

## Tartalom

<b>Mi a routingolás? .....</b>	<b>2</b>
<b>Útvonalak szerkezetének megadása .....</b>	<b>2</b>
<b>Konkrét modulra való route kialakítása .....</b>	<b>2</b>
<b>Átírányítás.....</b>	<b>2</b>
<b>404-es oldal, azaz bármilyen más eset.....</b>	<b>2</b>
<b>Elérési útvonal .....</b>	<b>3</b>
<b>Hivatkozás az oldalon .....</b>	<b>3</b>
<b>Felhasznált források.....</b>	<b>4</b>

## Mi a routingolás?

Az Angular routing rendszere lehetővé teszi, hogy az alkalmazás különböző nézetei között navigáljunk. A routing segítségével az egyoldalas alkalmazások (**SPA - Single Page Applications**) is könnyen kezelhetők, mivel az oldalak közötti navigációt a kliens oldalon valósítja meg, anélkül, hogy az egész oldal újra töltődne.

- Egy útvonal (route) egy objektum, amely URL-térképet és a hozzárendelt komponenst tartalmazza.
- Az útvonalak konfigurációját egy Routes típusú tömb tartalmazza

## Útvonalak szerkezetének megadása

Az útvonalakat, a routing modulon belül kell megadni, ami alapnak 3 féle lehet majd, mindent a Routes tömbbe kell majd beleírunk, ami egy objektumokból álló tömb.

```
const routes: Routes = [];
```

### Konkrét modulra való route kialakítása

```
const routes: Routes = [  
  { path: 'feladatok', component: FeladatokComponent },  
  { path: 'minta', component: MintaOldalakComponent }  
];
```

Itt a path paraméternél meg kell adni a route nevét, amivel szeretnénk hivatkozni rá, illetve a konkrét komponens nevét, ha jól adjuk meg a komponens beírásakor felkínál egy automatikus generálást is, az import részre ami az következőképpen fog létrejönni.

```
import { FeladatokComponent } from './feladatok/feladatok.component';  
import { MintaOldalakComponent } from './minta-oldalak/minta-oldalak.component';
```

Ha nem generálná ki a rendszer, akkor nekünk mindenképp létre kell hoznunk! Bár ez nagyon ritka.

## Átírányítás

A modulokra való átírányítás akkor szükséges, hogyha vagy nem adunk meg értéket, tehát üresen hagyjuk a modul nevét a böngészőszámban, vagy éppen egy másik néven és szeretnénk hivatkozni az elemre.

```
const routes: Routes = [  
  { path: 'feladatok', component: FeladatokComponent },  
  { path: 'minta', component: MintaOldalakComponent },  
  { path: '', redirectTo: '/minta', pathMatch: 'full' },  
];
```

Az alábbi példában, az üresen hagyott(alap) link esetén a minta oldal jelenik meg, aminek a teljes útvonalát ki is írjuk.

Fontos, a redirectTo-t csak akkor tudjuk használni, ha az az oldal már korábban meg lett hívva!

## 404-es oldal, azaz bármilyen más eset

Amennyiben szeretnénk, hogy ha az illető, aki az oldalunkra látogat, rossz címet próbál megadni, vagy épp az az oldal már nem létezik, is egy oldalra navigáljon, használjuk a '\*\*' paraméter a path megadásakor. Ekkor ugyanis, a komponesként megadott oldal lesz a célpontja, ami lehet az alap oldal (kezdőlap) vagy épp egy hibára felhívó oldal, hogy az oldal nem létezik.

```
{ path: '**', component: MintaOldalakComponent },
```

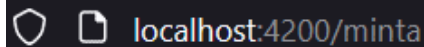
vagy

```
{ path: '**', component: NotfoundComponent }
```

## Elérési útvonal

Ha elkészült a routing „objektum tömb” akkor a benne hivatkozott elemekre egyszerűen hivatkozhatunk, csak meg kell adni az oldal címét és a végén a „/” jel után oda kell írni a nevet, amit megadtunk neki.

pl.: /minta vagy /feladatok

A screenshot of a web browser's address bar. It shows a shield icon on the left, followed by a document icon and the text 'localhost:4200/minta'.

## Hivatkozás az oldalon

Amennyiben készen vagyunk mindennel, immáron elhelyezhetünk hivatkozásokat az oldalon ehhez nem kell mást tenni, mint megadni a routerLink paramétert, és értéknek pedig a korábban a routing részben megadott path értéket, az alábbi módon

```
<a class="btn btn-primary m-2" routerLink="/minta">  
  Minta anyagok: Data-Binding  
</a>
```

## Felhasznált források

<https://angular.dev/guide/routing>

[https://dev.to/kafeel\\_ahmad/angular-router-in-depth-np4](https://dev.to/kafeel_ahmad/angular-router-in-depth-np4)

[https://en.wikipedia.org/wiki/Angular\\_\(web\\_framework\)](https://en.wikipedia.org/wiki/Angular_(web_framework))