
```

Armazenando no LocalStorage

39. Alterar em **./src/app/app.module.ts**:

```
/* Módulos */
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { HttpModule } from '@angular/http';
import { RoutingModule } from './routing/routing.module';
import { NgbModule } from '@ng-bootstrap/ng-bootstrap';
import { FormsModule, ReactiveFormsModule } from
'@angular/forms';


/* Serviços */
import { DespesaService } from './despesa.service';

/* Componentes */
```

```
import { AppComponent } from './app.component';

import { ListComponent } from './list/list.component';
import { AddComponent } from './add/add.component';

/* LocalStorage */
import { LocalStorageModule } from 'angular-2-local-storage';

@NgModule({
  declarations: [
    AppComponent,
    ListComponent,
    AddComponent
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpModule,
    RoutingModule,
    NgbModule.forRoot(),
    LocalStorageModule.withConfig({
      prefix: 'despesasApp',
      storageType: 'localStorage'
    })
  ],
  providers: [ DespesaService ],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

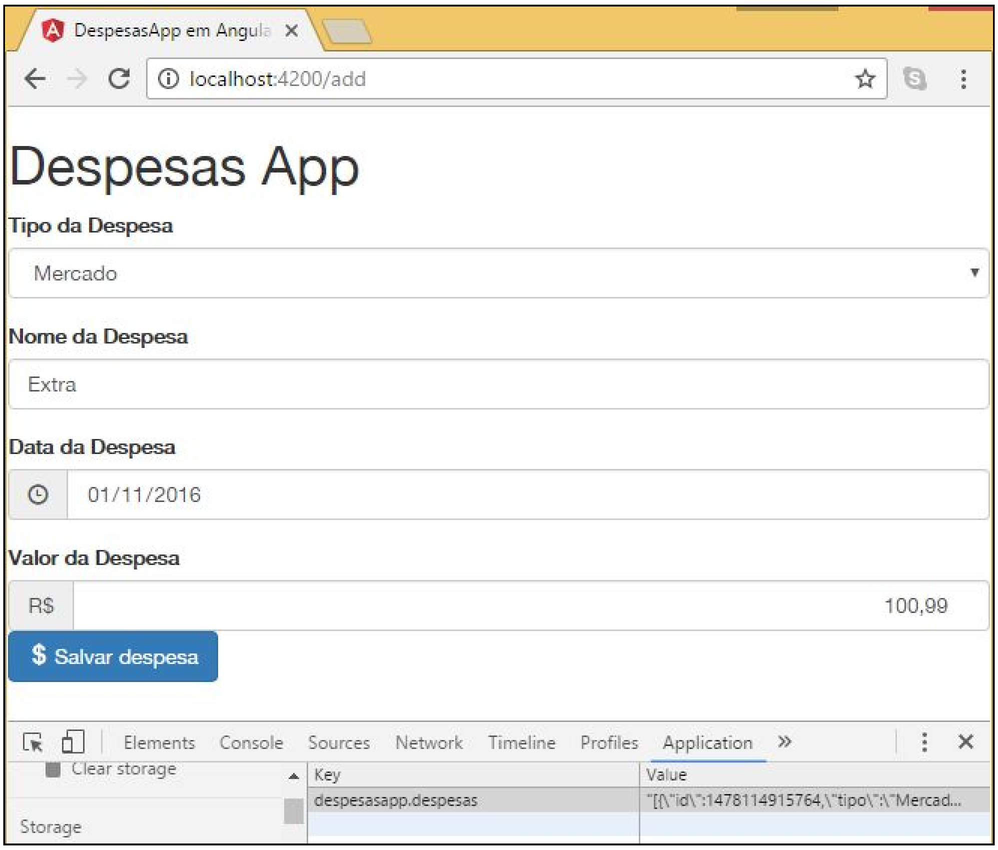
40. Alterar em **./src/app/despesa.service.ts**:

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

/* LocalStorage */
import { LocalStorageService } from 'angular-2-local-storage';

/* Classes */
import { Despesa }                from './despesa';

