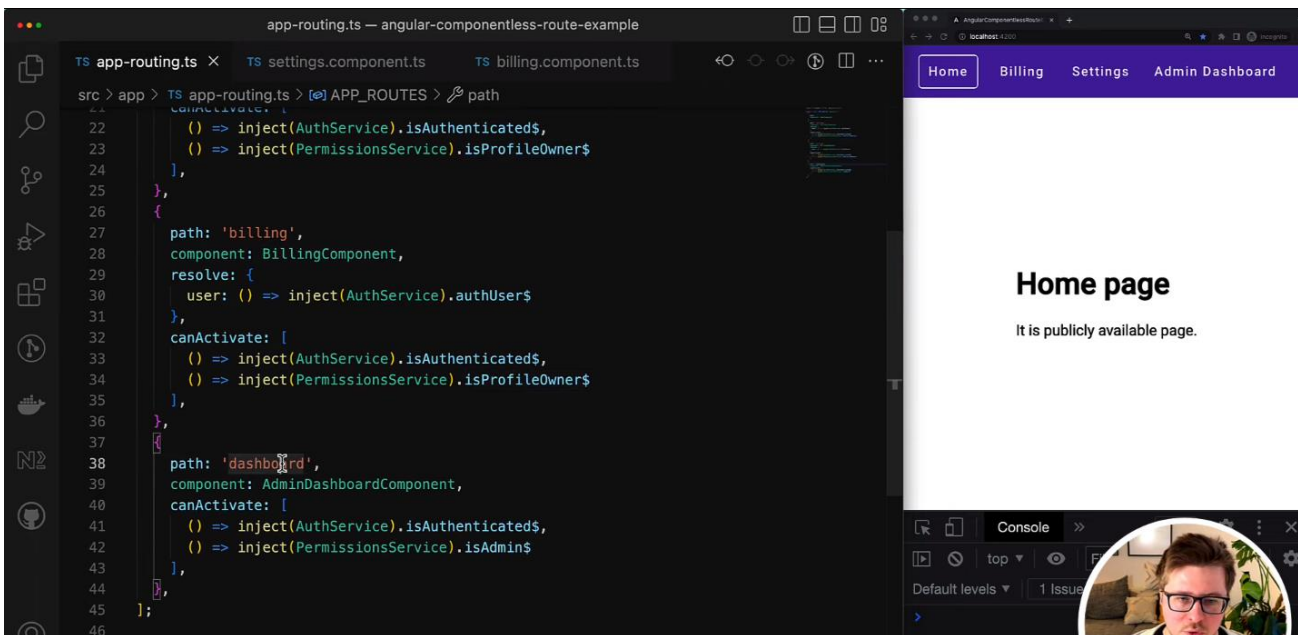
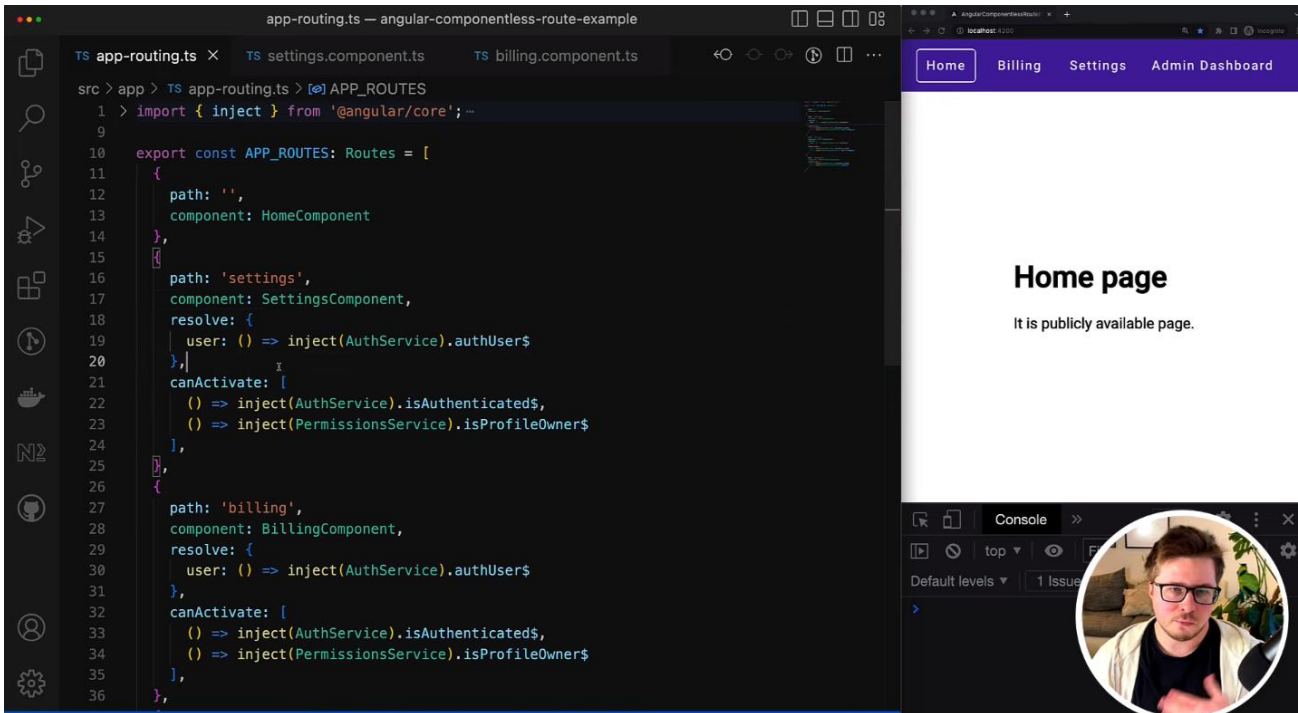




In this video, you will learn what component-less and empty-path routes in #angular are. You will see some use cases when this pattern might be helpful and help you to make your code base more reliable and flexible. I hope you will learn something new today and if so, please share this video with your colleagues and friends.



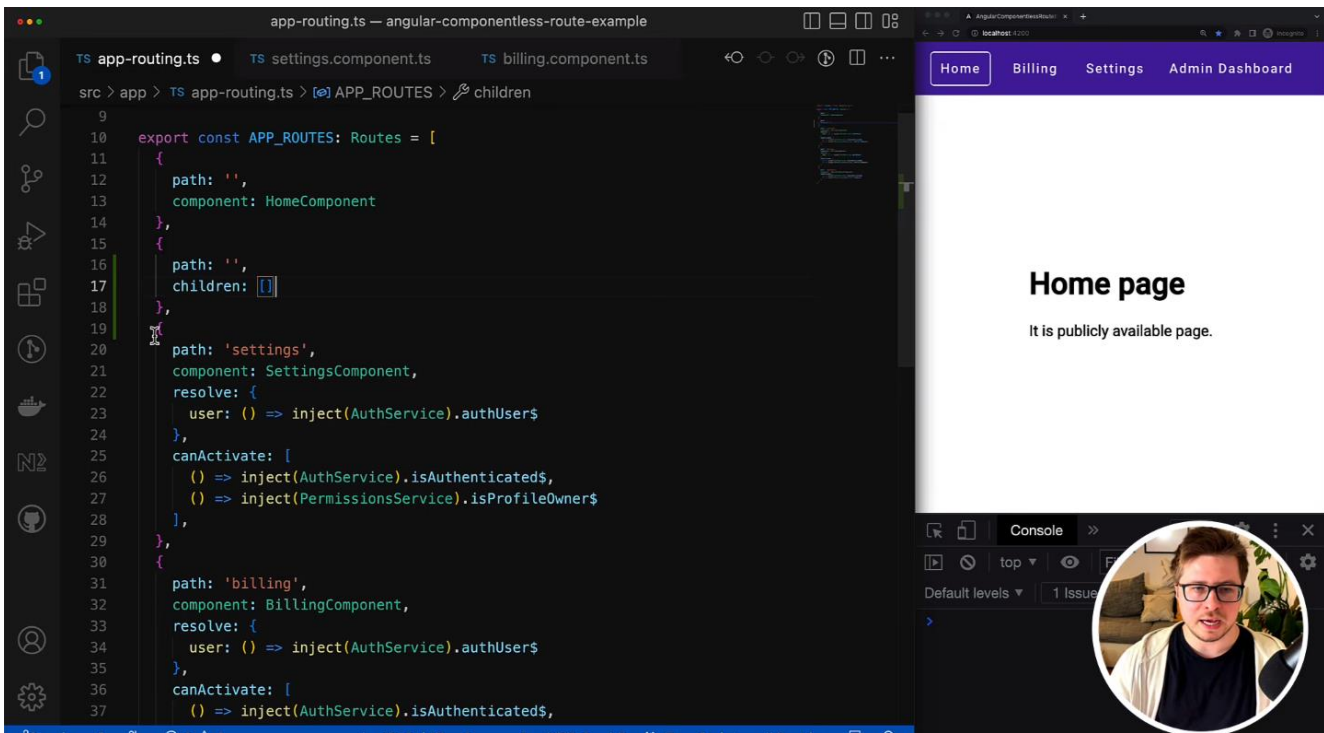
This configuration works fine but it has some disadvantages like repeated stuff like the `isAuthenticated$` guards for each route. Our resolvers will also run when loading each page and do multiple data fetching for same data.

```
src > app > pages > TS settings.component.ts > SettingsComponent > user$
1 > import { AsyncPipe, NgIf } from '@angular/common';
2
3
4
5
6 @Component({
7   selector: 'app-settings',
8   standalone: true,
9   imports: [AsyncPipe, NgIf],
10  template: `
11    <h1>Settings Page</h1>
12    <p *ngIf="(user$ | async) as user">Some User Data: <b>{{user?.email}}</b>
13    <p>
14      Only <b>authorized user</b> and <b>page owner</b> can see this page
15    </p>
16  `
17 })
18 export class SettingsComponent {
19   user$ = inject(ActivatedRoute).data.pipe(map(({ user }) => user));
20 }
21
```

