

Kisebb angular beépített funkciók használata

Az alábbi angular projekt azért készül, hogy megnézzük, hogyan lehet a következő funkciókat az angular belül használni:

1. JQuery importálás
2. ngIf funkció
3. ngFor funkció

Lépések listája

1. Elkészítem az angular projektet a `ng new KisebbAngularFunkciok --no-standalone`, majd a komponensek következnek
 - o Kezdőlap komponenshez: `ng g c kezdolap`
 - o Hibakezelő komponenshez: `ng g c hiba404`
 - o JQuery-t kipróbáló komponenshez: `ng g c jquery`
 - o ngIf használatát bemutató komponenshez: `ng g c ngif`
 - o ngFor használatát bemutató komponenshez: `ng g c ngfor`
 - o Adatokat loggoló feladat bemutatásához: `ng g c loggolo`
2. Routing elkészítése a `app-routing.module.ts` állományban, az alábbi módon:

```
const routes: Routes = [  
  { path: "kezdolap", component: KezdolapComponent },  
  { path: "hiba404", component: Hiba404Component },  
  { path: "jquery", component: JQueryComponent },  
  { path: "ngif", component: NgifComponent },  
  { path: "ngfor", component: NgforComponent },  
  { path: "loggolo", component: LoggoloComponent },  
  { path: "", redirectTo: "/kezdolap", pathMatch: "full" },  
  //Alapértelmezett útvonal, ha nem adnánk meg semmit...  
  { path: "**", component: Hiba404Component } //Amennyiben rossz értéket  
  adunk meg az útvonalnak  
];
```

3. Elkészítettük a navigácót az oldalra
 1. Kitöröltünk minden az `app.component.html` állományból, csak a `<router-outlet />` maradt
 2. Elkészítettük a gombokat, a komponensekhez:
`<button type="button" class="btn btn-primary m-1" routerLink="/kezdolap">Kezdőlap</button>`
`<button type="button" class="btn btn-info m-1" routerLink="/jquery">Jquery teszt</button>`

```

<button type="button" class="btn btn-success m-1" routerLink="/ngif">ngIf
funkció</button>
<button type="button" class="btn btn-warning m-1"
routerLink="/ngfor">ngFor funkció</button>
<button type="button" class="btn btn-dark m-1"
routerLink="/loggolo">Loggolás készítése</button>
<button type="button" class="btn btn-danger m-1"
routerLink="/hiba404">Hibakezelő oldal</button>

```

4. Az app.module.ts állományban, az **imports** részen belül, importáltuk a **formsmodule** komponenst.
5. Telepítettük a bootstrap-et a **npm install bootstrap** paranccsal
6. Telepítettük a jquery-t a **npm install jquery** paranccsal
7. importáltuk a komponenseket a projektbe az **angular.json** fájlban belül
 - Stílusfájlok importálása: **"styles": [**
"./node_modules/bootstrap/dist/css/bootstrap.css", "src/styles.css"]
 - Javascript állományok importálása: **"scripts":**
["./node_modules/bootstrap/dist/js/bootstrap.js", "./node_modules/jquery/dist/jquery.js"]
8. Futtattuk az eddig elkészült angular projektünket: **ng serve** paranccsal
9. A hiba 404 oldalon, létrehoztuk a hibaüzenetet: **<h1 class="text-center">Hiba 404 a keresett oldal nem található!</h1>**
10. JQuery tesztoldal elkészítése
 1. Importáljuk a dollárjelet a használathoz: **declare var \$: any;**
 2. Majd használhatjuk is a JQuery-t a kódban (Onlnit segítségével):

```

export class JQueryComponent implements OnInit {

  ngOnInit(): void {
    $(document).ready(function () {
      $("h1").click(function () {
        $("p").toggle();
      });

      console.log("JQuery is ready!");
      // Ide jöhet a jQuery kódod
    });
  }
}

```

11. ***ngIf** funkció kipróbálása:
 1. létrehozok egy változót logikai értékkel: **szovegMegjelenit: boolean = true;**
 2. Készítek egy függvényt, ami ennek a változónak az értékét módosítja!


```

megjelenitesKiBe() {
  if (this.szovegMegjelenit) { this.szovegMegjelenit = false; }
  else { this.szovegMegjelenit = true }
}

```

3. Elkészítünk egy elemet, aminek a megjelenését ki be tudom kapcsolni:
`<p>ngIf teszt</p>`
4. Készítünk egy gombot, ahova meghívom a függvényt:

```
<button type="button" class="btn btn-dark"
(click)="megjelenitesKiBe()">Megjelenit/elrejt</button>
```

5. A megjelenítendő elemhez, hozzárendelem az alábbi módon a változót:
`<p *ngIf="szovegMegjelenit">ngIf teszt</p>`

12. *ngFor funkció kipróbálása

1. Létrehozom a megjelenítendő adatszerkezetet: `bevasarloLista: string[] = ["tej", "kenyér", "herz szalámi", "alma", "sör", "maci sajt"];`
2. Készítünk egy számozatlan listát:
 - Hivatkoznom kell *ngFor-ral hogy bejárjuk a listát: `li`-tagen belül `*ngFor="let termék of bevasarloLista"`
 - Hivatkozok az *ngFor-on belül megadott listaelem értékre, az alábbi módon: `{{termék}}`

13. Loggoló funkció készítése:

1. Elkészítünk egy üres listát: `bevasarloLista: string[] = [];`
2. Csinálunk egy változót a terméknek, alpból lehet üres is: `ujTermek!: string;`
3. Készítünk egy függvényt a feltöltési folyamathoz: `termekHozzaad(): void { this.bevasarloLista.push(this.ujTermek); }`
4. Készítünk egy űrlap elemet a feltöltéshez, hozzárendeljük, two-way databindingggel az `ujTermek` elemet `[(ngModel)]`
5. Készítünk egy gombot, amihez hozzárendeljük az eventet: `<button type="button" class="btn btn-primary w-100" (click)="termekHozzaad()">Új termék hozzáadása</button>`
6. Elkészítjük az *ngFor elemet ami megjeleníti a listát:

```
<ul>
  <li *ngFor=" let termék of bevasarloLista">
    {{termék}}
  </li>
</ul>
```