

# CREATE PAGE



← Back to Notes

## Create New Note

Title

Content

Create Note

### ***Code in CreatePage.jsx:***

```
import {useState} from 'react'
import {Link} from "react-router"
import { ArrowLeftIcon } from "lucide-react";

const CreatePage = () => {
  const [title, setTitle] = useState("")
  const [content, setContent] = useState("")
  // After submission, it set to true
  const [loading, setLoading] = useState(false)
```

```

const handleSubmit = () => {

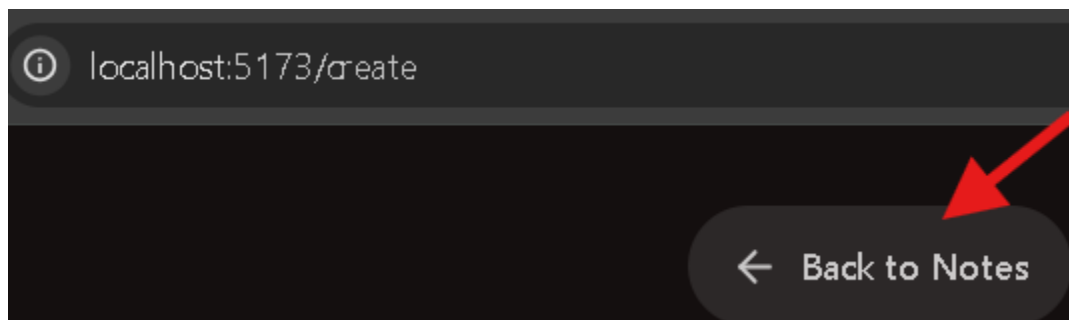
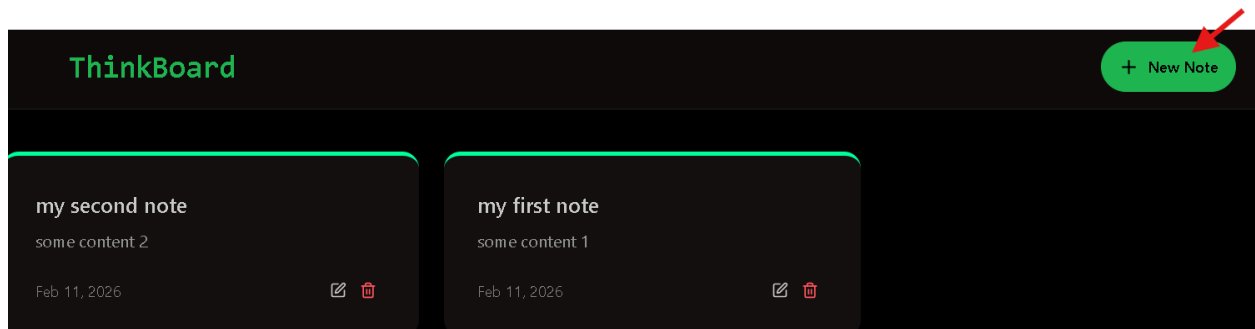
}

return <div className="min-h-screen bg-base-200">
  <div className="container mx-auto px-4 py-8">
    <div className="max-w-2xl mx-auto">
      <Link to={"/"} className="btn btn-ghost mb-6">
        <ArrowLeftIcon className="size-5" />
        Back to Notes
      </Link>
    </div>
  </div>
</div>
}

```

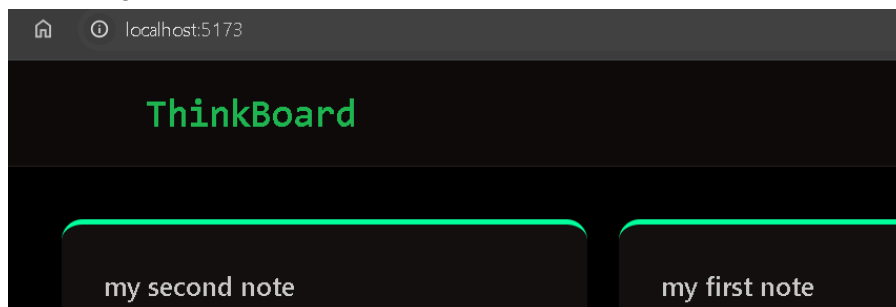
export default CreatePage

**In browser:**



get back to

home page:



## ***Code in CreatePage.jsx:***

```
import {useState} from 'react'
import {Link} from "react-router"
import { ArrowLeftIcon } from "lucide-react";

const CreatePage = () => {
  const [title, setTitle] = useState("")
  const [content, setContent] = useState("")
  // After submission, it set to true
  const [loading, setLoading] = useState(false)

  const handleSubmit = () => {

  }

  return <div className="min-h-screen bg-base-200">
    <div className="container mx-auto px-4 py-8">
      <div className="max-w-2xl mx-auto">
        <Link to={"/"} className="btn btn-ghost mb-6">
          <ArrowLeftIcon className="size-5" />
          Back to Notes
        </Link>

        <div className='card bg-base-100'>
          <div className='card-body'>
            <h2 className='card-title text-2xl mb-4'>Create New Note</h2>
            <form onSubmit={handleSubmit}>
              <div className='form-control mb-4'>
                <label className='label'>
                  <span className='label-text'>Title</span>
                </label>
                <input type="text"
                  placeholder='Note Title'
                  className="input input-bordered"
                  value={title}
                  onChange={(e) => setTitle(e.target.value)}
                />
              </div>
            </form>
          </div>
        </div>
      </div>
    </div>
  </div>
```

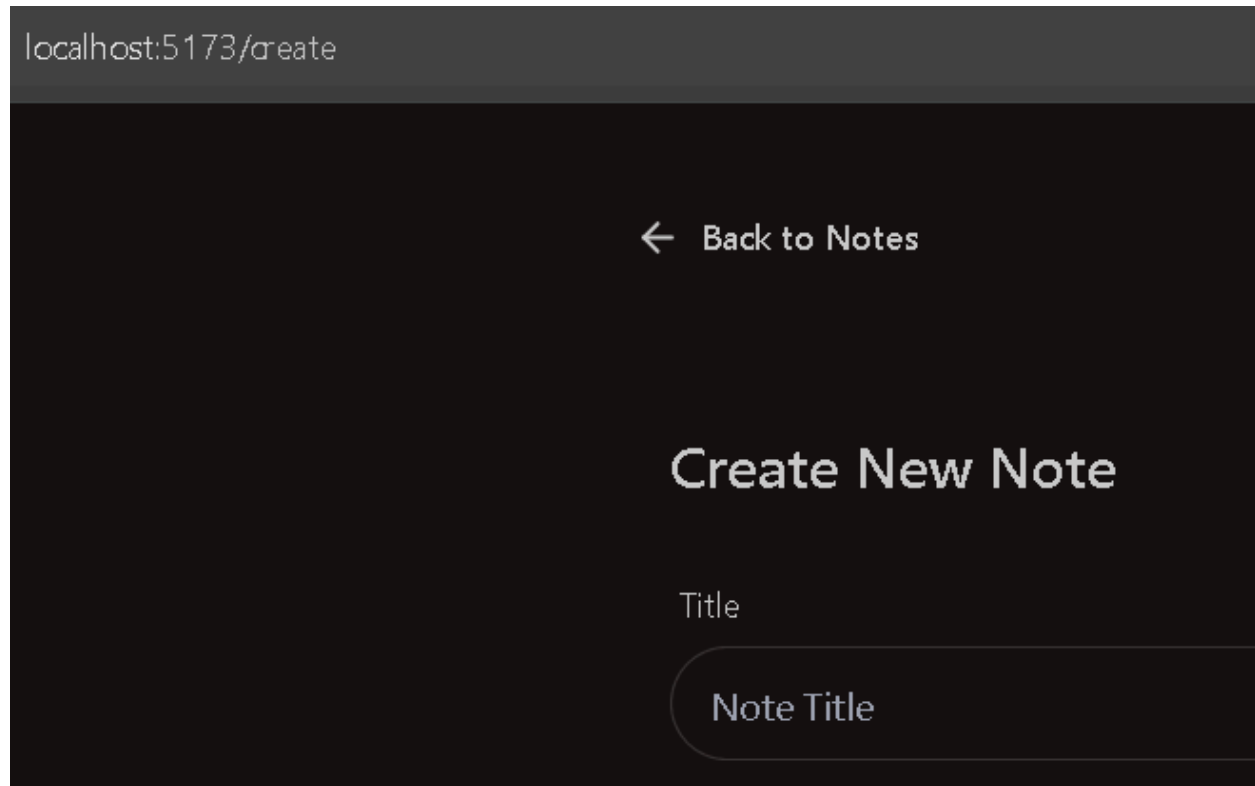
```

    </div>
  </div>
}

```

export default CreatePage

**In browser:**



### ***Code in CreatePage.jsx:***

```

import {useState} from 'react'
import {Link} from "react-router"
import { ArrowLeftIcon } from "lucide-react";

const CreatePage = () => {
  const [title, setTitle] = useState("")
  const [content, setContent] = useState("")
  // After submission, it set to true
  const [loading, setLoading] = useState(false)

  const handleSubmit = () => {

  }
}

```

```

return <div className="min-h-screen bg-base-200">
  <div className="container mx-auto px-4 py-8">
    <div className="max-w-2xl mx-auto">
      <Link to={"/"} className="btn btn-ghost mb-6">
        <ArrowLeftIcon className="size-5" />
        Back to Notes
      </Link>

      <div className='card bg-base-100'>
        <div className='card-body'>