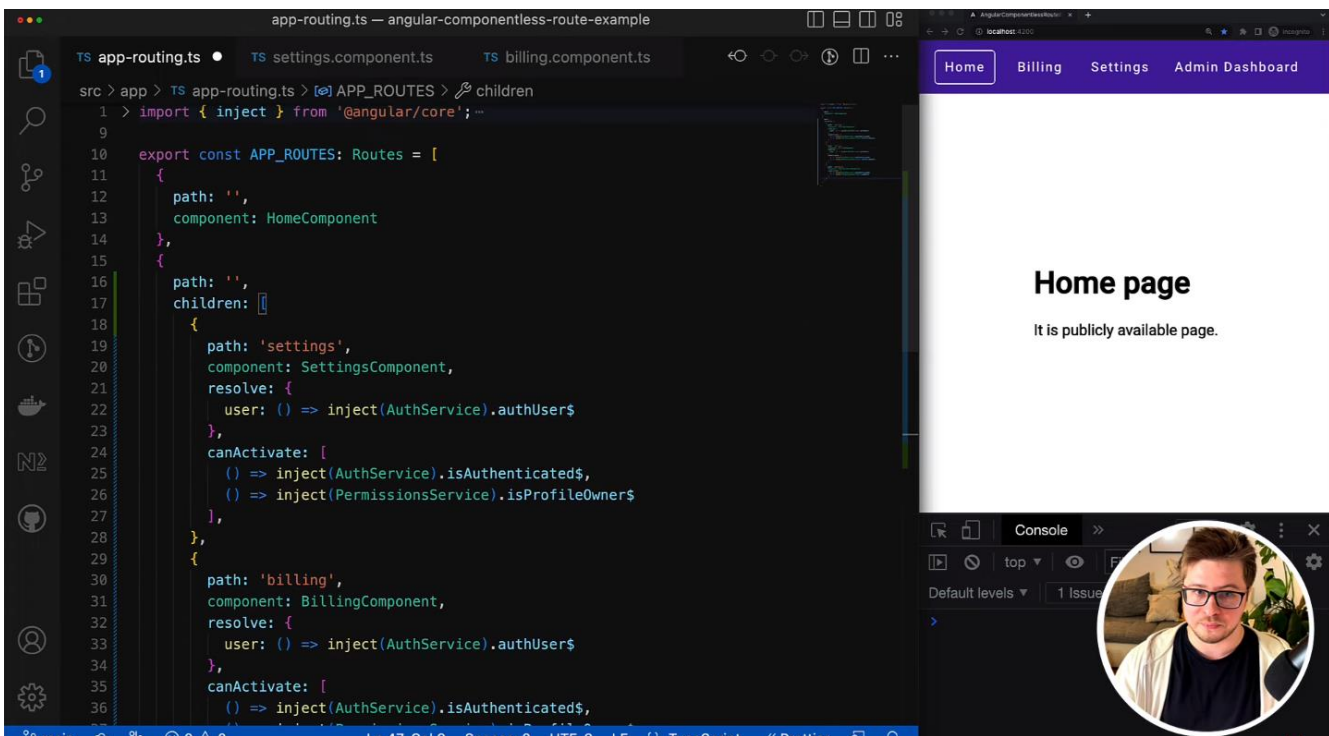
```
23   () => inject(PermissionsService).isProfileOwner$,
24 },
25 },
26 {
27   path: 'billing',
28   component: BillingComponent,
29   resolve: {
30     user: () => inject(AuthService).authUser$,
31   },
32   canActivate: [
33     () => inject(AuthService).isAuthenticated$,
34     () => inject(PermissionsService).isProfileOwner$,
35   ],
36 },
37 ],
38 },
39 ],
40 ],
41 ],
42 ],
43 ],
44 ],
45 ],
46 ],
47 ],
48 ],
49 ],
50 ],
51 ],
52 ],
53 ],
54 ],
55 ],
56 ],
57 ],
58 ],
59 ],
60 ],
61 ],
62 ],
63 ],
64 ],
65 ],
66 ],
67 ],
68 ],
69 ],
70 ],
71 ],
72 ],
73 ],
74 ],
75 ],
76 ],
77 ],
78 ],
79 ],
80 ],
81 ],
82 ],
83 ],
84 ],
85 ],
86 ],
87 ],
88 ],
89 ],
90 ],
91 ],
92 ],
93 ],
94 ],
95 ],
96 ],
97 ],
98 ],
99 ],
100 ],
```

```
src > app > TS app-routing.ts > APP_ROUTES
9
10 export const APP_ROUTES: Routes = [
11   {
12     path: '',
13     component: HomeComponent
14   },
15   {
16     path: 'settings',
17     component: SettingsComponent,
18     resolve: {
19       user: () => inject(AuthService).authUser$,
20     },
21     canActivate: [
22       () => inject(AuthService).isAuthenticated$,
23       () => inject(PermissionsService).isProfileOwner$,
24     ],
25   },
26   {
27     path: 'billing',
28     component: BillingComponent,
29     resolve: {
30       user: () => inject(AuthService).authUser$,
31     },
32     canActivate: [
33       () => inject(AuthService).isAuthenticated$,
34       () => inject(PermissionsService).isProfileOwner$,
35     ],
36   },
37 ],
38 ],
39 ],
40 ],
41 ],
42 ],
43 ],
44 ],
45 ],
46 ],
47 ],
48 ],
49 ],
50 ],
51 ],
52 ],
53 ],
54 ],
55 ],
56 ],
57 ],
58 ],
59 ],
60 ],
61 ],
62 ],
63 ],
64 ],
65 ],
66 ],
67 ],
68 ],
69 ],
70 ],
71 ],
72 ],
73 ],
74 ],
75 ],
76 ],
77 ],
78 ],
79 ],
80 ],
81 ],
82 ],
83 ],
84 ],
85 ],
86 ],
87 ],
88 ],
89 ],
90 ],
91 ],
92 ],
93 ],
94 ],
95 ],
96 ],
97 ],
98 ],
99 ],
100 ],
```

We can move the resolvers to the profile route and have it execute/load data once for all child routes. We need to create a parent/child relationship between our routes.



To create the **parent/child route relationship**, we can introduce a **new empty route** that doesn't have any component to render but instead has a property **children** where we can move all the child routes.





app-routing.ts — angular-componentless-route-example

TS app-routing.ts M X TS settings.component.ts TS billing.component.ts

src > app > TS app-routing.ts > APP\_ROUTES > children

1 > import { inject } from '@angular/core';--

9

10 export const APP\_ROUTES: Routes = [

11 {

12 path: '',

13 component: HomeComponent

14 },

15 {

16 path: '',

17 children: [

18 {

19 path: 'settings',

20 component: SettingsComponent,

21 resolve: {

22 user: () => inject(AuthService).authUser\$,

23 },

24 canActivate: [

25 () => inject(AuthService).isAuthenticated\$,

26 () => inject(PermissionsService).isProfileOwners\$,

27 ],

28 },

29 {

30 path: 'billing',

31 component: BillingComponent,

32 resolve: {

33 user: () => inject(AuthService).authUser\$,

34 },

35 canActivate: [

36 () => inject(AuthService).isAuthenticated\$,

AngularComponentlessRouteExample

Home Billing Settings Admin Dashboard

Billing page

Some User Data: Bret

Only authorized user and page owner can see this page

localhost:4200/billing

Console

Default levels 1 Issues

[webpack-dev-server] changed.

[webpack-dev-server] updated. Recompiling

[webpack-dev-server] changed.



