

Signals in Action: Loading Indicator

Deborah Kurata
19.5K subscribers

Subscribe

Like 290 | Share | Download | Thanks | ...

4,055 views Dec 18, 2024 #angular #resource #angularsignals

We often display a loading indicator when retrieving data. This is especially useful if the user's connection is slow.

This video demonstrates an easy way to display a loading indicator using signals.

00:00 Easiest way to display a loading indicator
00:14 Sample application
01:59 isLoading property of a resource

03:22 isLoading in action
04:51 Implementing a loading indicator
06:35 Use the isLoading property of a resource

```
stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts
TS vehicle.service.ts X TS vehicle-selection.component.ts
1 export class VehicleService {
2   private vehicleUrl = 'https://swapi.py4e.com/api/vehicles';
3
4   // Injected services
5   private http = inject(HttpClient);
6
7   // Signals managed by the service
8   selectedVehicle = signal<Vehicle | undefined>(undefined);
9
10  // Reset the quantity when the vehicle changes
11  quantity = linkedSignal({
12    source: this.selectedVehicle,
13    computation: (v) => {
14      if (v) {
15        return 1;
16      }
17      return 0;
18    }
19  });
20
21  // Computed signals
22  total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
23  color = computed(() => this.total() > 50000 ? 'green' : 'blue');
24
25  vehiclesResource = rxResource({
26    loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
27      map(vr => vr.results)
28    )
29  });
30
31
32
33
34
35
36
37
```

Star Wars Vehicle Sales

... loading vehicles

```
stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts
TS vehicle.service.ts X TS vehicle-selection.component.ts
1 export class VehicleService {
2   private vehicleUrl = 'https://swapi.py4e.com/api/vehicles';
3
4   // Injected services
5   private http = inject(HttpClient);
6
7   // Signals managed by the service
8   selectedVehicle = signal<Vehicle | undefined>(undefined);
9
10  // Reset the quantity when the vehicle changes
11  quantity = linkedSignal({
12    source: this.selectedVehicle,
13    computation: (v) => {
14      if (v) {
15        return 1;
16      }
17      return 0;
18    }
19  });
20
21  // Computed signals
22  total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
23  color = computed(() => this.total() > 50000 ? 'green' : 'blue');
24
25  vehiclesResource = rxResource({
26    loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
27      map(vr => vr.results)
28    )
29  });
30
31
32
33
34
35
36
37
```

Star Wars Vehicle Sales

Select a vehicle: Quantity: 0

Vehicle: Price: Total: 0

```

9  export class VehicleService {
10    private vehicleUrl = 'https://swapi.py4e.com/api/vehicles';
11
12    // Injected services
13    private http = inject(HttpClient);
14
15    // Signals managed by the service
16    selectedVehicle = signal<Vehicle | undefined>(<undefined>);
17
18    // Reset the quantity when the vehicle changes
19    quantity = linkedSignal({
20      source: this.selectedVehicle,
21      computation: (v) => {
22        if (v) {
23          return 1;
24        }
25        return 0;
26      }
27    });
28
29    // Computed signals
30    total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
31    color = computed(() => this.total() > 50000 ? 'green' : 'blue');
32
33    vehiclesResource = rxResource({
34      loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
35        map(vr => vr.results)
36      )
37    });
38    vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39  }
40
41  export interface VehicleResponse {
42    count: number;
43    next: string;
44    previous: string;
45    results: Vehicle[];
46  }
47
48  export interface Vehicle {
49    name: string;
50    cost_in_credits: number;
51  }

```

Star Wars Vehicle Sales

Select a vehicle—
Quantity: 0
Vehicle:
Price:
Total: 0

This returns the response as a signal in Angular version 19

```

28
29    // Computed signals
30    total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
31    color = computed(() => this.total() > 50000 ? 'green' : 'blue');
32
33    vehiclesResource = rxResource({
34      loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
35        map(vr => vr.results)
36      )
37    );
38    vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39  }
40
41  export interface VehicleResponse {
42    count: number;
43    next: string;
44    previous: string;
45    results: Vehicle[];
46  }
47
48  export interface Vehicle {
49    name: string;
50    cost_in_credits: number;
51  }

```

Star Wars Vehicle Sales

Select a vehicle—
Quantity: 0
Vehicle:
Price:
Total: 0

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdq?file=src/app/vehicle.service.ts

Signals in Action (forked)

Share

vehicle.service.ts vehicle-selection.component.ts

```
28 // Computed signals
29 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
30 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
31
32 vehiclesResource = rxResource({
33   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
34     map(vr => vr.results)
35   )
36 });
37 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
38
39 }
40
41 export interface VehicleResponse {
42   count: number;
43   next: string;
44   previous: string;
45   results: Vehicle[]
46 }
47
48 export interface Vehicle [
49   name: string;
50   cost_in_credits: number;
51 ]
```

Star Wars Vehicle Sales

Select a vehicle—
Quantity: 0
Vehicle:
Price:
Total: 0

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdq?file=src/app/vehicle.service.ts

Signals in Action (forked)

Share

vehicle.service.ts vehicle-selection.component.ts

```
28 // Computed signals
29 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
30 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
31
32 vehiclesResource = rxResource({
33   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
34     map(vr => vr.results)
35   )
36 });
37 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
38
39 }
40
41 export interface VehicleResponse {
42   count: number;
43   next: string;
44   previous: string;
45   results: Vehicle[]
46 }
47
48 export interface Vehicle [
49   name: string;
50   cost_in_credits: number;
51 ]
```

Star Wars Vehicle Sales

Select a vehicle—
Quantity: 0
Vehicle:
Price:
Total: 0

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdq?file=src/app/vehicle.service.ts

Signals in Action (forked)

Share

vehicle.service.ts vehicle-selection.component.ts

```
28 // Computed signals
29 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
30 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
31
32 (property) VehicleService.vehiclesResource: ResourceRef<Vehicle[]>
33 vehiclesResource = rxResource({
34   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
35     map(vr => vr.results)
36   )
37 });
38 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39
40
41 export interface VehicleResponse {
42   count: number;
43   next: string;
44   previous: string;
45   results: Vehicle[]
46 }
47
48 export interface Vehicle [
49   name: string;
50   cost_in_credits: number;
51 ]
```

