

???? ?????? ??????????????: router.beforeEach((to) => { ... });

`router.beforeEach((to) => { ... })` — это **глобальный навигационный гард (guard)** во **Vue Router**.

Проще:

“ `□` код, который выполняется **ПЕРЕД** каждым переходом по роуту

???? ?????? ??????????????

```
router.beforeEach((to) => {  
  // ...  
})
```

Каждый раз, когда пользователь:

- переходит по ссылке
- вызывает `router.push()`
- обновляет страницу

Vue Router:

- 1 `□` собирается перейти на новый маршрут
- 2 `□` **вызывает** `beforeEach`
- 3 `□` ждёт его результат
- 4 `□` либо продолжает переход, либо отменяет / перенаправляет

???????????

Полная сигнатура:

```
router.beforeEach((to, from) => {})
```

- `to` — куда идём
- `from` — откуда идём

Чаще используют `to`.

???? ?????? ??????? ????????

? ?????????? ????????

```
return true
```

или ничего не возвращать

? ??????????????????

```
return { path: '/login' }
```

? ??????????????

```
return false
```

????????? ??????? — ??????????????? ?

```
router.beforeEach((to) => {  
  if (to.meta.requiresAuth && !isAuth()) {  
    return { path: '/login' }  
  }  
})
```

```
{  
  path: '/profile',  
  component: Profile,  
  meta: { requiresAuth: true }  
}
```

```
}
```

????? ??????? ????????????????

□ Порядок:

1. `beforeEach`
2. `beforeResolve`
3. `afterEach`

□□ `beforeEach` — **самый ранний хук**

??????? ??????? ????????????????????

- проверка авторизации
- проверка ролей
- редиректы
- логирование
- смена title страницы

```
router.beforeEach((to) => {  
  document.title = to.meta.title || 'App'  
})
```

?????? ??

- Нельзя делать бесконечные редиректы
- Если возвращаешь объект → это **навигация**
- Если используешь `async` — можно `await`

```
router.beforeEach(async (to) => {  
  await checkAuth()  
})
```

???????

“ `router.beforeEach` — это **фильтр маршрутов**, который срабатывает **перед каждым переходом**”

Revision #1

Created 2026-01-19 05:16:57 UTC by Nikolay

Updated 2026-01-19 05:17:43 UTC by Nikolay