app-routing.ts — angular-componentless-route-example

TS app-routing.ts M X TS settings.component.ts TS billing.component.ts

src > app > TS app-routing.ts > APP\_ROUTES > children

1 > import { inject } from '@angular/core';--

9

10 export const APP\_ROUTES: Routes = [

11 {

12 path: '',

13 component: HomeComponent

14 },

15 {

16 path: '',

17 children: [

18 {

19 path: 'settings',

20 component: SettingsComponent,

21 resolve: {

22 user: () => inject(AuthService).authUser\$,

23 },

24 canActivate: [

25 () => inject(AuthService).isAuthenticated\$,

26 () => inject(PermissionsService).isProfileOwners\$,

27 ],

28 },

29 {

30 path: 'billing',

31 component: BillingComponent,

32 resolve: {

33 user: () => inject(AuthService).authUser\$,

34 },

35 canActivate: [

36 () => inject(AuthService).isAuthenticated\$,

AngularComponentlessRouteExample

Home Billing Settings Admin Dashboard

Settings Page

Some User Data: Sincere@april.biz

Only authorized user and page owner can see this page

localhost:4200/billing

Console

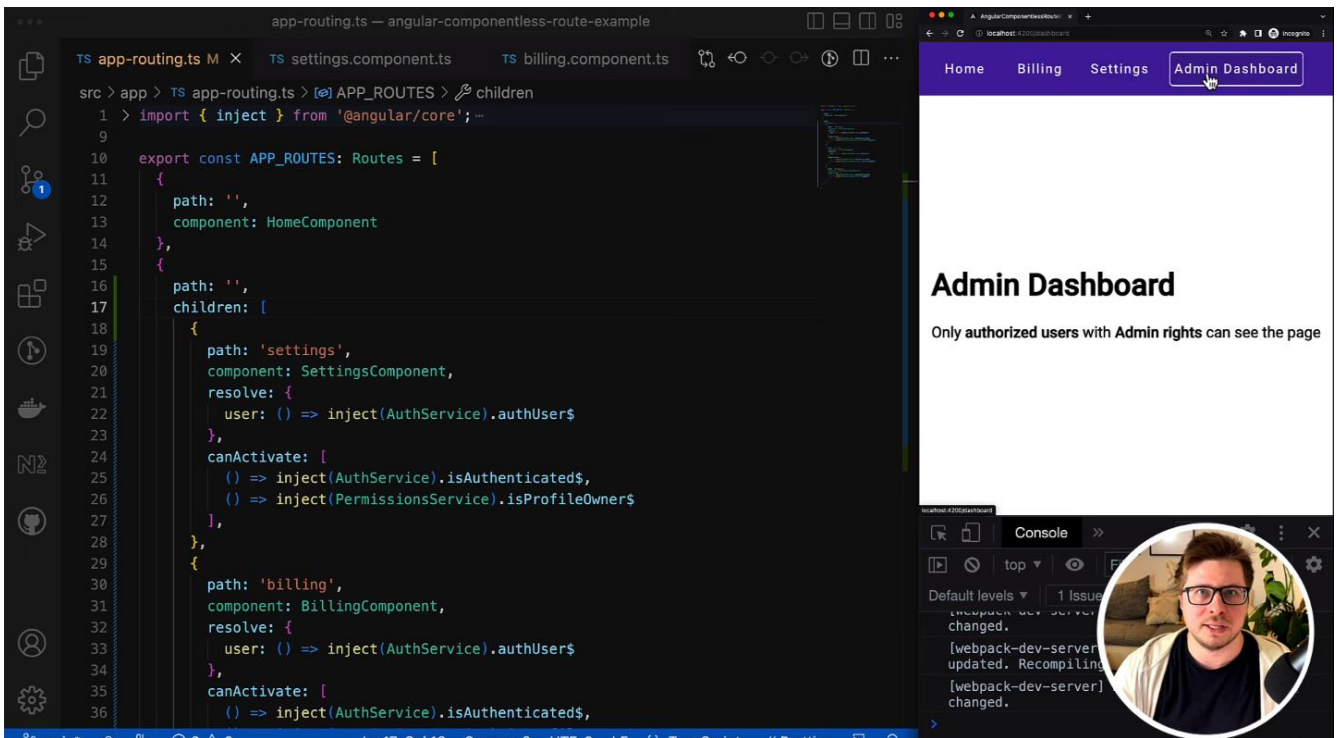
Default levels 1 Issues

[webpack-dev-server] changed.

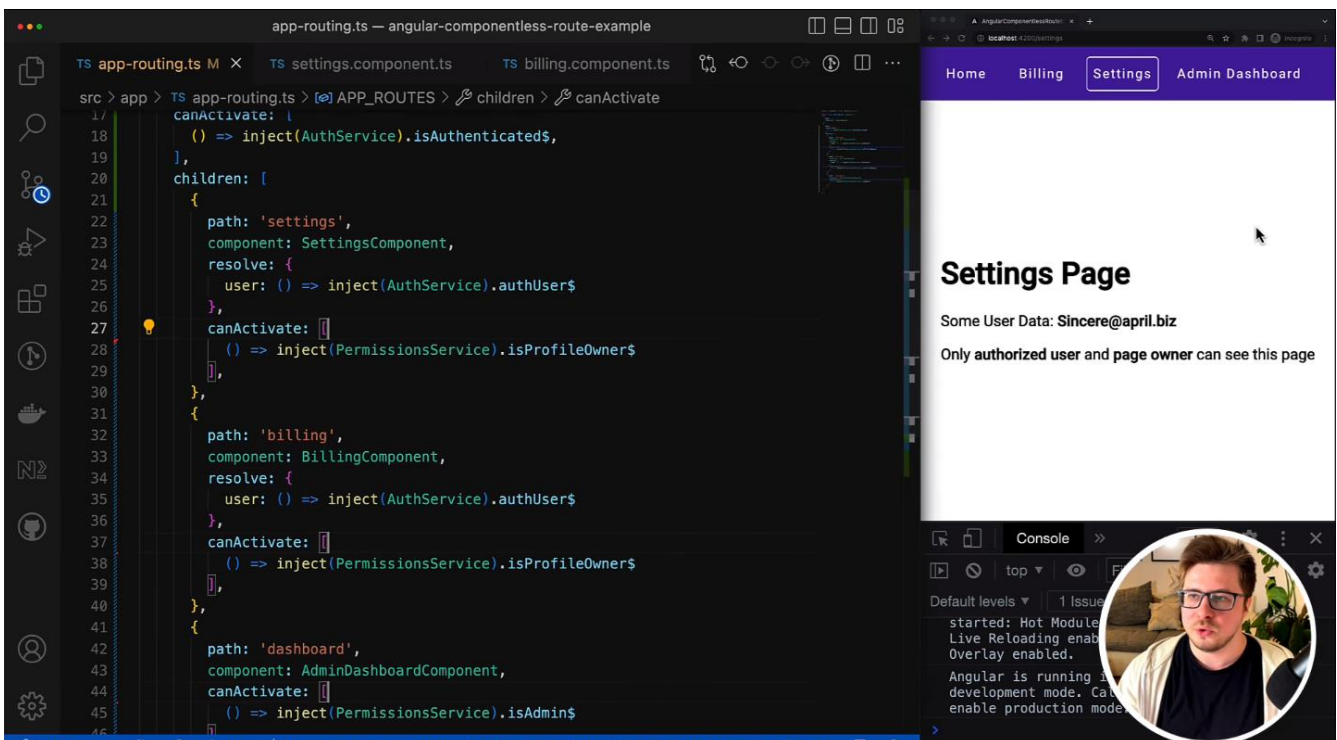
[webpack-dev-server] updated. Recompiling

[webpack-dev-server] changed.





The app still works fine even though we have introduced a new parent route, this is because the parent route has an empty `path: ''` path.



We can now move common children route guards up to their parent route.

The screenshot shows a development environment with VS Code on the left and a web browser on the right. In VS Code, the file `auth.service.ts` is open, showing the `AuthService` class. The `isAuthenticated$` property is initialized to `true`. The browser displays the 'Admin Dashboard' with a message: 'Only authorized users with Admin rights can see the page'. The browser's console shows messages about Hot Module Replacement and Angular's development mode.

```
src > app > auth > TS auth.service.ts > AuthService
1 > import { HttpClient } from '@angular/common/http';
4
5
6 @Injectable({
7   providedIn: 'root'
8 })
9 export class AuthService {
10   isAuthenticated$ = new BehaviorSubject(true);
11
12   authUsers$ = inject(HttpClient).get('https://jsonplaceholder.typicode.com/user
13 }
```

Admin Dashboard

Only authorized users with Admin rights can see the page

Console

Default levels ▾ 1 Issue

started: Hot Module Replacement

Live Reloading enabled

Overlay enabled.

Angular is running in development mode. Call enable production mode

The screenshot shows the same development environment as the previous one. In VS Code, the `isAuthenticated$` property in `AuthService` is now initialized to `false`. The browser displays the 'Home page' with a message: 'It is publicly available page.' The browser's console shows the same messages as before.

```
src > app > auth > TS auth.service.ts > AuthService > isAuthenticated$
1 > import { HttpClient } from '@angular/common/http';
4
5
6 @Injectable({
7   providedIn: 'root'
8 })
9 export class AuthService {
10   isAuthenticated$ = new BehaviorSubject(false);
11
12   authUsers$ = inject(HttpClient).get('https://jsonplaceholder.typicode.com/user
13 }
```

Home

Home page

It is publicly available page.

Console

Default levels ▾ 1 Issue

started: Hot Module Replacement

Live Reloading enabled

Overlay enabled.

Angular is running in development mode. Call enable production mode

We can now set the default authentication state to false and see that only the Home page is now accessible because it does not have the `isAuthenticated$` route set on it.

auth.service.ts — angular-componentless-route-example

TS app-routing.ts M TS auth.service.ts X TS settings.component.ts TS bill


```
src > app > auth > TS auth.service.ts > AuthService > isAuthenticated$
1 > import { HttpClient } from '@angular/common/http';
4
5 @Injectable({
6   providedIn: 'root'
7 })
8 export class AuthService {
9   isAuthenticated$ = new BehaviorSubject(false);
10
11   authUsers$ = inject(HttpClient).get('https://jsonplaceholder.typicode.com/user
12 }
13
```

Home Billing Settings Admin Dashboard

Console

started: Hot Module Live Reloading enabled. Overlay enabled.

Angular is running in development mode. Call enable production mode



auth.service.ts — angular-componentless-route-example

TS app-routing.ts M TS auth.service.ts X TS settings.component.ts TS bill


```
src > app > auth > TS auth.service.ts > AuthService > isAuthenticated$
1 > import { HttpClient } from '@angular/common/http';
4
5 @Injectable({
6   providedIn: 'root'
7 })
8 export class AuthService {
9   isAuthenticated$ = new BehaviorSubject(false);
10
11   authUsers$ = inject(HttpClient).get('https://jsonplaceholder.typicode.com/user
12 }
13
```