          <h2 className='card-title text-2xl mb-4'>Create New Note</h2>
          <form onSubmit={handleSubmit}>
            <div className='form-control mb-4'>
              <label className='label'>
                <span className='label-text'>Title</span>
              </label>
              <input type="text"
                placeholder='Note Title'
                className="input input-bordered"
                value={title}
                onChange={(e) => setTitle(e.target.value)}
              />
            </div>

            <div className='form-control mb-4'>
              <label className='label'>
                <span className='label-text'>Content</span>
              </label>
              <textarea
                placeholder='Write your notes here...'
                className='textarea textarea-bordered h-32'
                value={content}
                onChange={(e) => setContent(e.target.value)}
              />
            </div>

            {/* justify-end <- to make the button available at right side
            disabled at loading state*/}
            <div className='card-actions justify-end'>
              <button type="submit" className='btn btn-primary' disabled={loading}>
                {loading? "Creating...": "Create Note"}
              </button>
            </div>
          </form>

```

```

    </div>

    </div>
  </div>
</div>
</div>
}

```

export default CreatePage

**In Browser:**

**For:**

```

8  // After submission, it set to true
9  const [loading, setLoading] = useState(true)

```

**In browser:**

Content

Write your notes here...

Creating...

```
// Called after submitting the form
const handleSubmit = () => {
  console.log(title)
  console.log(content)
}
```

## Create New Note

Title

first title

Content

some content

Create Note

After click on **Create Note**:

## Create New Note

Title

Note Title

Content

Write your notes here...

Create Note

But if we want to have the same text, even after Create Note.

**// Called after submitting the form**

```
const handleSubmit = (e) => {  
  e.preventDefault()  
  console.log(title)  
  console.log(content)  
}
```



## Create New Note

Title

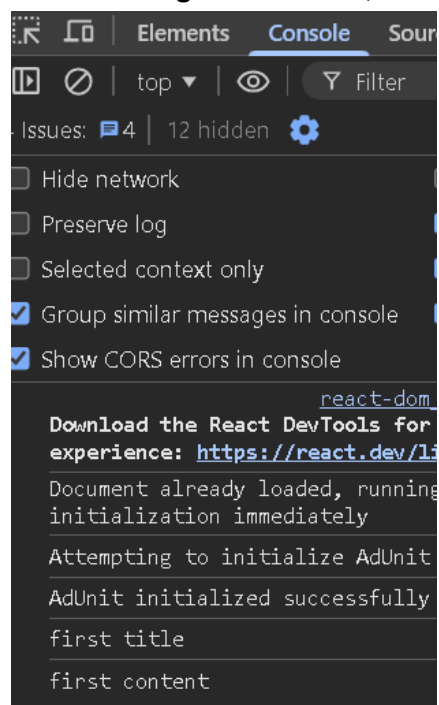
first title

Content

first content

Create Note

After clicking Create Note, remains same.