@Injectable()
export class DespesaService {
  private despesas: Despesa[];

  constructor(private localStorageService: LocalStorageService) {
}

  getDespesas(): Despesa[] {
      let a: Despesa[] = [
        new Despesa(1, "Mercado", "glyphicon-shopping-cart",
"Extra", new Date("2016-10-01T03:00:00Z"), 200.5),
        new Despesa(2, "Lazer", "glyphicon-knight", "Cinema", new
Date("2016-10-20T03:00:00Z"), 59.9)
      ];
      return a;
    }

  getDb(): void{
    this.despesas = [];
    if (this.localStorageService.get("despesas") != null)
    {
        let jsonObjectArray =
JSON.parse(<string>this.localStorageService.get("despesas"));
        for (let jsonObject of jsonObjectArray)
        {
          this.despesas.push(new Despesa(jsonObject.id,
jsonObject.tipo, jsonObject.icone, jsonObject.nome, new
Date(jsonObject.data), jsonObject.valor));
        }
    }
  }

  save(despesa: Despesa): void {
    switch(despesa.tipo)
    {
      case "Mercado":
        despesa.icone = "glyphicon-shopping-cart";
      break;
      case "Lazer":
        despesa.icone = "glyphicon-knight";
      break;
      case "Saúde":
        despesa.icone = "glyphicon-heart-empty";
      break;
    }
    ~~console.log(this.despesa);~~
    this.getDb();
    this.despesas.push(despesa);
    this.localStorageService.set("despesas",
JSON.stringify(this.despesas));

  }
}
```

41. Subir o servidor para testar:

```
ng serve
```

42. Testar a adição de uma despesa



Despesas App

Tipo da Despesa: Mercado

Nome da Despesa: Extra

Data da Despesa: 01/11/2016

Valor da Despesa: R$ 100,99

$ Salvar despesa

Key	Value
despesasapp.despesas	"[{\"id\":1478114915764,\"tipo\":\"Mercad...

Adicionando roteamento

43. Alterar em **./src/app/add/add.component.ts**:

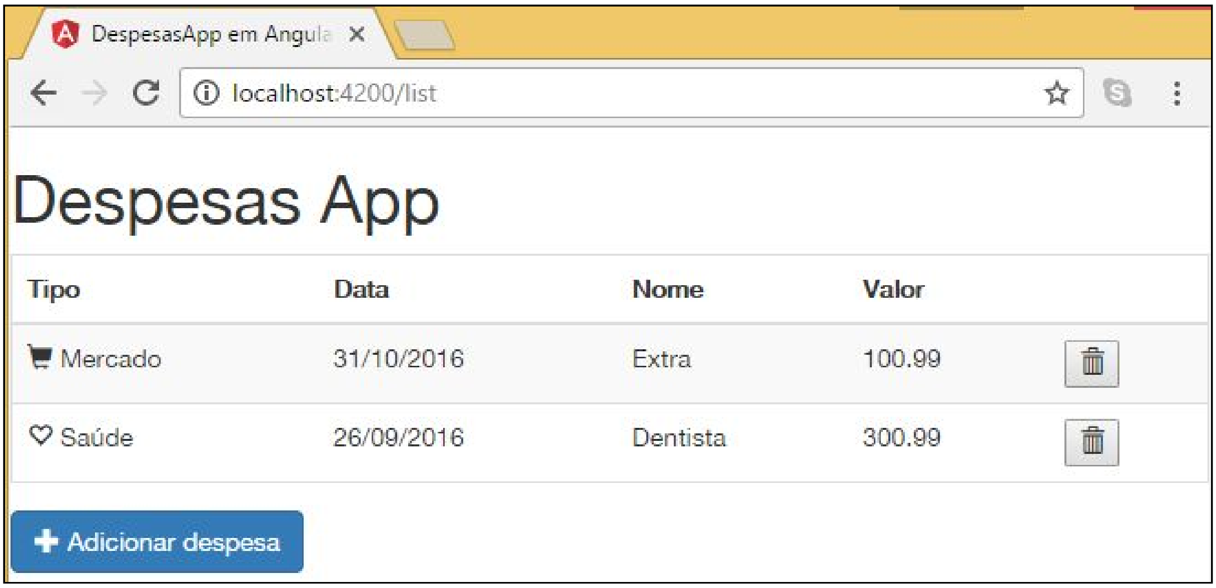
```
import { Component, OnInit } from '@angular/core';
import { Router }            from '@angular/router';

/* Serviços */
import { DespesaService }         from '../despesa.service';

/* Classes */
import { Despesa }                from '../despesa';

@Component({
  selector: 'app-add',
  templateUrl: './add.component.html',
  styleUrls: ['./add.component.css']
})
export class AddComponent implements OnInit {
  despesa: Despesa;

  constructor(private despesaService: DespesaService,
              private router: Router) { }
  ngOnInit() {
    this.despesa = new Despesa(new Date().getTime(), "", "", "",
new Date(), 0);
  }