Star Wars Vehicle Sales

Select a vehicle—
Quantity: 0
Vehicle:
Price:
Total: 0

```
28 // Computed signals
29 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
30 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
31
32 vehiclesResource = rxResource({
33   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
34     map(vr => vr.results)
35   )
36 });
37
38 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39 }
40
41 export interface VehicleResponse {
42   count: number;
43   next: string;
44   previous: string;
45   results: Vehicle[]
46 }
47
48 export interface Vehicle {
49   name: string;
50   cost_in_credits: number;
51 }
```

Star Wars Vehicle Sales

--Select a vehicle--
Quantity: 0
Vehicle:
Price:
Total: 0

```
28 // Computed signals
29 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
30 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
31
32 vehiclesResource = rxResource({
33   loader: () => this.http.get<VehicleResponse>()
34     .pipe((property) WritableResource<Vehicle[]>.value: WritableSignal<Vehicle[] | undefined>)
35     .map(vr => vr.results)
36   )
37 });
38
39 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
40 }
```

Star Wars Vehicle Sales

--Select a vehicle--
Quantity: 0
Vehicle:
Price:
Total: 0

The current value of the `Resource` , or `undefined` if there is no current value.

```
28 // Computed signals
29 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
30 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
31
32 vehiclesResource = rxResource({
33   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
34     map(vr => vr.results)
35   )
36 });
37
38 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39 }
40
```

Star Wars Vehicle Sales

--Select a vehicle--
Quantity: 0
Vehicle:
Price:
Total: 0

Star Wars Vehicle Sales

```

28 // Computed signals
29 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
30 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
31
32 vehiclesResource = rxResource({
33   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
34     map(vr => vr.results)
35   )
36 });
37
38 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39 }
40
41 export interface VehicleResponse {
42   count: number;
43   next: string;
44   previous: string;
45   results: Vehicle[]
46 }
47
48 export interface Vehicle {
49   name: string;
50   cost_in_credits: number;
51 }

```

The `isLoading` property is also a signal that we can use as below

```

28 // Computed signals
29 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
30 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
31
32 vehiclesResource = rxResource({
33   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
34     map(vr => vr.results)
35   )
36 });
37
38 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39
40 ! loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
41 }
42
43 export interface VehicleResponse {
44   count: number;
45   next: string;
46   previous: string;
47   results: Vehicle[]
48 }
49
50 export interface Vehicle {
51   name: string;
52   cost_in_credits: number;
53 }

```

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Share

vehicle.service.ts

```
28
29 // Computed signals
30 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
31 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
32
33 vehiclesResource = rxResource({
34   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
35     map(vr => vr.results)
36   )
37 });
38 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39
40 loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
41 vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
42 }
43
44 export interface VehicleResponse {
45   count: number;
46   next: string;
47   previous: string;
48   results: Vehicle[]
49 }
50
51 export interface Vehicle {
52   name: string;
53   cost_in_credits: number;
54 }
```

Star Wars Vehicle Sales

--Select a vehicle--

Quantity: 0

Vehicle:

Price:

Total: 0

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Share

vehicle.service.ts

```
28
29 // Computed signals
30 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
31 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
32
33 vehiclesResource = rxResource({
34   loader: () => {
35     console.log('Before http request');
36     return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37       tap(() => console.log('Before map')),
38       map(vr => vr.results),
39       tap(() => console.log('After map'))
40     )
41   }
42 });
43 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
44
45 loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
46 vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
47
48 export interface VehicleResponse {
49   count: number;
50   next: string;
51   previous: string;
52   results: Vehicle[]
53 }
54
55 export interface Vehicle {
56   name: string;
```

Star Wars Vehicle Sales

--Select a vehicle--

Quantity: 0

Vehicle:

Price:

Total: 0

stackblitz.com/-/edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Star Wars Vehicle Sales

```
28
29 // Computed signals
30 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
31 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
32
33 vehiclesResource = rxResource({
34   loader: () => {
35     console.log('Before http request');
36     return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37       tap(() => console.log('Before map')),
38       map(vr => vr.results),
39       tap(() => console.log('After map'))
40     )
41   });
42   vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43 
```

Elements Console Performance insights Recorder Sources Network Performance Memory Application Security Lighthouse

Default levels ▾ | 43 Issues: 2 4 37 2 hidden

Before http request
Loading indicator: true
Vehicle data: ▶ []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {}]

vehicle.service.ts:35
vehicle.service.ts:44
vehicle.service.ts:45
core.mjs:16728
vehicle.service.ts:37
vehicle.service.ts:39
vehicle.service.ts:44
vehicle.service.ts:45

stackblitz.com/-/edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Star Wars Vehicle Sales

```
@Injectable({
  providedIn: 'root'
})
export class VehicleService {
  private vehicleUrl = 'https://swapi.py4e.com/api/vehicles';

  // Injected services
  private http = inject(HttpClient);

  // Signals managed by the service
  selectedVehicle = signal<Vehicle | undefined>(undefined);

  // Reset the quantity when the vehicle changes
  quantity = linkedSignal({
    source: this.selectedVehicle,
    computation: (v) => {
```