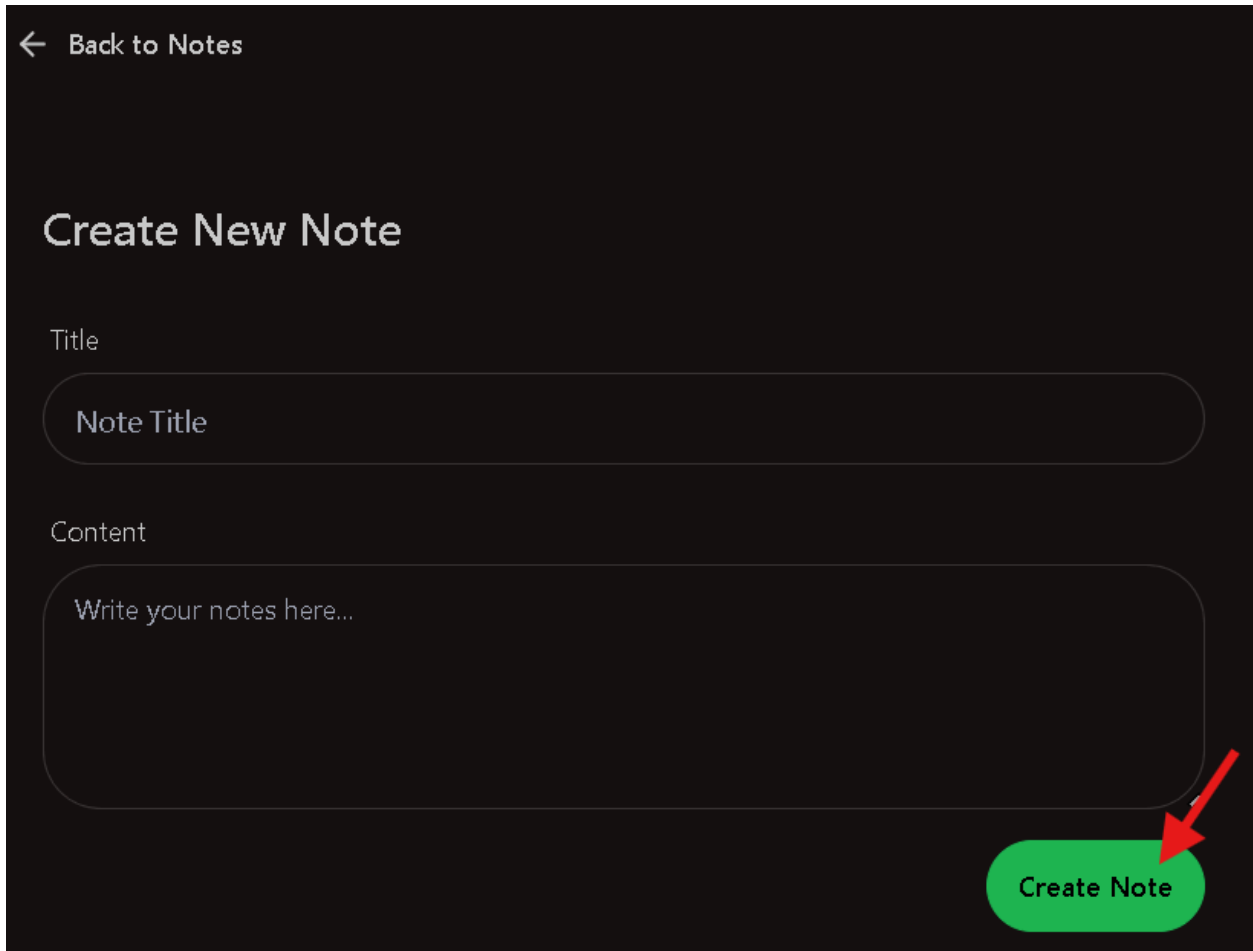
**For Code:**

```
const handleSubmit = (e) => {  
  e.preventDefault()  
  // console.log(title)
```

```
// console.log(content)

// If title or content are not provided, even for empty value(use .trim())
if (!title.trim() || !content.trim()){
  toast.error("All fields are Required")
  return
}
}
```