Home Billing Settings Admin Dashboard

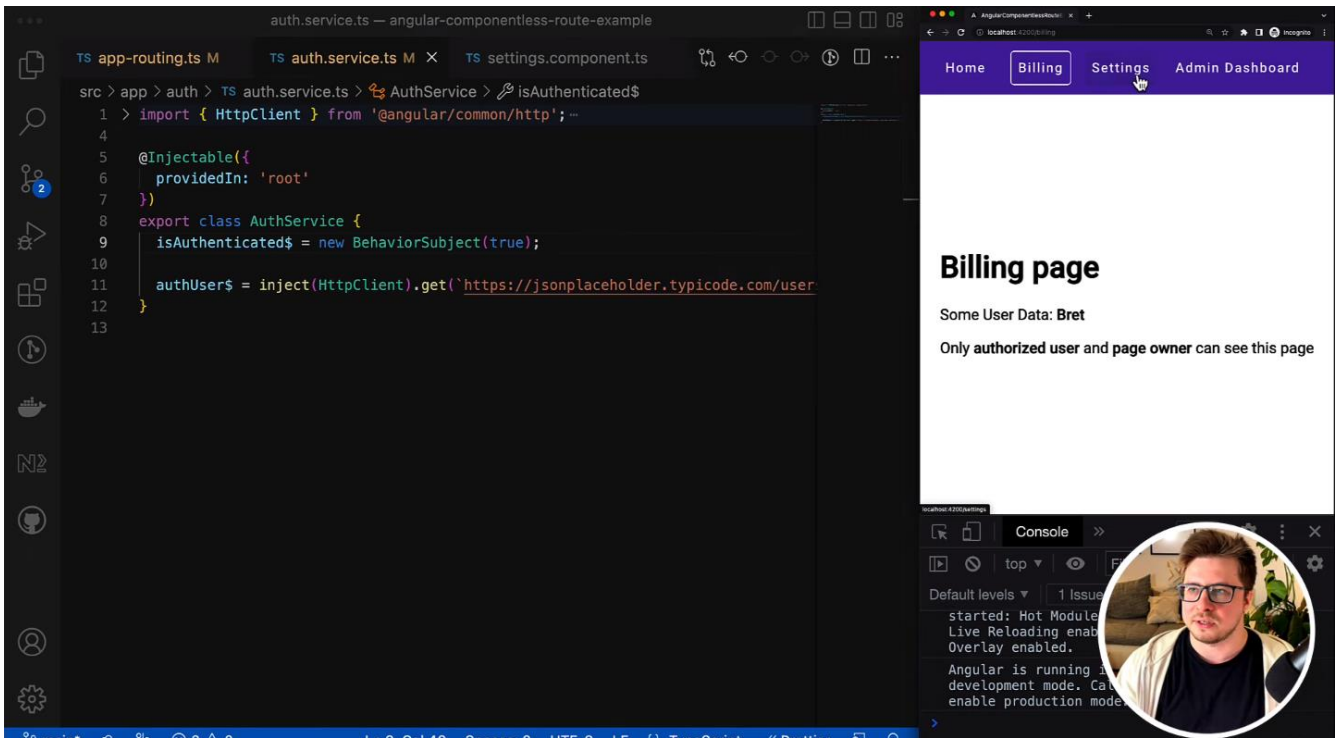
Console

started: Hot Module Live Reloading enabled. Overlay enabled.

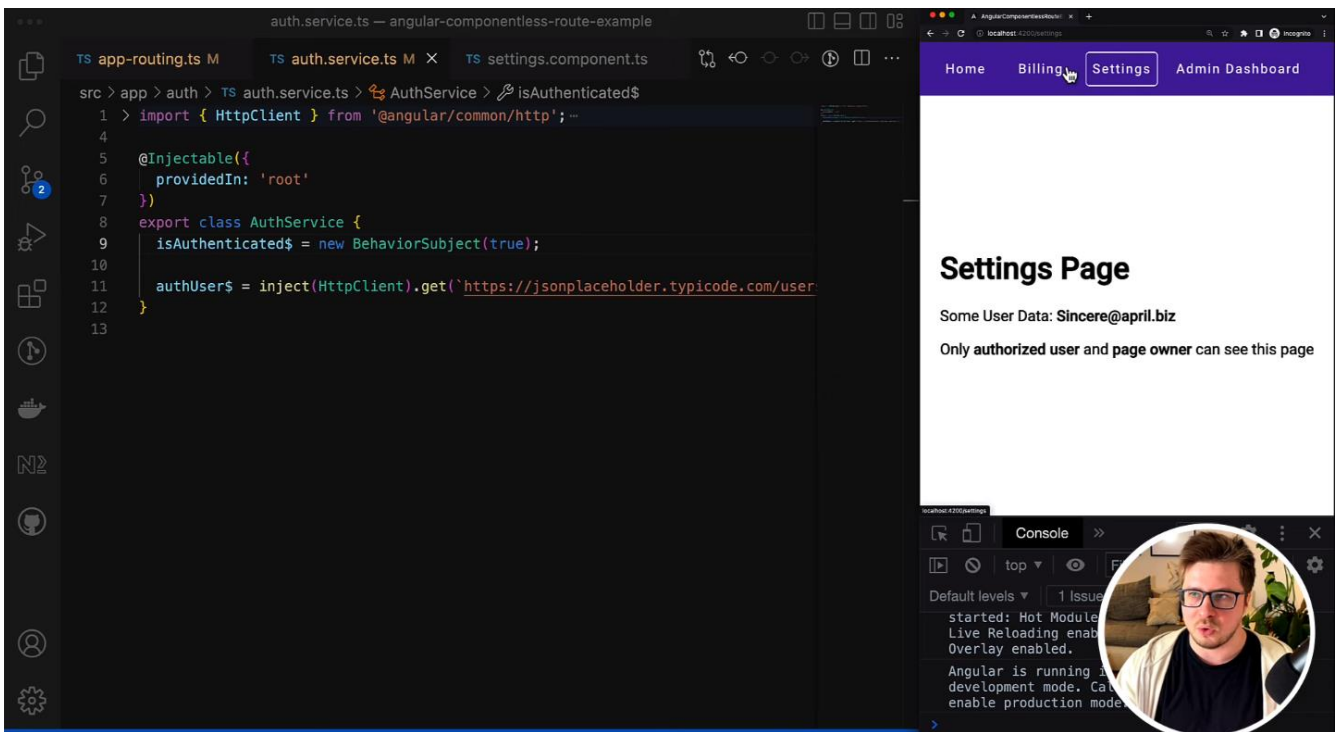
Angular is running in development mode. Call enable production mode



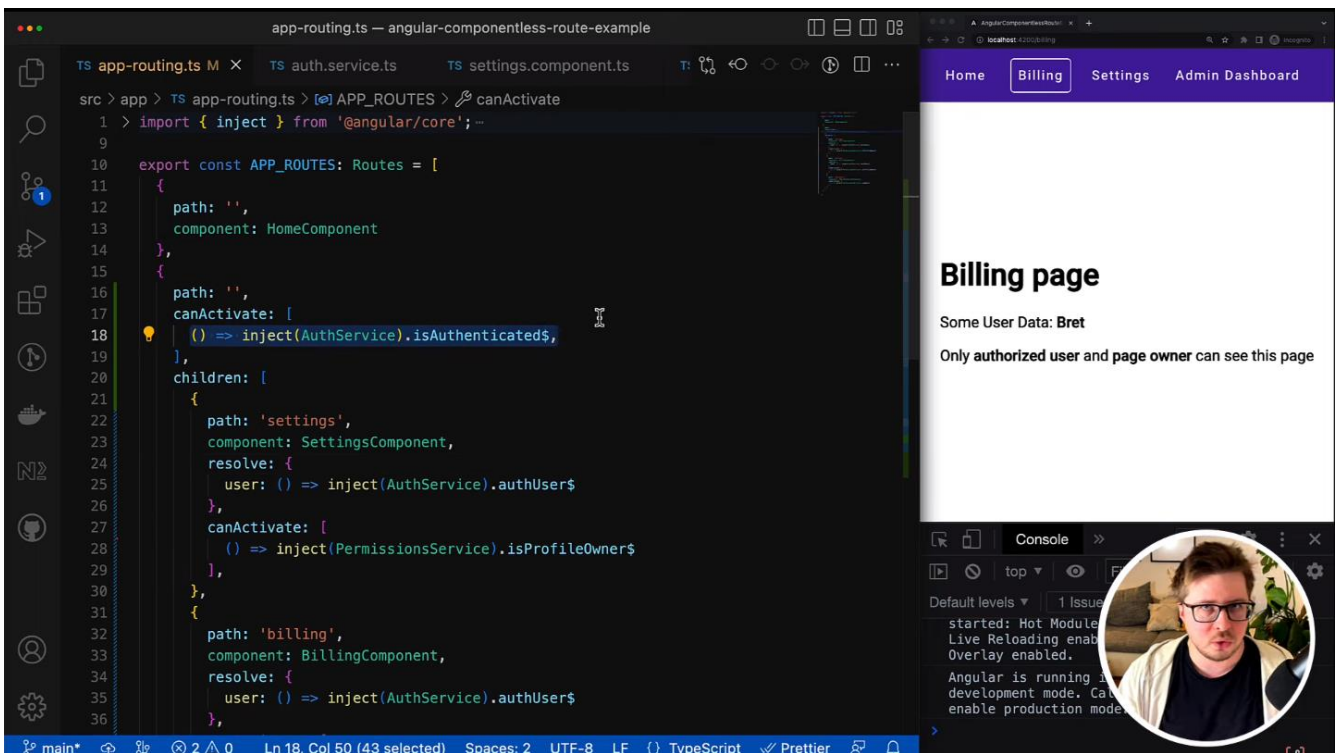
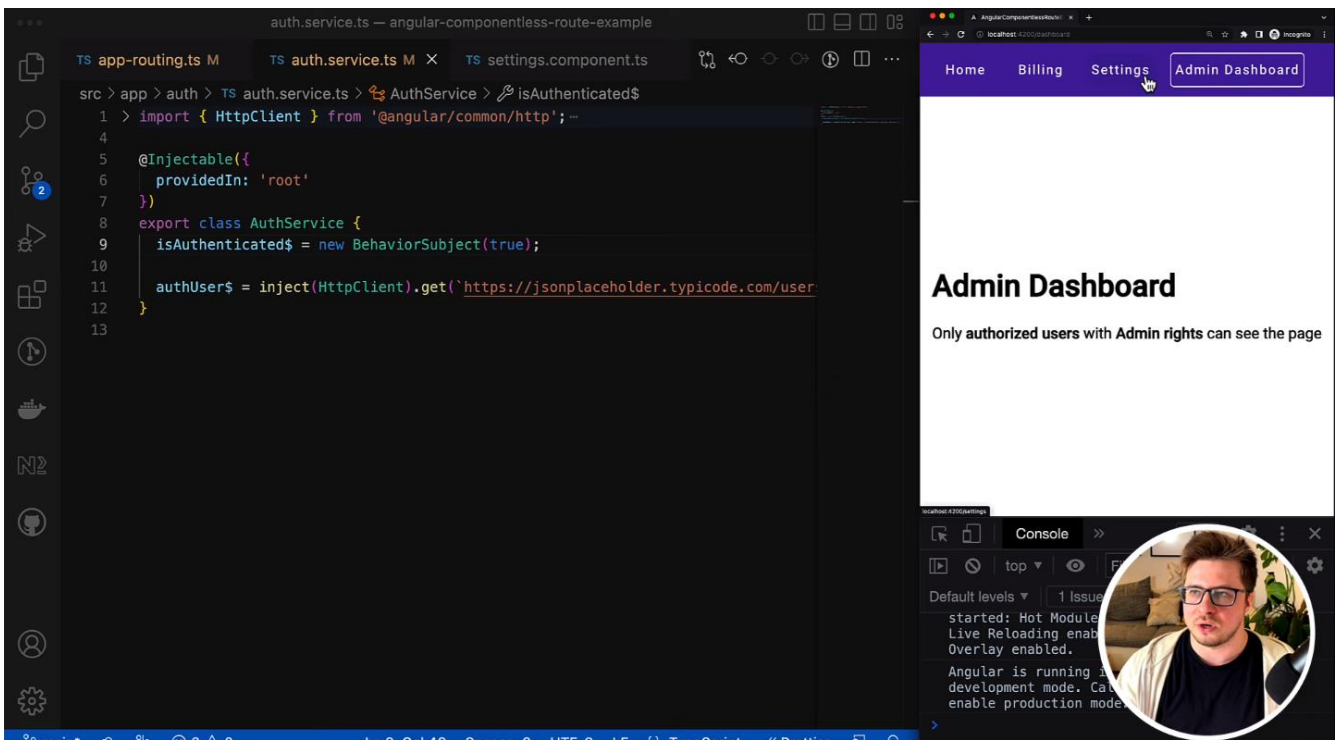