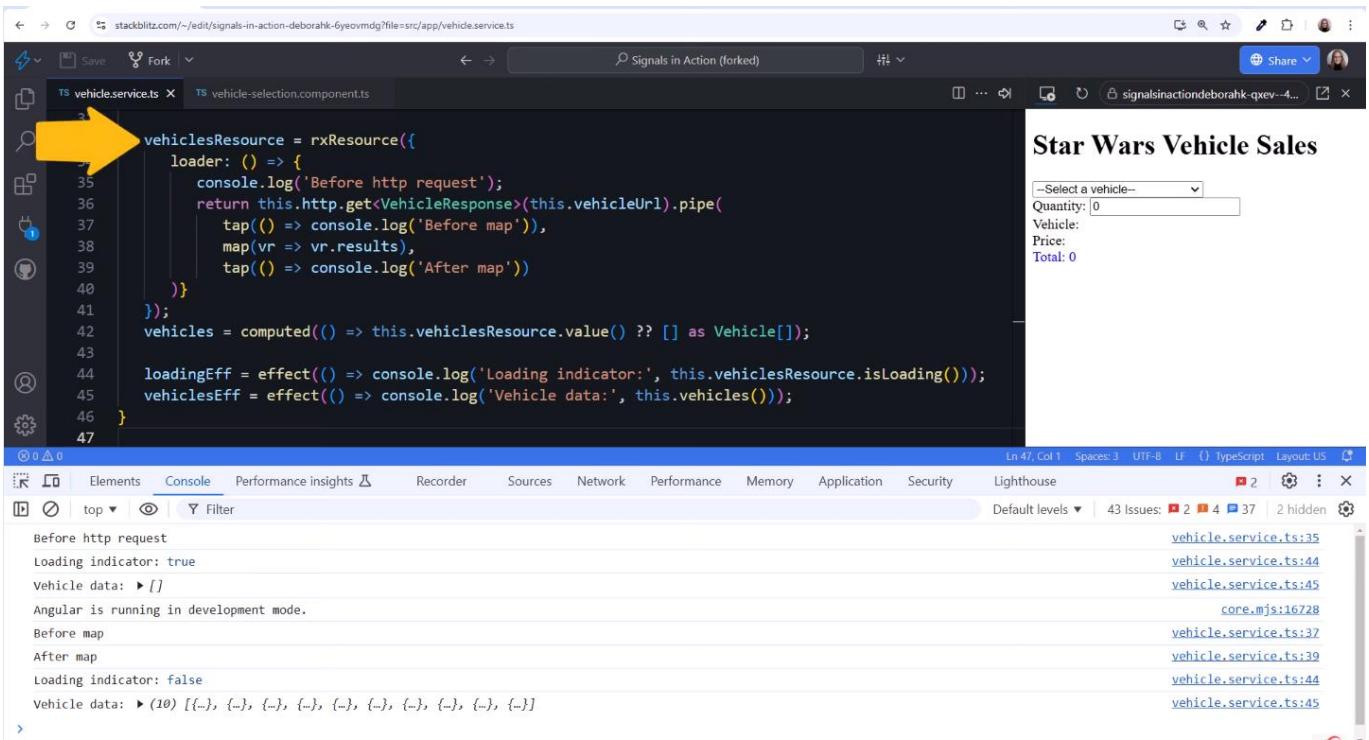
Elements Console Performance insights Recorder Sources Network Performance Memory Application Security Lighthouse

Default levels ▾ | 43 Issues: 2 4 37 2 hidden

Before http request
Loading indicator: true
Vehicle data: ▶ []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {}]

vehicle.service.ts:35
vehicle.service.ts:44
vehicle.service.ts:45
core.mjs:16728
vehicle.service.ts:37
vehicle.service.ts:39
vehicle.service.ts:44
vehicle.service.ts:45





stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Share

vehicle.service.ts

```
35 vehiclesResource = rxResource({
36   loader: () => {
37     console.log('Before http request');
38     return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
39       tap(() => console.log('Before map')),
40       map(vr => vr.results),
41       tap(() => console.log('After map'))
42     );
43   };
44   vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
45
46   loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
47   vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
48 }
```

Star Wars Vehicle Sales

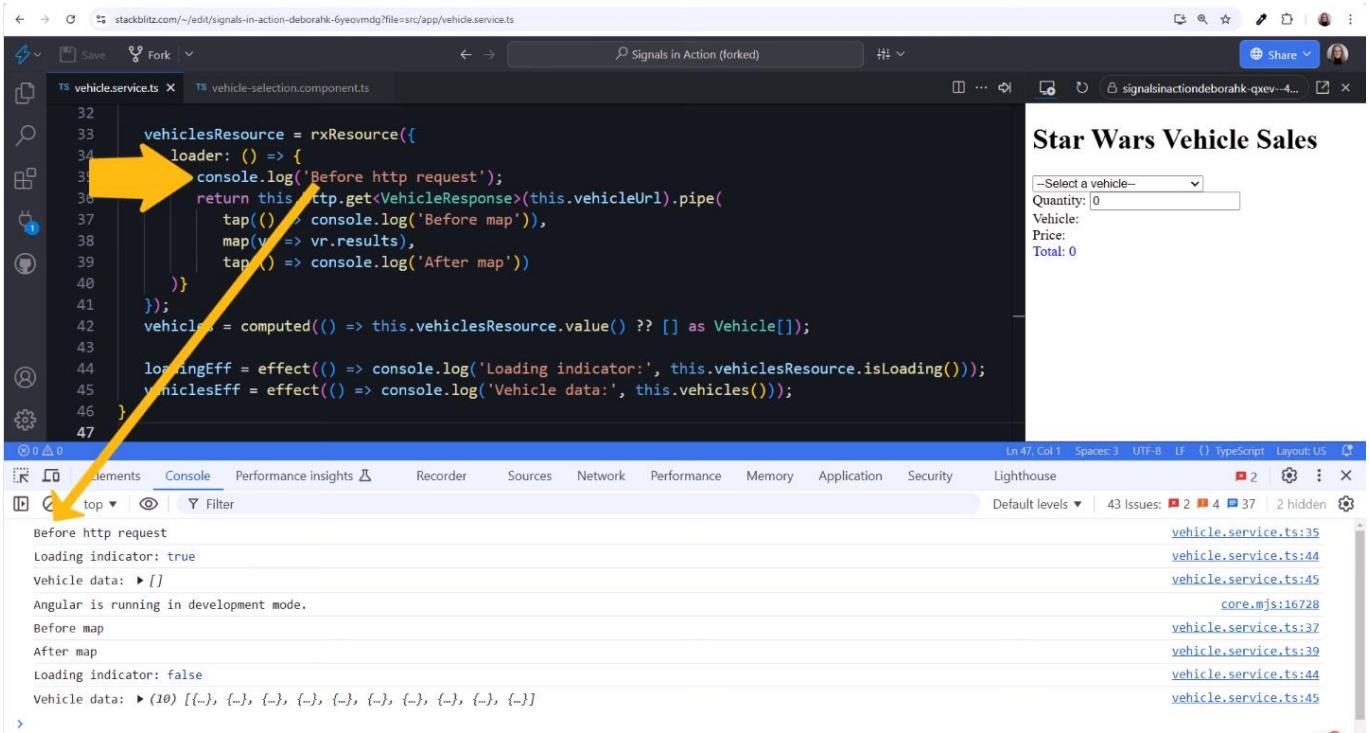
-Select a vehicle-
Quantity: 0
Vehicle:
Price:
Total: 0

Elements Console Performance insights Recorder Sources Network Performance Memory Application Security Lighthouse