In browser:



← Back to Notes

## Create New Note

Title

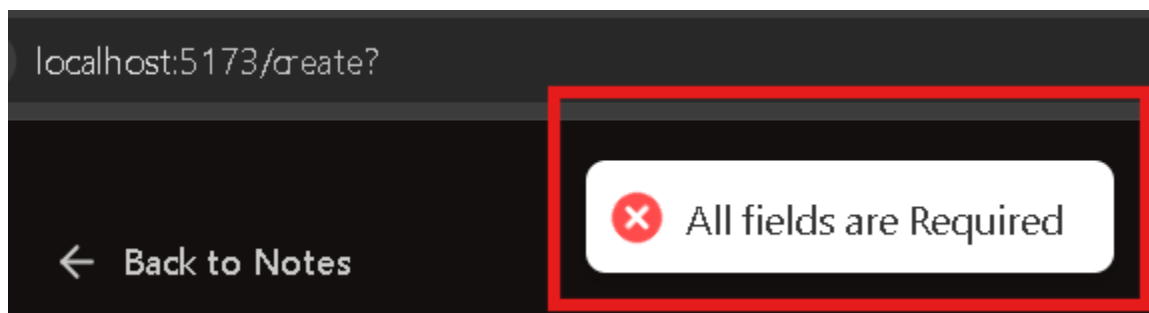
Note Title

Content

Write your notes here...

Create Note

Get's an toast error:



## ***Code in CreatePage.jsx:***

```
import {useState} from 'react'
import {Link, useNavigate} from "react-router"
import { ArrowLeftIcon } from "lucide-react";
import toast from "react-hot-toast"
import axios from "axios"

const CreatePage = () => {
  const [title, setTitle] = useState("")
  const [content, setContent] = useState("")
  // After submission, it set to true
  const [loading, setLoading] = useState(false)
  const navigate = useNavigate()
  // Called after submitting the form
  // async to use await
  const handleSubmit = async (e) => {
    e.preventDefault()
    // console.log(title)
    // console.log(content)

    // If title or content are not provided, even for empty value(use .trim())
    if (!title.trim() || !content.trim()){
      toast.error("All fields are Required")
      return
    }

    setLoading(true)
    try {
      // Send body {title, content}
      await axios.post("http://localhost:5001/api/notes", {title, content})
      toast.success("Note Created Successfully!")
      navigate("/")
    } catch (error) {
      console.log("Error creating note:", error)
      toast.error("Failed to Create Note")
    } finally{
      setLoading(false)
    }
  }

  return <div className="min-h-screen bg-base-200">
    <div className="container mx-auto px-4 py-8">
      <div className="max-w-2xl mx-auto">
```

```

<Link to={"/"} className="btn btn-ghost mb-6">
  <ArrowLeftIcon className="size-5" />
  Back to Notes
</Link>

<div className='card bg-base-100'>
  <div className='card-body'>
    <h2 className='card-title text-2xl mb-4'>Create New Note</h2>
    <form onSubmit={handleSubmit}>
      <div className='form-control mb-4'>
        <label className='label'>
          <span className='label-text'>Title</span>
        </label>
        <input type="text"
          placeholder='Note Title'
          className="input input-bordered"
          value={title}
          onChange={(e) => setTitle(e.target.value)}
        />
      </div>

      <div className='form-control mb-4'>
        <label className='label'>
          <span className='label-text'>Content</span>
        </label>
        <textarea
          placeholder='Write your notes here...'
          className='textarea textarea-bordered h-32'
          value={content}
          onChange={(e) => setContent(e.target.value)}
        />
      </div>

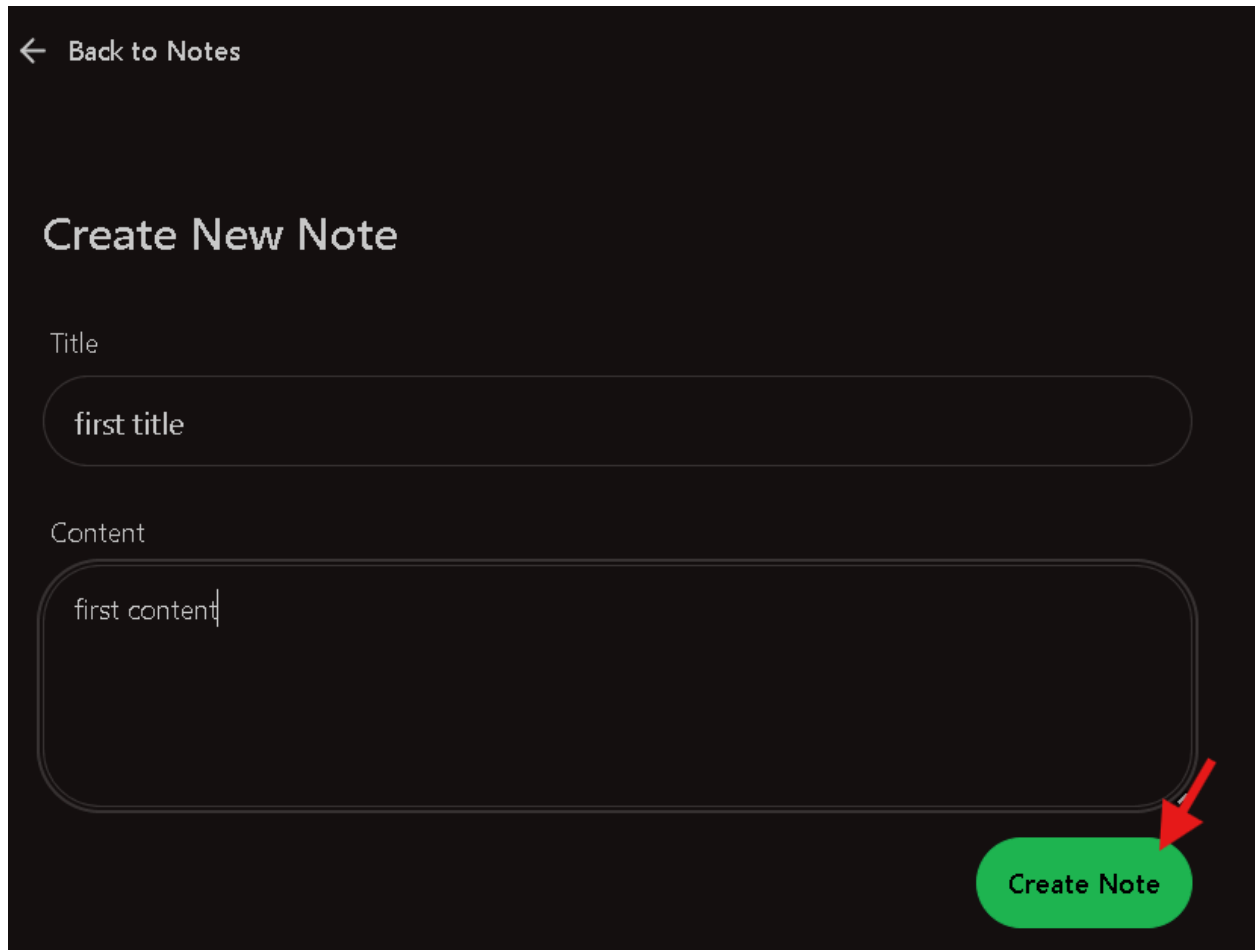
      {/* justify-end <- to make the button available at right side
      disabled at loading state*/}
      <div className='card-actions justify-end'>
        <button type="submit" className='btn btn-primary' disabled={loading}>
          {loading? "Creating...": "Create Note"}
        </button>
      </div>
    </form>
  </div>
</div>