When we set the default authentication state to true, we can now access the billing, settings and admin pages that require authenticated state.







The guard will not be re-executed when we navigate between child routes of the parent route like going from the settings to the billing page.

app-routing.ts — angular-componentless-route-example

TS app-routing.ts M X

TS auth.service.ts

TS settings.component.ts

T: ↶ ↷ ↺ ↻

src > app > TS app-routing.ts > [APP\_ROUTES] > canActivate

```
1 > import { inject } from '@angular/core';
9
10 export const APP_ROUTES: Routes = [
11   {
12     path: '',
13     component: HomeComponent
14   },
15   {
16     path: '',
17     canActivate: [
18       () => {
19         console.log('canActivate guard was executed...');
20         return inject(AuthService).isAuthenticated$
21       },
22     ],
23     children: [
24       {
25         path: 'settings',
26         component: SettingsComponent,
27         resolve: {
28           user: () => inject(AuthService).authUser$
29         },
30         canActivate: [
31           () => inject(PermissionsService).isProfileOwners$
32         ],
33       },
34       {
35         path: 'billing',
36         component: BillingComponent,
```

Ln 18, Col 14

Spaces: 2

UTF-8

LF

() TypeScript

✓ Prettier

AngularComponentlessRouteExample

Home Billing Settings Admin Dashboard

Billing page

Some User Data: Bret

Only authorized user and page owner can see this page

Console

Angular is running in development mode. Click here to enable production mode.

canActivate guard was executed...

app-routing.ts — angular-componentless-route-example

TS app-routing.ts M X

TS auth.service.ts

TS settings.component.ts

T: ↶ ↷ ↺ ↻

src > app > TS app-routing.ts > [APP\_ROUTES] > canActivate

```
1 > import { inject } from '@angular/core';
9
10 export const APP_ROUTES: Routes = [
11   {
12     path: '',
13     component: HomeComponent
14   },
15   {
16     path: '',
17     canActivate: [
18       () => {
19         console.log('canActivate guard was executed...');
20         return inject(AuthService).isAuthenticated$
21       },
22     ],
23     children: [
24       {
25         path: 'settings',
26         component: SettingsComponent,
27         resolve: {
28           user: () => inject(AuthService).authUser$
29         },
30         canActivate: [
31           () => inject(PermissionsService).isProfileOwners$
32         ],
33       },
34       {
35         path: 'billing',
36         component: BillingComponent,
```

Ln 18, Col 14

Spaces: 2

UTF-8

LF

() TypeScript

✓ Prettier

AngularComponentlessRouteExample

Home Billing Settings Admin Dashboard

Settings Page

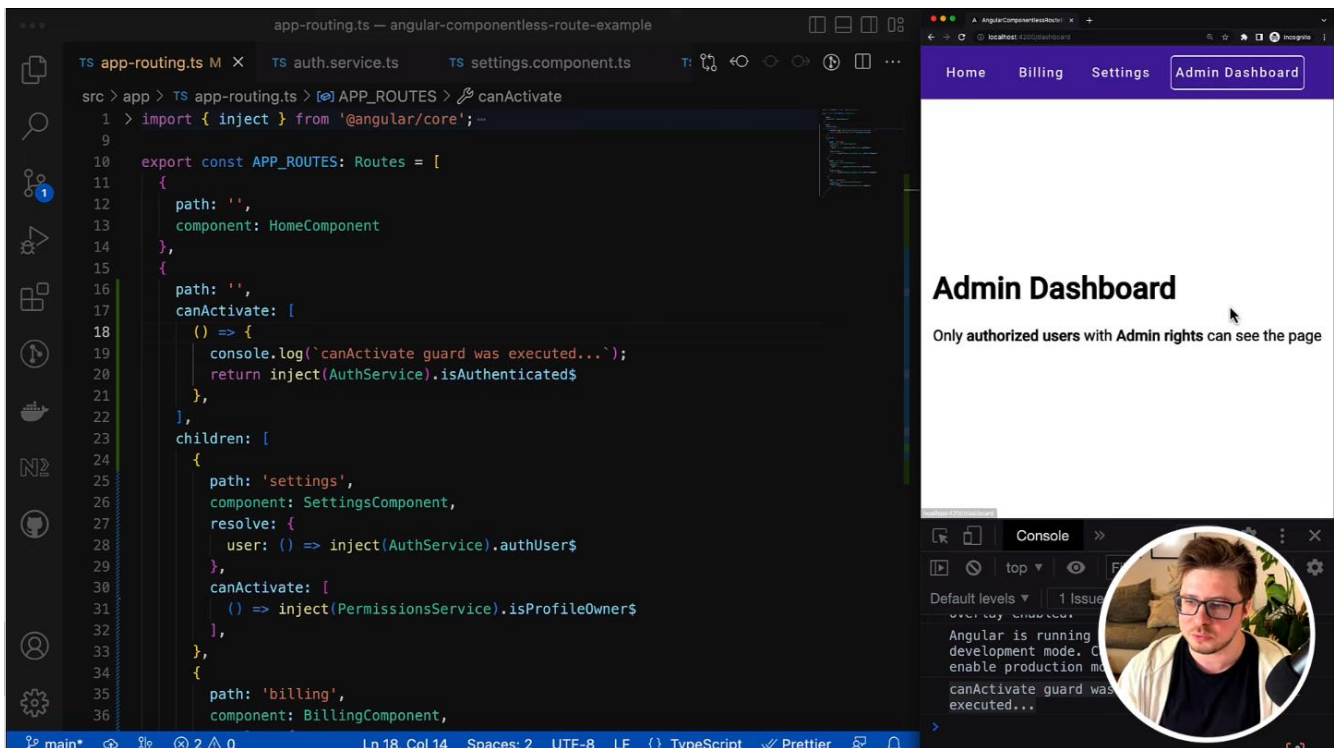
Some User Data: Sincere@april.biz

Only authorized user and page owner can see this page

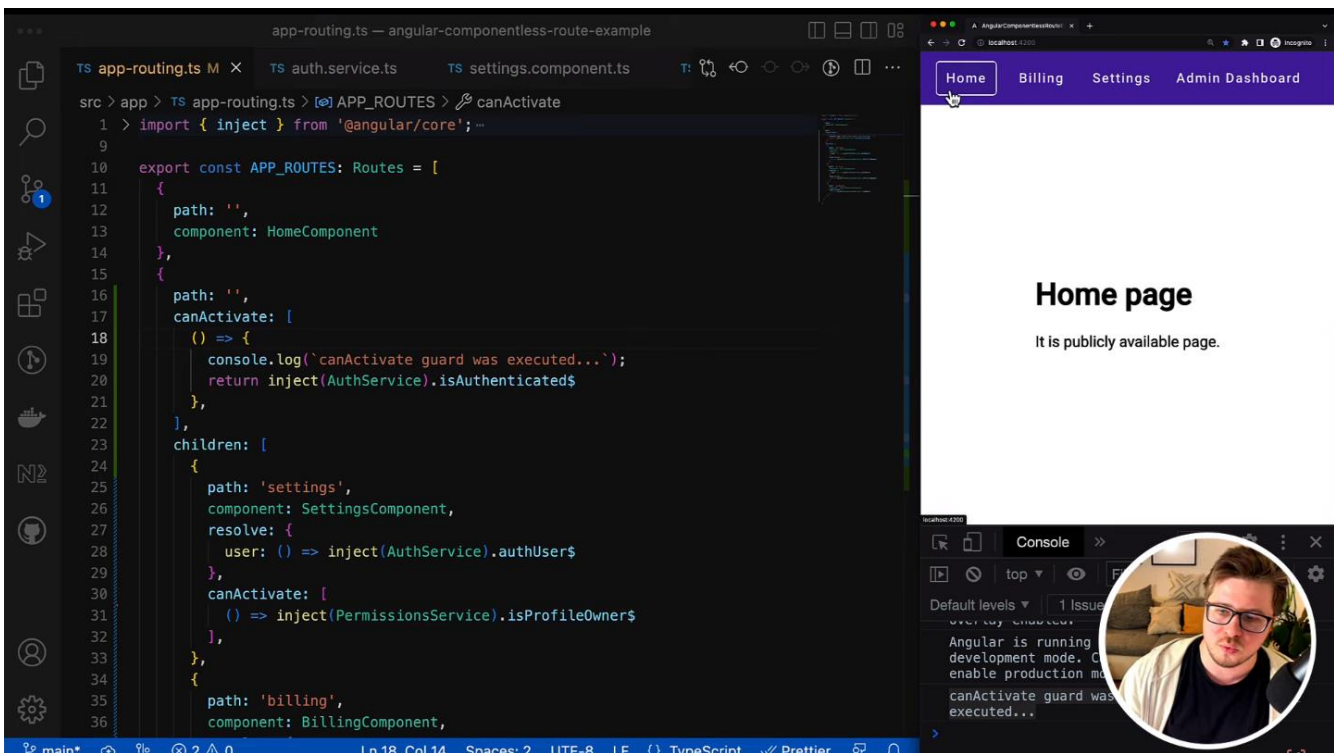
Console

Angular is running in development mode. Click here to enable production mode.