Default levels | 43 Issues: 2 4 37 2 hidden

Console output:

```
Before http request
Loading indicator: true
Vehicle data: ▶ []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {} ]
```



stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Share

vehicle.service.ts

```
32
33   vehiclesResource = rxResource({
34     loader: () => {
35       console.log('Before http request');
36       return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37         tap(() => console.log('Before map')),
38         map(vr => vr.results),
39         tap(() => console.log('After map'))
40       );
41     };
42     vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44     loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45     vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46   }
47 }
```

Star Wars Vehicle Sales

-Select a vehicle-
Quantity: 0
Vehicle:
Price:
Total: 0

Elements Console Performance insights Recorder Sources Network Performance Memory Application Security Lighthouse

Default levels | 43 Issues: 2 4 37 2 hidden

Console output:

```
Before http request
Loading indicator: true
Vehicle data: ▶ []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {} ]
```

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Star Wars Vehicle Sales

```
32
33   vehiclesResource = rxResource({
34     loader: () => {
35       console.log('Before http request');
36       return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37         tap(() => console.log('Before map')),
38         map(vr => vr.results),
39         tap(() => console.log('After map'))
40       );
41     });
42     vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44     loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45     vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46   }
47 }
```

Console Output:

```
Before http request
Loading indicator: true
Vehicle data: []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {} ]
```

Stack Trace:

- vehicle.service.ts:35
- vehicle.service.ts:44
- vehicle.service.ts:45
- core.mjs:16728
- vehicle.service.ts:37
- vehicle.service.ts:39
- vehicle.service.ts:44
- vehicle.service.ts:45

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Star Wars Vehicle Sales

```
32
33   vehiclesResource = rxResource({
34     loader: () => {
35       console.log('Before http request');
36       return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37         tap(() => console.log('Before map')),
38         map(vr => vr.results),
39         tap(() => console.log('After map'))
40       );
41     });
42     vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44     loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45     vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46   }
47 }
```

Console Output:

```
Before http request
Loading indicator: true
Vehicle data: []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {} ]
```

Stack Trace:

- vehicle.service.ts:35
- vehicle.service.ts:44
- vehicle.service.ts:45
- core.mjs:16728
- vehicle.service.ts:37
- vehicle.service.ts:39
- vehicle.service.ts:44
- vehicle.service.ts:45

stackblitz.com/~/edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Star Wars Vehicle Sales

```
32
33   vehiclesResource = rxResource({
34     loader: () => {
35       console.log('Before http request');
36       return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37         tap(() => console.log('Before map')),
38         map(vr => vr.results),
39         tap(() => console.log('After map'))
40       );
41     });
42     vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44     loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45     vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46   }
47 
```

Console

```
Before http request
Loading indicator: true
Vehicle data: []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {} ]
```

Elements Performance insights Recorder Sources Network Performance Memory Application Security Lighthouse

Default levels 43 Issues: 2 4 37 2 hidden

stackblitz.com/~/edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle.service.ts

Signals in Action (forked)

Star Wars Vehicle Sales

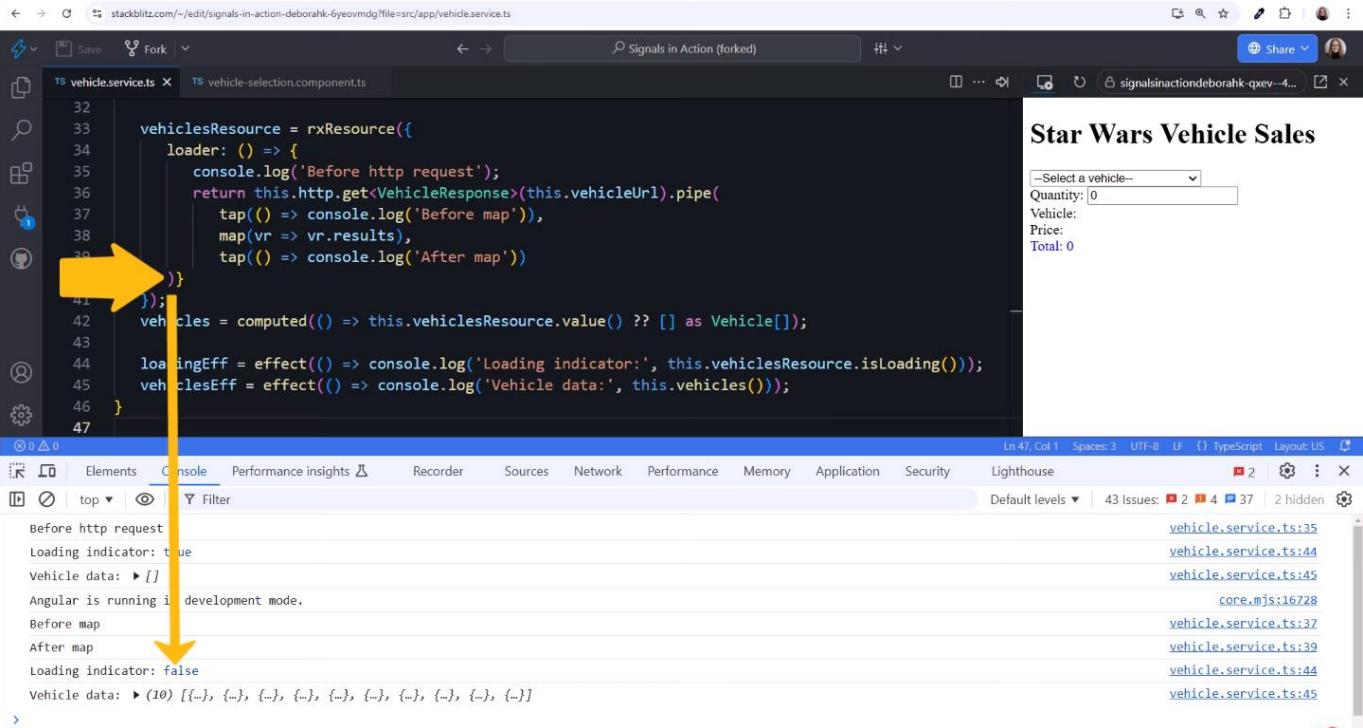
```
32
33   vehiclesResource = rxResource({
34     loader: () => {
35       console.log('Before http request');
36       return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37         tap(() => console.log('Before map')),
38         map(vr => vr.results),
39         tap(() => console.log('After map'))
40       );
41     });
42     vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44     loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45     vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46   }
47 
```

Console

```
Before http request
Loading indicator: true
Vehicle data: []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {} ]
```