  save(): void {
    this.despesaService.save(this.despesa);
    this.router.navigate(['/list']);
  }
}
```

Listando a partir do LocalStorage

44. Alterar em **./src/app/despesa.service.ts**:

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

/* LocalStorage */
import { LocalStorageService } from 'angular-2-local-storage';

/* Classes */
import { Despesa }                from './despesa';

/* Interfaces */
import { Despesajson }                from './despesajson';

@Injectable()
export class DespesaService {
  private despesas: Despesa[];

  constructor(private localStorageService: LocalStorageService) {
    this.despesas = [];
  }

  getDespesas(): Despesa[] {
      this.getDb();
      return this.despesas;
    }

  getDb(): void{
    this.despesas = [];
    console.log(this.localStorageService.get("despesas"));
    if (this.localStorageService.get("despesas") != null)
    {
        let jsonObjectArray =
JSON.parse(this.localStorageService.get("despesas"));
        for (let jsonObject of jsonObjectArray)
        {
          this.despesas.push(new Despesa(jsonObject.id,
jsonObject.tipo, jsonObject.icone, jsonObject.nome, new
Date(jsonObject.data), jsonObject.valor));
        }
    }
  }

  save(despesa: Despesa): void {
    switch(despesa.tipo)
    {
      case "Mercado":
        despesa.icone = "glyphicon-shopping-cart";
      break;
      case "Lazer":
        despesa.icone = "glyphicon-knight";
      break;
      case "Saúde":
        despesa.icone = "glyphicon-heart-empty";
      break;
    }
    this.getDb();
    this.despesas.push(despesa);
    this.localStorageService.set("despesas",
JSON.stringify(this.despesas));

  }
}
```

45. Testar a listagem:



Despesas App

Tipo	Data	Nome	Valor	
Mercado	31/10/2016	Extra	100.99	
♡ Saúde	26/09/2016	Dentista	300.99	

Apagando registros

46. Alterar em **.src/app/despesa.service.ts**:

```
...
  save(despesa: Despesa): void {
    switch(despesa.tipo)
    {
      case "Mercado":
        despesa.icone = "glyphicon-shopping-cart";
      break;
      case "Lazer":
        despesa.icone = "glyphicon-knight";
      break;
      case "Saúde":
        despesa.icone = "glyphicon-heart-empty";
      break;
    }
    this.getDb();
    this.despesas.push(despesa);
    this.localStorageService.set("despesas",
JSON.stringify(this.despesas));

  }

  delete(despesaId: number)
  {
    // this.getDb();
    for (var i=0; i < this.despesas.length; i++)
    {
      if (this.despesas[i].id == despesaId)
      {
        this.despesas.splice(i, 1);
      }
    }
    this.localStorageService.set("despesas",
JSON.stringify(this.despesas));
  }
}
```

47. Alterar em **./src/app/list/list.component.ts**:

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

/* Serviços */
import { DespesaService }         from '../despesa.service';

/* Classes */
import { Despesa }                from '../despesa';

@Component({
  selector: 'app-list',
  templateUrl: './list.component.html',
  styleUrls: ['./list.component.css']
})
export class ListComponent implements OnInit {
  despesas: Despesa[];

  constructor(private despesaService: DespesaService,
              private router: Router) { }

  ngOnInit() {
    this.despesas = this.despesaService.getAll();
  }

  add(): void {
    this.router.navigate(['/add']);
  }

  del(despesaId: number): void {
    this.despesaService.delete(despesaId);
  }
}
```

48. Alterar em **./src/app/list/list.component.html**:

```
<div class="table-responsive">
            <table class="table table-striped">
                <thead>
                    <tr>
                        <th>Tipo</th>
                        <th>Data</th>
                        <th>Nome</th>
                        <th>Valor</th>
                        <th></th>
                    </tr>
                </thead>
                <tbody>
                    <tr *ngFor="let despesa of despesas">
                        <td><div class="glyphicon
{{despesa.icone}}"></div> {{despesa.tipo}}</td>
                        <td>{{despesa.data | date: 'dd/MM/yyyy'
}}</td>
                        <td>{{despesa.nome}}</td>
                        <td>{{despesa.valor}}</td>
                        <td>
                    <button class="delete"
(click)="del(despesa.id)"><div class="glyphicon
glyphicon-trash"></div></button>
                   </tr>
                </tbody>
```

```
              </table>
</div>
<button class="btn btn-primary" (click)="add()"><div
class="glyphicon glyphicon-plus"></div> Adicionar
despesa</button>
```

49. Teste!

Corrigindo a data (adicionando 3 horas no UTC)

50. Alterar em **./src/app/add/add.component.html**:

```
    <div class="form-group">
      <label for="nome">Nome da Despesa</label>
      <input type="text" [(ngModel)]="despesa.nome" id="nome"
required placeholder="Digite o nome..." class="form-control">
          </div>
    <div class="form-group">
      <label for="data">Data da Despesa</label>
      <div class="input-group">
        <div class="input-group-addon"><span class="glyphicon
glyphicon-time"></span></div>
        <input type="date" [(ngModel)]="~~despesa.~~data" id="data"
class="form-control" required>
      </div>
    </div>
```

51. Alterar em **./src/app/add/add.component.ts**:

```
export class AddComponent implements OnInit {
  data: string;
  despesa: Despesa;

  constructor(private despesaService: DespesaService,
              private router: Router) { }

  ngOnInit() {
    this.despesa = new Despesa(new Date().getTime(), "", "", "",
new Date(), 0);
  }

  save(): void {
    this.despesa.data = new Date(this.data+"T12:00:00-03:00");
    this.despesaService.save(this.despesa);
    this.router.navigate(['/list']);
  }
}
```

52. Teste!

EXTRA: Um pipe para melhorar a formatação de moeda!

53. Criando um: pipe chamado Decimal Br:

```
ng g pipe decimalBr
```

54. Alterar em **./src/app/decimal-br.pipe.ts**:

```
import { Pipe, PipeTransform } from '@angular/core';

@Pipe({
  name: 'decimalBr'
})
export class DecimalBrPipe implements PipeTransform {

  transform(value: number): string {
    return value.toString().replace(",",
"&").replace(".",",").replace("&",".");
  }

}
```

55. Alterar em **./src/app/list/list.component.html**:

```
<div class="table-responsive">
            <table class="table table-striped">
                <thead>
                    <tr>
                        <th>Tipo</th>
                        <th>Data</th>
                        <th>Nome</th>
                        <th>Valor</th>
                        <th></th>
                    </tr>
                </thead>
                <tbody>
                    <tr *ngFor="let despesa of despesas">
                        <td><div class="glyphicon
{{despesa.icone}}"></div> {{despesa.tipo}}</td>
                        <td>{{despesa.data}}</td>
                        <td>{{despesa.nome}}</td>
                        <td>{{despesa.valor |
currency:'BRL':true:'1.2-2' | decimalBr}}</td>
                        <td>
                    <button class="delete"
(click)="del(despesa.id)"><div class="glyphicon
glyphicon-trash"></div></button>
```

```
                   </tr>
                </tbody>
              </table>
</div>
<button class="btn btn-primary" (click)="add()"><div
class="glyphicon glyphicon-plus"></div> Adicionar
despesa</button>
```

56. Teste!

EXTRA: Instalando o angular2-moment:

57. Instalar o angular2-moment:

```
npm install --save angular2-moment
```

58. Alterar em **./src/app/app.module.ts**:

```
/* Módulos */
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { HttpModule } from '@angular/http';
import { RoutingModule } from './routing/routing.module';
import { NgbModule } from '@ng-bootstrap/ng-bootstrap';
import { LocalStorageModule } from 'angular-2-local-storage';
import {MomentModule} from 'angular2-moment';
import 'moment/locale/pt-br';

/* Serviços */
import { DespesaService } from './despesa.service';

/* Componentes */
import { AppComponent } from './app.component';
import { ListComponent } from './list/list.component';
import { AddComponent } from './add/add.component';

import { DecimalBrPipe } from './decimal-br.pipe';

@NgModule({
  declarations: [
    AppComponent,
    ListComponent,
    AddComponent,
    DecimalBrPipe
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpModule,
    RoutingModule,
    NgbModule.forRoot(),
    LocalStorageModule.withConfig({
      prefix: 'despesasApp',
      storageType: 'localStorage'
    }),
    MomentModule
  ],
  providers: [DespesaService],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

59. Alterar em **./src/app/list/list.component.html**:

```
<div class="table-responsive">
            <table class="table table-striped">
                <thead>
                    <tr>
                        <th>Tipo</th>
                        <th>Data</th>
                        <th>Nome</th>
                        <th>Valor</th>
                        <th></th>
                    </tr>
                </thead>
                <tbody>
                    <tr [hidden]="invisiveis[despesa.id]"
*ngFor="let despesa of despesas">
                        <td><div class="glyphicon
{{despesa.icone}}"></div> {{despesa.tipo}}</td>
                        <td>{{despesa.data  | amTimeAgo ~~| date:
'dd/MM/yyyy'~~ }}</td>
                        <td>{{despesa.nome}}</td>
                        <td>{{despesa.valor |
currency:'BRL':true:'3.2-2' | decimalBr}}</td>
                        <td>
                    <button class="delete"
(click)="del(despesa.id)"><div class="glyphicon
glyphicon-trash"></div></button>
                   </tr>
                </tbody>
              </table>
</div>
<button class="btn btn-primary" (click)="add()"><div
class="glyphicon glyphicon-plus"></div> Adicionar
despesa</button>
```

60. Teste!