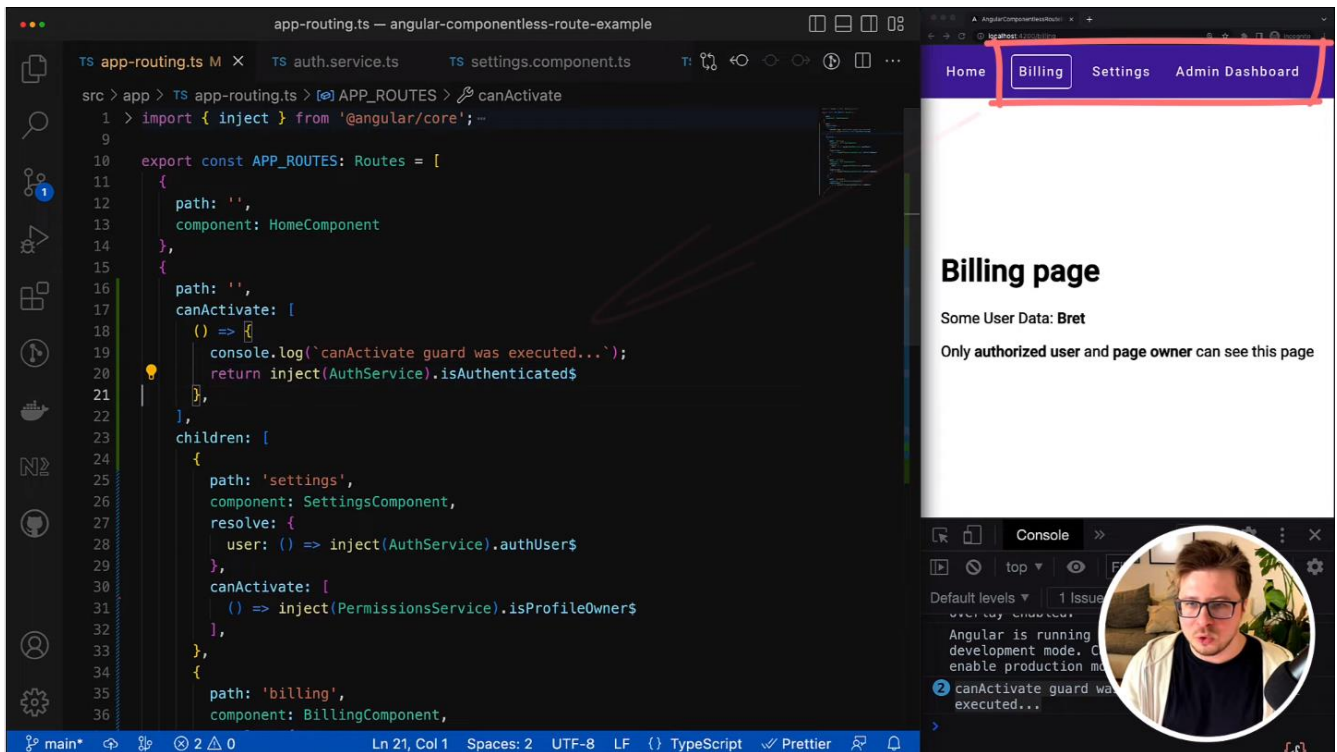
canActivate guard was executed...



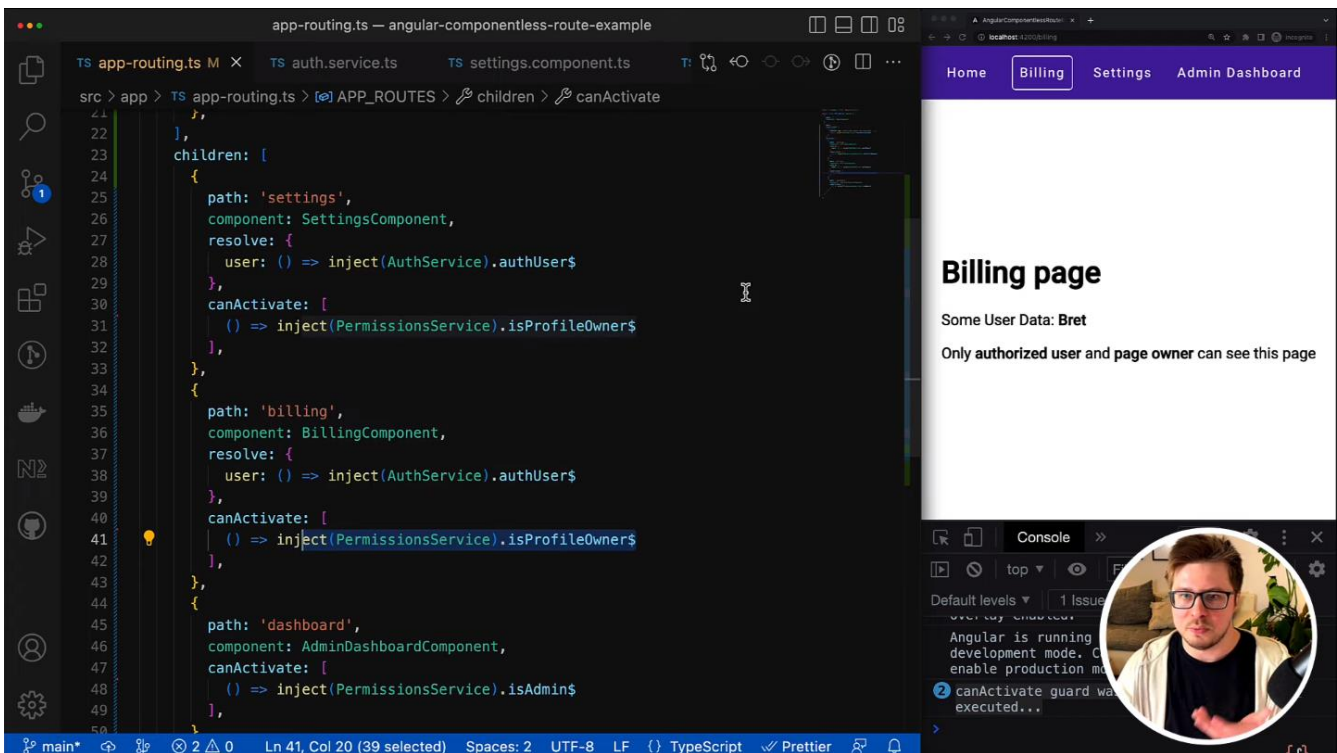
Nothing happens again







It was executed from Home page to Billing page



Recall also that the **settings** and **billing** routes have common data, guards and resolvers, we can create a higher-level nested component route for common things between them as below.



app-routing.ts — angular-componentless-route-example

TS app-routing.ts M • TS auth.service.ts TS settings.component.ts

src > app > TS app-routing.ts > APP\_ROUTES > children > children

21

},

22

children: [

23

{

24

path: '',

25

children: []