Elements Performance insights Recorder Sources Network Performance Memory Application Security Lighthouse

Default levels 43 Issues: 2 4 37 2 hidden

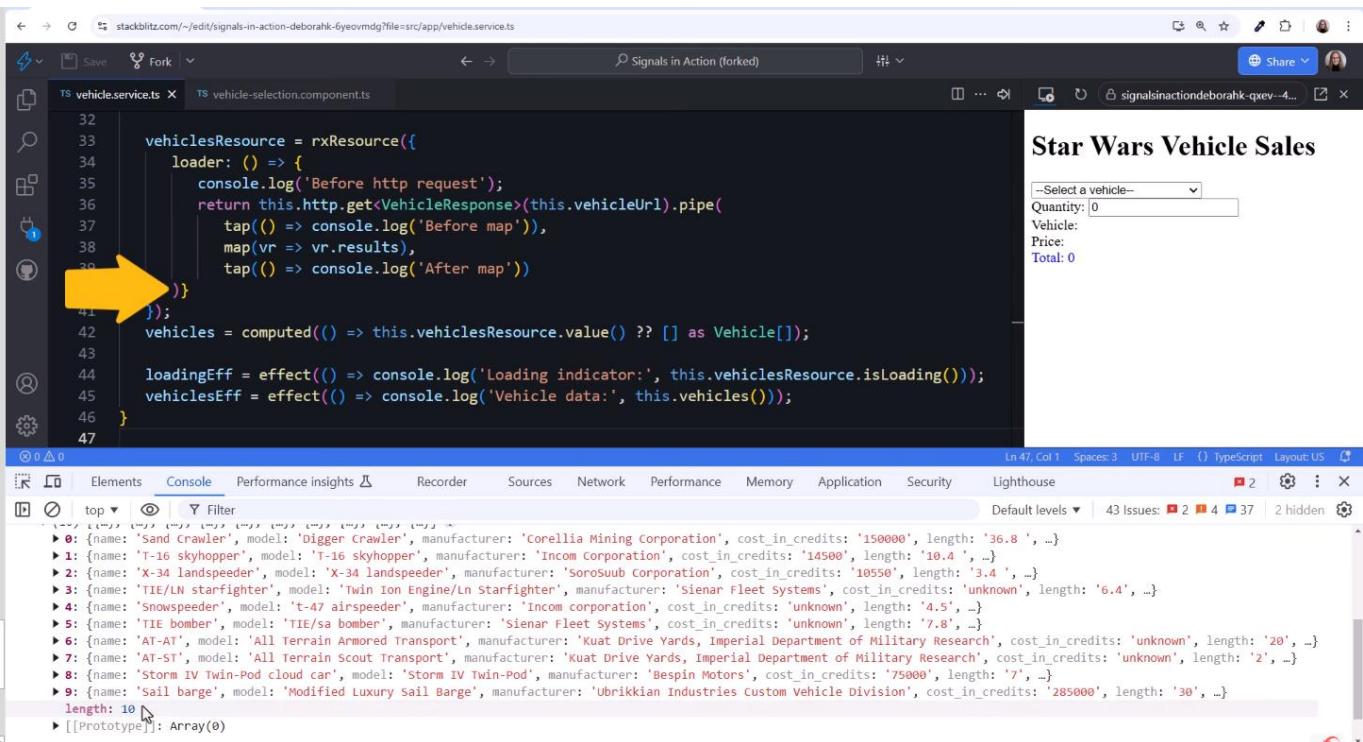


```
32
33     vehiclesResource = rxResource({
34         loader: () => {
35             console.log('Before http request');
36             return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37                 tap(() => console.log('Before map')),
38                 map(vr => vr.results),
39                 tap(() => console.log('After map'))
40             );
41         });
42         vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44         loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45         vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46     }
47 
```

Elements Console Performance insights Recorder Sources Network Performance Memory Application Security Lighthouse Default levels ▾ | 43 Issues: 2 4 37 2 hidden

Before http request
Loading indicator: true
Vehicle data: ▶ []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {}]

vehicle.service.ts:35
vehicle.service.ts:44
vehicle.service.ts:45
core.mjs:16728
vehicle.service.ts:37
vehicle.service.ts:39
vehicle.service.ts:44
vehicle.service.ts:45



```
32
33     vehiclesResource = rxResource({
34         loader: () => {
35             console.log('Before http request');
36             return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37                 tap(() => console.log('Before map')),
38                 map(vr => vr.results),
39                 tap(() => console.log('After map'))
40             );
41         });
42         vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44         loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45         vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46     }
47 
```

Elements Console Performance insights Recorder Sources Network Performance Memory Application Security Lighthouse Default levels ▾ | 43 Issues: 2 4 37 2 hidden

Before http request
Loading indicator: true
Vehicle data: ▶ []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data: ▶ (10) [{} , {} , {} , {} , {} , {} , {} , {} , {} , {}]
▶ 0: {name: 'Sand Crawler', model: 'Digger Crawler', manufacturer: 'Corellia Mining Corporation', cost_in_credits: '150000', length: '36.8', ...}
▶ 1: {name: 'T-16 skyhopper', model: 'T-16 skyhopper', manufacturer: 'Incom Corporation', cost_in_credits: '14500', length: '10.4', ...}
▶ 2: {name: 'X-34 landspeeder', model: 'X-34 landspeeder', manufacturer: 'SoroSuub Corporation', cost_in_credits: '10550', length: '3.4', ...}
▶ 3: {name: 'TIE/LN starfighter', model: 'Twin Ion Engine/LN Starfighter', manufacturer: 'Sienar Fleet Systems', cost_in_credits: 'unknown', length: '6.4', ...}
▶ 4: {name: 'Snowspeeder', model: 'T-47 airspeeder', manufacturer: 'Incom corporation', cost_in_credits: 'unknown', length: '4.5', ...}
▶ 5: {name: 'TIE bomber', model: 'TIE/sa bomber', manufacturer: 'Sienar Fleet Systems', cost_in_credits: 'unknown', length: '7.8', ...}
▶ 6: {name: 'AT-AT', model: 'All Terrain Armored Transport', manufacturer: 'Kuat Drive Yards, Imperial Department of Military Research', cost_in_credits: 'unknown', length: '20', ...}
▶ 7: {name: 'AT-ST', model: 'All Terrain Scout Transport', manufacturer: 'Kuat Drive Yards, Imperial Department of Military Research', cost_in_credits: 'unknown', length: '2', ...}
▶ 8: {name: 'Storm IV Twin-Pod cloud car', model: 'Storm IV Twin-Pod', manufacturer: 'Bespin Motors', cost_in_credits: '75000', length: '7', ...}
▶ 9: {name: 'sail barge', model: 'Modified Luxury Sail Barge', manufacturer: 'Ubrikian Industries Custom Vehicle Division', cost_in_credits: '285000', length: '30', ...}
length: 10
▶ [[Prototype]: Array(0)]