26

,

27

{

28

path: 'settings',

29

component: SettingsComponent,

30

resolve: {

31

user: () => inject(AuthService).authUsers\$

32

},

33

canActivate: [

34

() => inject(PermissionsService).isProfileOwners\$

35

],

36

},

37

{

38

path: 'billing',

39

component: BillingComponent,

40

resolve: {

41

user: () => inject(AuthService).authUsers\$

42

},

43

canActivate: [

44

() => inject(PermissionsService).isProfileOwners\$

45

],

46

},

47

{

48

path: 'dashboard',

49

component: AdminDashboardComponent,

50

},

51

],

52

},

53

],

54

},

55

},

56

},

57

},

58

},

59

},

60

},

61

},

62

},

63

},

64

},

65

},

66

},

67

},

68

},

69

},

70

},

71

},

72

},

73

},

74

},

75

},

76

},

77

},

78

},

79

},

80

},

81

},

82

},

83

},

84

},

85

},

86

},

87

},

88

},

89

},

90

},

91

},

92

},

93

},

94

},

95

},

96

},

97

},

98

},

99

},

100

},

101

},

102

},

103

},

104

},

105

},

106

},

107

},

108

},

109

},

110

},

111

},

112

},

113

},

114

},

115

},

116

},

117

},

118

},

119

},

120

},

121

},

122

},

123

},

124

},

125

},

126

},

127

},

128

},

129

},

130

},

131

},

132

},

133

},

134

},

135

},

136

},

137

},

138

},

139

},

140

},

141

},

142

},

143

},

144

},

145

},

146

},

147

},

148

},

149

},

150

},

151

},

152

},

153

},

154

},

155

},

156

},

157

},

158

},

159

},

160

},

161

},

162

},

163

},

164

},

165

},

166

},

167

},

168

},

169

},

170

},

171

},

172

},

173

},

174

},

175

},

176

},

177

},

178

},

179

},

180

},

181

},

182

},

183

},

184

},

185

},

186

},

187

},

188

},

189

},

190

},

191

},

192

},

193

},

194

},

195

},

196

},

197

},

198

},

199

},

200

},

201

},

202

},

203

},

204

},

205

},

206

},

207

},

208

},

209

},

210

},

211

},

212

},

213

},

214

},

215

},

216

},

217

},

218

},

219

},

220

},

221

},

222

},

223

},

224

},

225

},

226

},

227

},

228

},

229

},

230

},

231

},

232

},

233

},

234

},

235

},

236

},

237

},

238

},

239

},

240

},

241

},

242

},

243

},

244

},

245

},

246

},

247

},

248

},

249

},

250

},

251

},

252

},

253

},

254

},

255

},

256

},

257

},

258

},

259

},

260

},

261

},

262

},

263

},

264

},

265

},

266

},

267

},

268

},

269

},

270

},

271

},

272

},

273

},

274

},

275

},

276

},

277

},

278

},

279

},

280

},

281

},

282

},

283

},

284

},

285

},

286

},

287

},

288

},

289

},

290

},

291

},

292

},

293

},

294

},

295

},

296

},

297

},

298

},

299

},

300

},

301

},

302

},

303

},

304

},

305

},

306

},

307

},

308

},

309

},

310

},

311

},

312

},

313

},

314

},

315

},

316

},

317

},

318

},

319

},

320

},

321

},

322

},

323

},

324

},

325

},

326

},

327

},

328

},

329

},

330

},

331

},

332

},

333

},

334

},

335

},

336

},

337

},

338

},

339

},

340

},

341

},

342

},

343

},

344

},

345

},

346

},

347

},

348

},

349

},

350

},

351

},

352

},

353

},

354

},

355

},

356

},

357

},

358

},

359

},

360

},

361

},

362

},

363

},

364

},

365

},

366

},

367

},

368

},

369

},

370

},

371

},

372

},

373

},

374

},

375

},

376

},

377

},

378

},

379

},

380

},

381

},

382

},

383

},

384

},

385

},

386

},

387

},

388

},

389

},

390

},

391

},

392

},

393

},

394

},

395

},

396

},

397

},

398

},

399

},

400

},

401

},

402

},

403

},

404

},

405

},

406

},

407

},

408

},

409

},

410

},

411

},

412

},

413

},

414

},

415

},

416

},

417

},

418

},

419

},

420

},

421

},

422

},

423

},

424

},

425

},

426

},

427

},

428

},

429

},

430

},

431

},

432

},

433

},

434

},

435

},

436

},

437

},

438

},

439

},

440

},

441

},

442

},

443

},

444

},

445

},

446

},

447

},

448

},

449

},

450

},

451

},

452

},

453

},

454

},

455

},

456

},

457

},

458

},

459

},

460

},

461

},

462

},

463

},

464

},

465

},

466

},

467

},

468

},

469

},

470

},

471

},

472

},

473

},

474

},

475

},

476

},

477

},

478

},

479

},

480

},

481

},

482

},

483

},

484

},

485

},

486

},

487

},

488

},

489

},

490

},

491

},

492

},

493

},

494

},

495

},

496

},

497

},

498

},

499

},

500

},

501

},

502

},

503

},

504

},

505

},

506

},

507

},

508

},

509

},

510

},

511

},

512

},

513

},

514

},

515

},

516

},

517

},

518

},

519

},

520

},

521

},

522

},

523

},

524

},

525

},

526

},

527

},

528

},

529

},

530

},

531

},

532

},

533

},

534

},

535

},

536

},

537

},

538

},

539

},

540

},

541

},

542

},

543

},

544

},

545

},

546

},

547

},

548

},

549

},

550

},

551

},

552

},

553

},

554

},

555

},

556

},

557

},

558

},

559

},

560

},

561

},

562

},

563

},

564

},

565

},

566

},

567

},

568

},

569

},

570

},

571

},

572

},

573

},

574

},

575

},

576

},

577

},

578

},

579

},

580

},

581

},

582

},

583

},

584

},

585

},

586

},

587

},

588

},

589

},

590

},

591

},

592

},

593

},

594

},

595

},

596

},

597

},

598

},

599

},

600

},

601

},

602

},

603

},

604

},

605

},

606

},

607

},

608

},

609

},

610

},

611

},

612

},

613

},

614

},

615

},

616

},

617

},

618

},

619

},

620

},

621

},

622

},

623

},

624

},

625

},

626

},

627

},

628

},

629

},

630

},

631

},

632

},

633

},

634

},

635

},

636

},

637

},

638

},

639

},

640

},

641

},

642

},

643

},

644

},

645

},

646

},

647

},

648

},

649

},

650

},

651

},

652

},

653

},

654

},

655

},

656

},

657

},

658

},

659

},

660

},

661

},

662

},

663

},

664

},

665

},

666

},

667

},

668

},

669

},

670

},

671

},

672

},

673

},

674

},

675

},

676

},

677

},

678

},

679

},

680

},

681

},

682

},

683

},

684

},

685

},

686

},

687

},

688

},

689

},

690

},

691

},

692

},

693

},

694

},

695

},

696

},

697

},

698

},

699

},

700

},

701

},

702

},

703

},

704

},

705

},

706

},

707

},

708

},

709

},

710

},

711

},

712

},

713

},

714

},

715

},

716

},

717

},

718

},

719

},

720

},

721

},

722

},

723

},

724

},

725

},

726

},

727

},

728

},

729

},

730

},

731

},

732

},

733

},

734

},

735

},

736

},

737

},

738

},

739

},

740

},

741

},

742

},

743

},

744

},

745

},

746

},

747

},

748

},

749

},

750

},

751

},

752

},

753

},

754

},

755

},

756

},

757

},

758

},

759

},

760

},

761

},

762

},

763

},

764

},

765

},

766

},

767

},

768

},

769

},

770

},

771

},

772

},

773

},

774

},

775

},

776

},

777

},

778

},

779

},

780

},

781