StackBlitz editor interface showing code in vehicle.service.ts and vehicle-selection.component.ts. The vehicle-selection component displays a dropdown menu and some summary statistics.

```
32
33 vehiclesResource = rxResource({
34   loader: () => {
35     console.log('Before http request');
36     return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37       tap(() => console.log('Before map')),
38       map(vr => vr.results),
39       tap(() => console.log('After map'))
40     );
41   });
42   vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44   loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45   vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46 }
```

Console output:

```
Before http request
Loading indicator: true
Vehicle data: []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data:
▼ (10) [{}]
▶ 0: {name: 'Sand Crawler', model: 'Digger Crawler', manufacturer: 'Corellia Mining Corporation', cost_in_credits: '150000', length: '36.8', ...}
```

StackBlitz editor interface showing code in vehicle.service.ts and vehicle-selection.component.ts. The vehicle-selection component displays a dropdown menu and some summary statistics.

```
32
33 vehiclesResource = rxResource({
34   loader: () => {
35     console.log('Before http request');
36     return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37       tap(() => console.log('Before map')),
38       map(vr => vr.results),
39       tap(() => console.log('After map'))
40     );
41   });
42   vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44   loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45   vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46 }
```

Console output:

```
Before http request
Loading indicator: true
Vehicle data: []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data:
▼ (10) [{}]
▶ 0: {name: 'Sand Crawler', model: 'Digger Crawler', manufacturer: 'Corellia Mining Corporation', cost_in_credits: '150000', length: '36.8', ...}
```

StackBlitz editor interface showing code in vehicle.service.ts and vehicle-selection.component.ts. The vehicle-selection component displays a dropdown menu and some summary statistics.

```
32
33 vehiclesResource = rxResource({
34   loader: () => {
35     console.log('Before http request');
36     return this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
37       tap(() => console.log('Before map')),
38       map(vr => vr.results),
39       tap(() => console.log('After map'))
40     );
41   });
42   vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
43
44   loadingEff = effect(() => console.log('Loading indicator:', this.vehiclesResource.isLoading()));
45   vehiclesEff = effect(() => console.log('Vehicle data:', this.vehicles()));
46 }
```

Console output:

```
Before http request
Loading indicator: true
Vehicle data: []
Angular is running in development mode.
Before map
After map
Loading indicator: false
Vehicle data:
▼ (10) [{}]
▶ 0: {name: 'Sand Crawler', model: 'Digger Crawler', manufacturer: 'Corellia Mining Corporation', cost_in_credits: '150000', length: '36.8', ...}
```



```
28
29 // Computed signals
30 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
31 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
32
33 vehiclesResource = rxResource({
34   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
35     map(vr => vr.results)
36   )
37 });
38 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39
40 }
41
42 export interface VehicleResponse {
43   count: number;
44   next: string;
45   previous: string;
46   results: Vehicle[]
47 }
48
49 export interface Vehicle {
50   name: string;
51   cost_in_credits: number;
52 }
```

Star Wars Vehicle Sales

--Select a vehicle--
Quantity: 0
Vehicle:
Price:
Total: 0

Now we can use the isLoading property to display a loading indicator as below



```
28
29 // Computed signals
30 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
31 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
32
33 private vehiclesResource = rxResource({
34   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
35     map(vr => vr.results)
36   )
37 });
38 (property) VehicleService.isLoading: Signal<boolean> ) ?? [] as Vehicle[]);
39 isLoading = this.vehiclesResource.isLoading;
40 }
41
42 export interface VehicleResponse {
43   count: number;
44   next: string;
45   previous: string;
46   results: Vehicle[]
47 }
48
49 export interface Vehicle {
50   name: string;
51   cost_in_credits: number;
52 }
```

Star Wars Vehicle Sales

--Select a vehicle--
Quantity: 0
Vehicle:
Price:
Total: 0

```

28
29 // Computed signals
30 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
31 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
32
33 private vehiclesResource = rxResource({
34   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
35     map(vr => vr.results),
36     delay(2000)
37   );
38   vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
39   isLoading = this.vehiclesResource.isLoading;
40 }
41
42 export interface VehicleResponse {
43   count: number;
44   next: string;
45   previous: string;
46   results: Vehicle[]
47 }
48
49 export interface Vehicle {
50   name: string;
51   cost_in_credits: number;
52 }

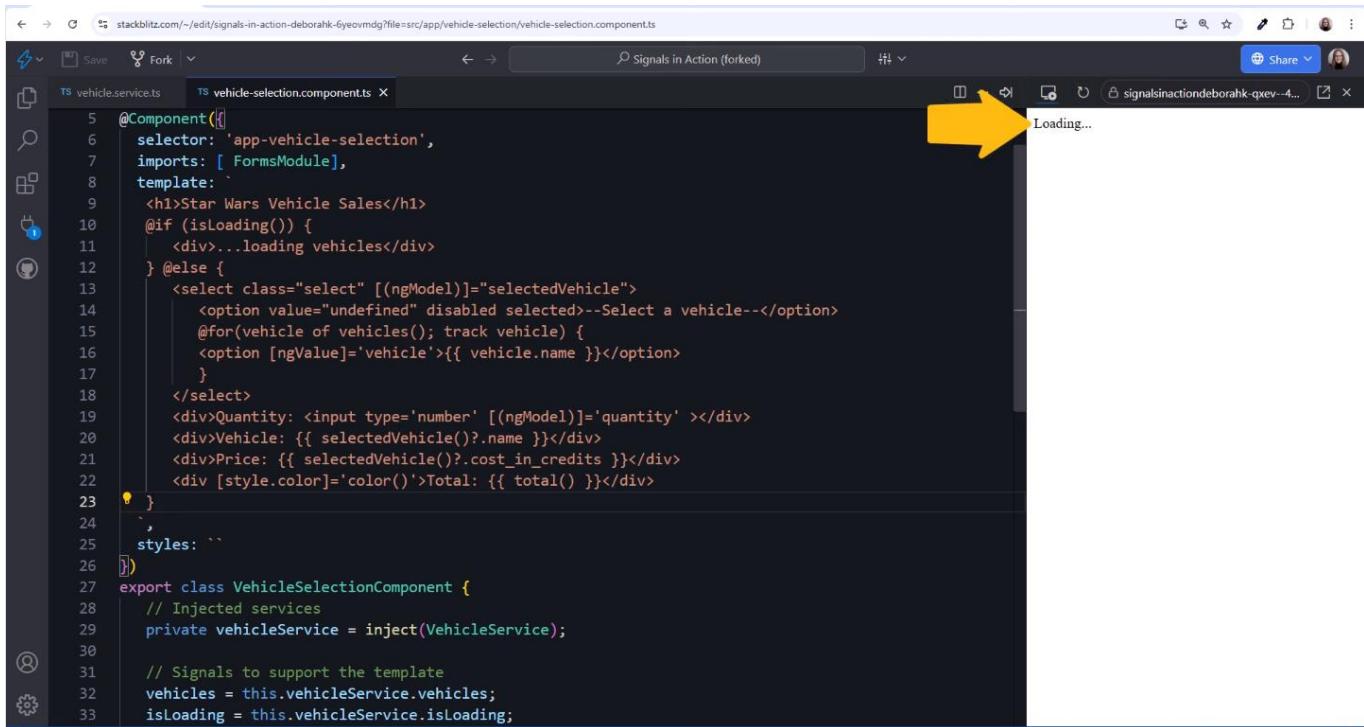
```