},

782

},

783

},

784

},

785

},

786

},

787

},

788

},

789

},

790

},

791

},

792

},

793

},

794

},

795

},

796

},

797

},

798

},

799

},

800

},

801

},

802

},

803

},

804

},

805

},

806

},

807

},

808

},

809

},

810

},

811

},

812

},

813

},

814

},

815

},

816

},

817

},

818

},

819

},

820

},

821

},

822

},

823

},

824

},

825

},

826

},

827

},

828

},

829

},

830

},

831

},

832

},

833

},

834

},

835

},

836

},

837

},

838

},

839

},

840

},

841

},

842

},

843

},

844

},

845

},

846

},

847

},

848

},

849

},

850

},

851

},

852

},

853

},

854

},

855

},

856

},

857

},

858

},

859

},

860

},

861

},

862

},

863

},

864

},

865

},

866

},

867

},

868

},

869

},

870

},

871

},

872

},

873

},

874

},

875

},

876

},

877

},

878

},

879

},

880

},

881

},

882

},

883

},

884

},

885

},

886

},

887

},

888

},

889

},

890

},

891

},

892

},

893

},

894

},

895

},

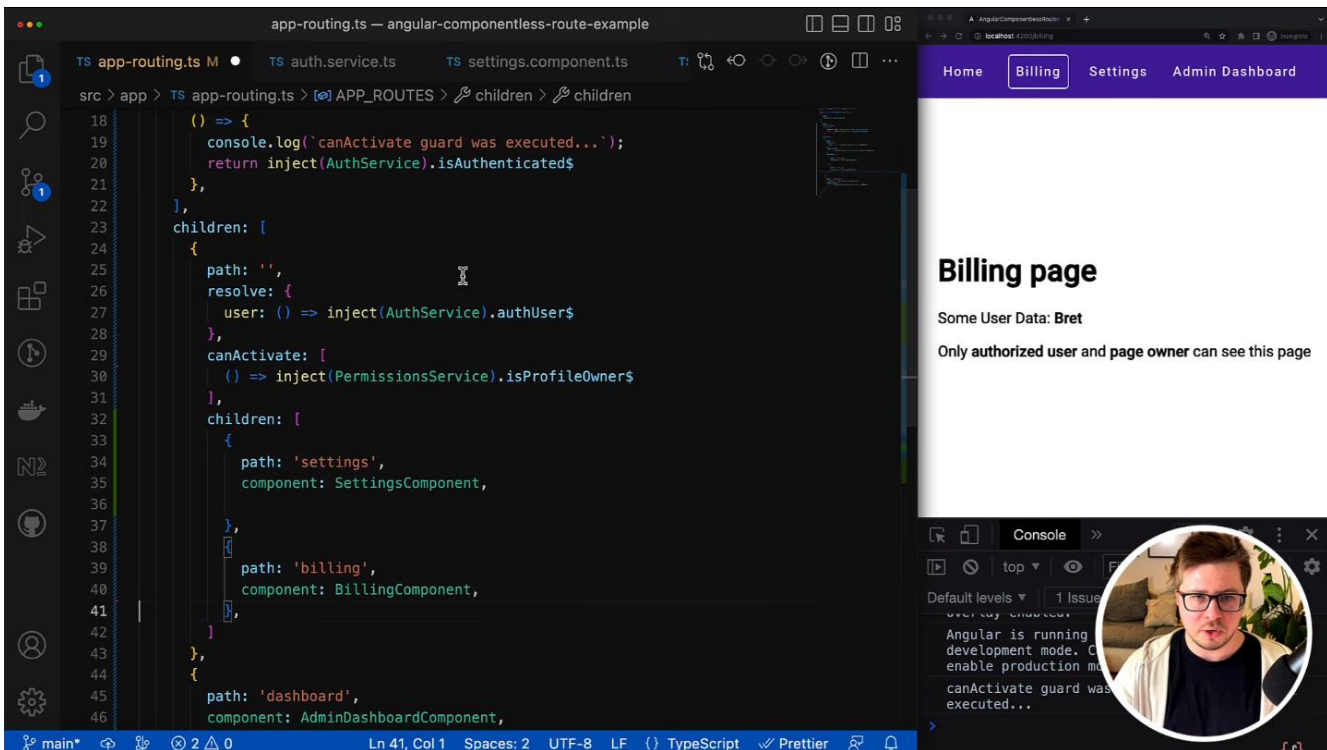
896

},

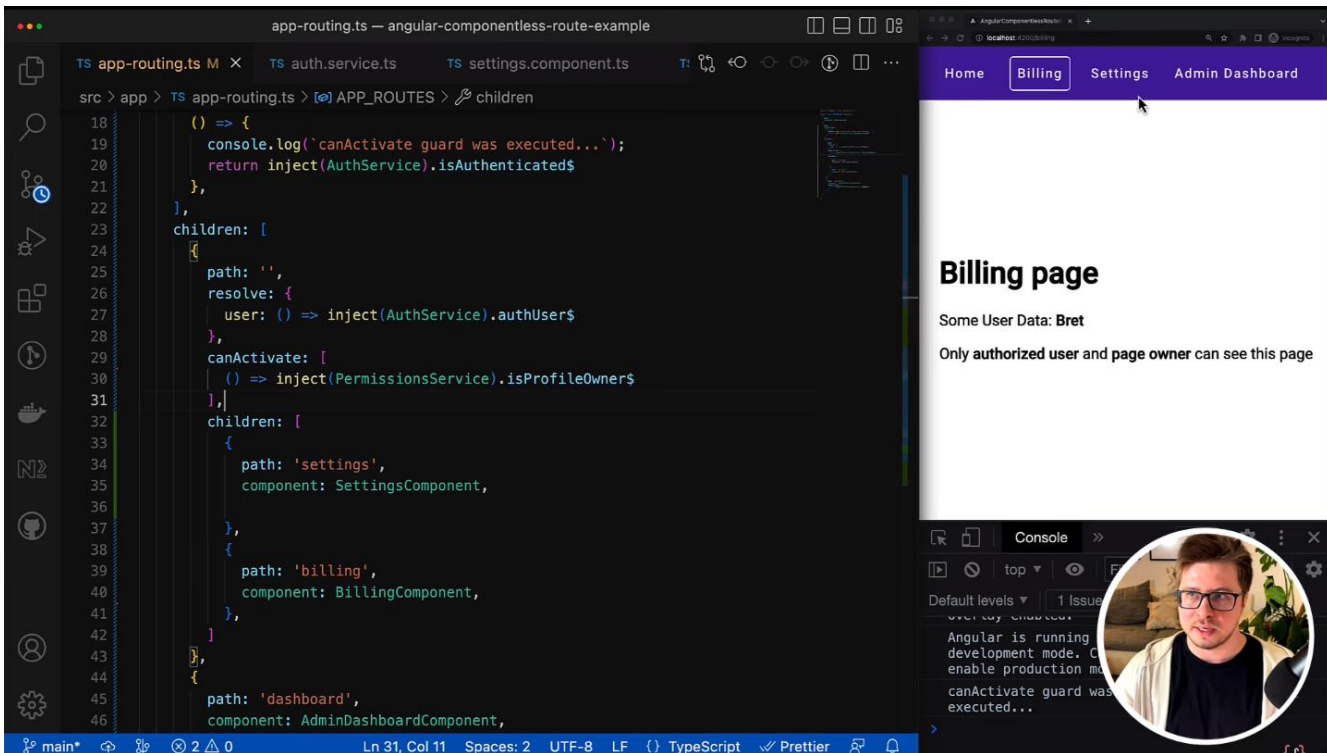
897

},

898



The resolver `user()` will now not be re-executed when we navigate between the billing and settings page except once.



app-routing.ts — angular-componentless-route-example

TS app-routing.ts M X

TS auth.service.ts

TS settings.component.ts

src > app > TS app-routing.ts > APP\_ROUTES > children

```
18      () => {
19          console.log('canActivate guard was executed...');
20          return inject(AuthService).isAuthenticated$
21      },
22  ],
23  children: [
24      {
25          path: '',
26          resolve: {
27              user: () => inject(AuthService).authUsers$
28          },
29          canActivate: [
30              () => inject(PermissionsService).isProfileOwner$
31          ],
32          children: [
33              {
34                  path: 'settings',
35                  component: SettingsComponent,
36              },
37              {
38                  path: 'billing',
39                  component: BillingComponent,
40              },
41          ],
42      },
43      {
44          path: 'dashboard',
45          component: AdminDashboardComponent,
46      },
47  ],
48  },
49  ],
50  ],
51  ],
52  ],
53  ],
54  ],
55  ],
56  ],
57  ],
58  ],
59  ],
60  ],
61  ],
62  ],
63  ],
64  ],
65  ],
66  ],
67  ],
68  ],
69  ],
70  ],
71  ],
72  ],
73  ],
74  ],
75  ],
76  ],
77  ],
78  ],
79  ],
80  ],
81  ],
82  ],
83  ],
84  ],
85  ],
86  ],
87  ],
88  ],
89  ],
90  ],
91  ],
92  ],
93  ],
94  ],
95  ],
96  ],
97  ],
98  ],
99  ],
100 ],
```

AngularComponentlessRouteExample

Home Billing Settings Admin Dashboard

Settings Page

Some User Data: Sincere@april.biz

Only authorized user and page owner can see this page

localhost:4200/console

Console

Angular is running in development mode. Click here to enable production mode.

canActivate guard was executed...

app-routing.ts — angular-componentless-route-example

TS app-routing.ts M X

TS auth.service.ts

TS settings.component.ts

src > app > TS app-routing.ts > APP\_ROUTES > children

```
18      () => {
19          console.log('canActivate guard was executed...');
20          return inject(AuthService).isAuthenticated$
21      },
22  ],
23  children: [
24      {
25          path: '',
26          resolve: {
27              user: () => inject(AuthService).authUsers$
28          },
29          canActivate: [
30              () => inject(PermissionsService).isProfileOwner$
31          ],
32          children: [
33              {
34                  path: 'settings',
35                  component: SettingsComponent,
36              },
37              {
38                  path: 'billing',
39                  component: BillingComponent,
40              },
41          ],
42      },
43      {
44          path: 'dashboard',
45          component: AdminDashboardComponent,
46      },
47  ],
48  },
49  ],
50  ],
51  ],
52  ],
53  ],
54  ],
55  ],
56  ],
57  ],
58  ],
59  ],
60  ],
61  ],
62  ],
63  ],
64  ],
65  ],
66  ],
67  ],
68  ],
69  ],
70  ],
71  ],
72  ],
73  ],
74  ],
75  ],
76  ],
77  ],
78  ],
79  ],
80  ],
81  ],
82  ],
83  ],
84  ],
85  ],
86  ],
87  ],
88  ],
89  ],
90  ],
91  ],
92  ],
93  ],
94  ],
95  ],
96  ],
97  ],
98  ],
99  ],
100 ],
```

AngularComponentlessRouteExample

Home Billing Settings Admin Dashboard

Admin Dashboard

Only authorized users with Admin rights can see the page

localhost:4200/console

Console

Angular is running in development mode. Click here to enable production mode.

canActivate guard was executed...



app-routing.ts — angular-componentless-route-example

TS app-routing.ts M X

TS auth.service.ts

TS settings.component.ts

TS bill

src > app > TS app-routing.ts > [e] APP\_ROUTES > path

```
15 {
16   path: '',
17   canActivate: [
18     () => {
19       console.log('canActivate guard was executed...');
20       return inject(AuthService).isAuthenticated$
21     },
22   ],
23   children: [
24     {
25       path: '',
26       resolve: {
27         user: () => inject(AuthService).authUser$,
28       },
29       canActivate: [
30         () => inject(PermissionsService).isProfileOwner$,
31       ],
32       children: [
33         {
34           path: 'settings',
35           component: SettingsComponent,
36         },
37       ],
38     },
39   ],
40 },
41 },
42 },
43 }
```

“settings” + “” + “” = /settings

AngularComponentlessRouteExample

Home Billing Settings Admin Dashboard

Settings Page

Some User Data: Sincere@april.biz

Only authorized user and page owner can see this page

Elements Console Network

Filter

Fetch/XHR JS CSS Img Media Font Other

Has blocked cookies Blocked Request

10000 ms 20000 ms 30000 ms

Name Status T

1 200 X

1 200 X

1 200 X

3 / 20 requests 3.1 kB / 3.7 MB transferred 1.5 kB / 3.7 MB resources

settings.component.ts — angular-componentless-route-example

TS app-routing.ts M

TS auth.service.ts

TS settings.component.ts X

TS bill

src > app > pages > TS settings.component.ts > SettingsComponent

```
1 > import { AsyncPipe, NgIf } from '@angular/common';
2
3
4
5
6 @Component({
7   selector: 'app-settings',
8   standalone: true,
9   imports: [AsyncPipe, NgIf],
10  template: `
11    <h1>Settings Page</h1>
12    <p *ngIf="(user$ | async) as user">Some User Data: <b>{{user?.email}}</b>
13    <p>
14      Only <b>authorized user</b> and <b>page owner</b> can see this page
15    </p>
16  `
17 })
18 export class SettingsComponent {
19   user$ = inject(ActivatedRoute).data.pipe(map(({ user }) => user));
20 }
21
```

Ln 18, Col 33 Spaces: 2 UTF-8 LF TypeScript Prettier

AngularComponentlessRouteExample

Home Billing Settings Admin Dashboard

Settings Page

Some User Data: Sincere@april.biz

Only authorized user and page owner can see this page

Elements Console Network

Filter

Fetch/XHR JS CSS Img Media Font Other

Has blocked cookies Blocked Request

10000 ms 20000 ms 30000 ms

Name Status T

1 200 X

1 200 X

1 200 X

3 / 20 requests 3.1 kB / 3.7 MB transferred 1.5 kB / 3.7 MB resources