Let us add a delay so we can see the indicator

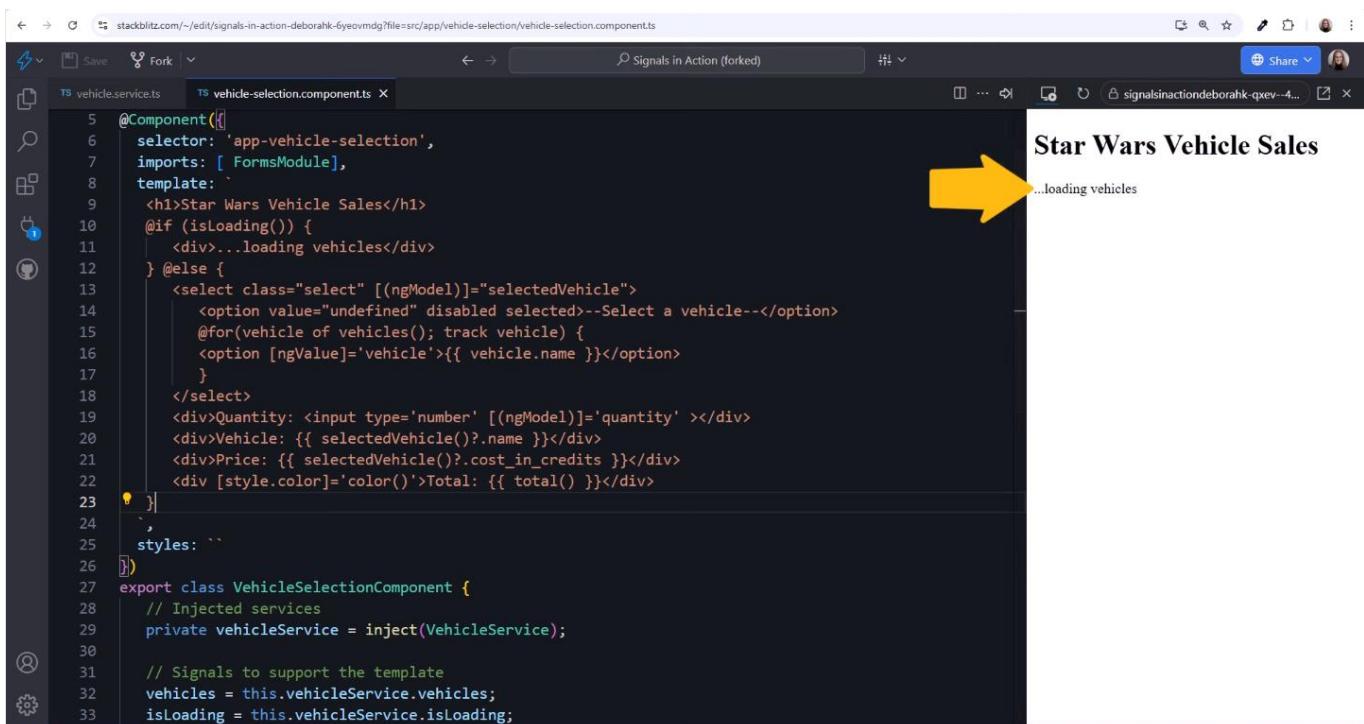
```

5 @Component({
6   selector: 'app-vehicle-selection',
7   imports: [FormsModule],
8   template: `
9     <h1>Star Wars Vehicle Sales</h1>
10    <select class="select" [(ngModel)]="selectedVehicle">
11      <option value="undefined" disabled selected>--Select a vehicle--</option>
12      <for(vehicle of vehicles(); track vehicle) {
13        <option [ngValue]=`vehicle`>{{ vehicle.name }}</option>
14      }
15    </select>
16    <div>Quantity: <input type='number' [(ngModel)]='quantity' ></div>
17    <div>Vehicle: {{ selectedVehicle()?.name }}</div>
18    <div>Price: {{ selectedVehicle()?.cost_in_credits }}</div>
19    <div [style.color]='color()'>Total: {{ total() }}</div>
20  ,
21  styles: ``
22 })
23 export class VehicleSelectionComponent {
24   // Injected services
25   private vehicleService = inject(VehicleService);
26
27   // Signals to support the template
28   vehicles = this.vehicleService.vehicles;
29   isLoading = this.vehicleService.isLoading;
30   selectedVehicle = this.vehicleService.selectedVehicle;
31   quantity = this.vehicleService.quantity;
32   total = this.vehicleService.total;
33   color = this.vehicleService.color;

```



```
5 @Component([
6   selector: 'app-vehicle-selection',
7   imports: [ FormsModule ],
8   template: `
9     <h1>Star Wars Vehicle Sales</h1>
10    @if (isLoading) {
11      <div>...loading vehicles</div>
12    } @else {
13      <select class="select" [(ngModel)]="selectedVehicle">
14        <option value="undefined" disabled selected>--Select a vehicle--</option>
15        @for(vehicle of vehicles()); track vehicle) {
16          <option [ngValue]=`vehicle`>{{ vehicle.name }}</option>
17        }
18      </select>
19      <div>Quantity: <input type='number' [(ngModel)]='quantity' ></div>
20      <div>Vehicle: {{ selectedVehicle()?.name }}</div>
21      <div>Price: {{ selectedVehicle()?.cost_in_credits }}</div>
22      <div [style.color]`color()`>Total: {{ total() }}</div>
23    }
24  `,
25  styles: ``
26])
27 export class VehicleSelectionComponent {
28   // Injected services
29   private vehicleService = inject(VehicleService);
30
31   // Signals to support the template
32   vehicles = this.vehicleService.vehicles;
33   isLoading = this.vehicleService.isLoading;
```



Star Wars Vehicle Sales

...loading vehicles

```
5 @Component([
6   selector: 'app-vehicle-selection',
7   imports: [ FormsModule ],
8   template: `
9     <h1>Star Wars Vehicle Sales</h1>
10    @if (isLoading) {
11      <div>...loading vehicles</div>
12    } @else {
13      <select class="select" [(ngModel)]="selectedVehicle">
14        <option value="undefined" disabled selected>--Select a vehicle--</option>
15        @for(vehicle of vehicles()); track vehicle) {
16          <option [ngValue]=`vehicle`>{{ vehicle.name }}</option>
17        }
18      </select>
19      <div>Quantity: <input type='number' [(ngModel)]='quantity' ></div>
20      <div>Vehicle: {{ selectedVehicle()?.name }}</div>
21      <div>Price: {{ selectedVehicle()?.cost_in_credits }}</div>
22      <div [style.color]`color()`>Total: {{ total() }}</div>
23    }
24  `,
25  styles: ```
26])
27 export class VehicleSelectionComponent {
28   // Injected services
29   private vehicleService = inject(VehicleService);
30
31   // Signals to support the template
32   vehicles = this.vehicleService.vehicles;
33   isLoading = this.vehicleService.isLoading;
```

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle-selection/vehicle-selection.component.ts

Signals in Action (forked)

Share

vehicle.service.ts vehicle-selection.component.ts

```
5 @Component({
6   selector: 'app-vehicle-selection',
7   imports: [ FormsModule ],
8   template: `
9     <h1>Star Wars Vehicle Sales</h1>
10    <if (isLoading()) {
11      <div>...loading vehicles</div>
12    } @else {
13      <select class="select" [(ngModel)]="selectedVehicle">
14        <option value="undefined" disabled selected>--Select a vehicle--</option>
15        <for(vehicle of vehicles(); track vehicle) {
16          <option [ngValue]=`vehicle`>{{ vehicle.name }}</option>
17        }
18      </select>
19      <div>Quantity: <input type='number' [(ngModel)]='quantity' ></div>
20      <div>Vehicle: {{ selectedVehicle()?.name }}</div>
21      <div>Price: {{ selectedVehicle()?.cost_in_credits }}</div>
22      <div [style.color]='color()'>Total: {{ total() }}</div>
23    }
24  `,
25  styles: ``
```

export class VehicleSelectionComponent {
 // Injected services
 private vehicleService = inject(VehicleService);

 // Signals to support the template
 vehicles = this.vehicleService.vehicles;
 isLoading = this.vehicleService.isLoading;

Star Wars Vehicle Sales

--Select a vehicle--

Quantity: 0

Vehicle:

Price:

Total: 0

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdg?file=src/app/vehicle-selection/vehicle-selection.component.ts

Signals in Action (forked)

Share

vehicle.service.ts vehicle-selection.component.ts

```
5 @Component({
6   selector: 'app-vehicle-selection',
7   imports: [ FormsModule ],
8   template: `
9     <h1>Star Wars Vehicle Sales</h1>
10    <if (isLoading()) {
11      <div>...loading vehicles</div>
12    } @else {
13      <select class="select" [(ngModel)]="selectedVehicle">
14        <option value="undefined" disabled selected>--Select a vehicle--</option>
15        <for(vehicle of vehicles(); track vehicle) {
16          <option [ngValue]=`vehicle`>{{ vehicle.name }}</option>
17        }
18      </select>
19      <div>Quantity: <input type='number' [(ngModel)]='quantity' ></div>
20      <div>Vehicle: {{ selectedVehicle()?.name }}</div>
21      <div>Price: {{ selectedVehicle()?.cost_in_credits }}</div>
22      <div [style.color]='color()'>Total: {{ total() }}</div>
23    }
24  `,
25  styles: ``
```

export class VehicleSelectionComponent {
 // Injected services
 private vehicleService = inject(VehicleService);

 // Signals to support the template
 vehicles = this.vehicleService.vehicles;
 isLoading = this.vehicleService.isLoading;

Star Wars Vehicle Sales

--Select a vehicle--

-Select a vehicle-

- Sand Crawler
- T-16 skyhopper
- X-34 landspeeder
- TIE/LN starfighter
- Snowspeeder
- TIE bomber
- AT-AT
- AT-ST
- Storm IV Twin-Pod cloud car
- Sail barge

stackblitz.com/~edit/signals-in-action-deborahk-6yeovmdq?file=src/app/vehicle.service.ts

Save Fork Signals in Action (forked) Share

vehicle.service.ts vehicle-selection.component.ts

```
28
29 // Computed signals
30 total = computed(() => (this.selectedVehicle()?.cost_in_credits ?? 0) * this.quantity());
31 color = computed(() => this.total() > 50000 ? 'green' : 'blue');
32
33 private vehiclesResource = rxResource({
34   loader: () => this.http.get<VehicleResponse>(this.vehicleUrl).pipe(
35     map(vr => vr.results),
36     delay(2000)
37   )
38 });
39 vehicles = computed(() => this.vehiclesResource.value() ?? [] as Vehicle[]);
40 isLoading = this.vehiclesResource.isLoading;
41 }
42
43 export interface VehicleResponse {
44   count: number;
45   next: string;
46   previous: string;
47   results: Vehicle[]
48 }
49
50 export interface Vehicle [
51   name: string;
52   cost_in_credits: number;
53 ]
```

Star Wars Vehicle Sales

--Select a vehicle--
Quantity: 0
Vehicle:
Price:
Total: 0