```

```
    </div>  
  </div>  
</div>  
}
```

export default CreatePage

**In browser:**



← Back to Notes

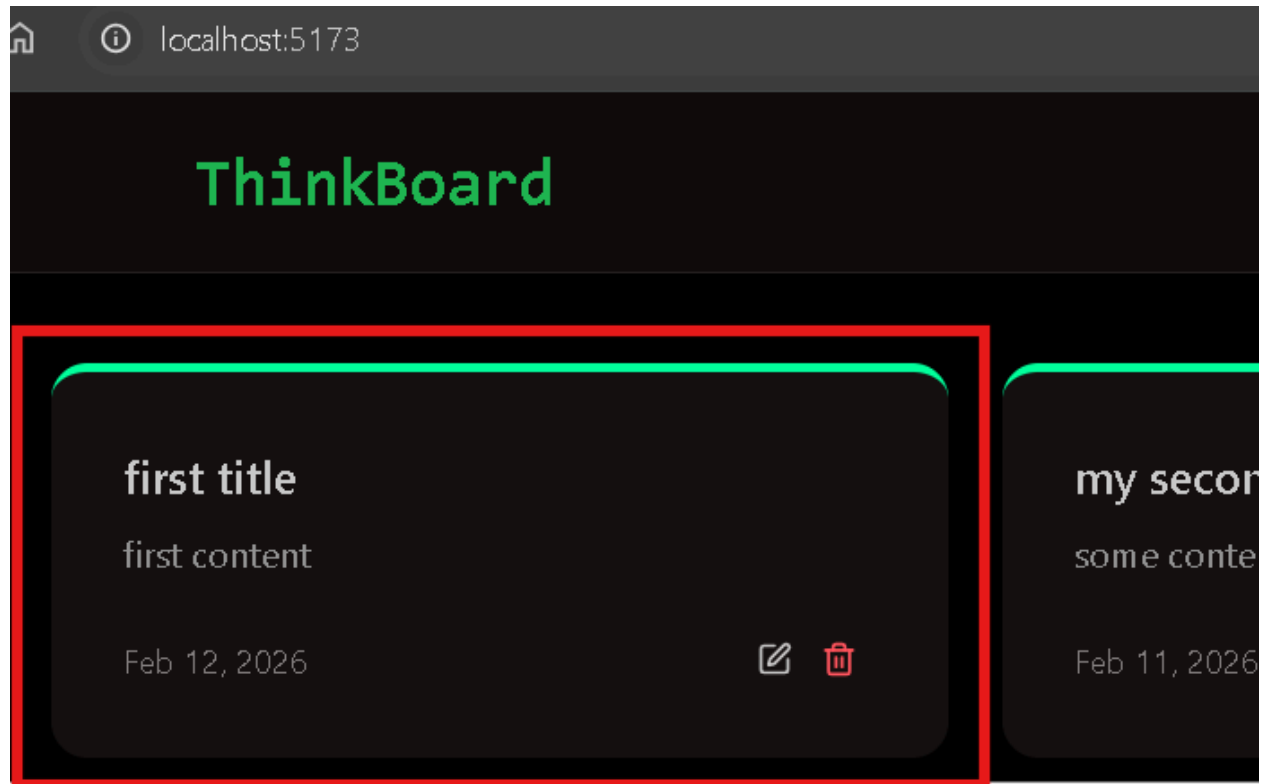
## Create New Note

Title

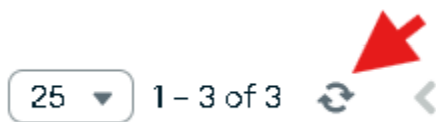
Content

Create Note

**Sends to the homepage, and adds the note:**



**In database:** Refresh it



**Adds into the Database:**

```
_id: ObjectId('698c0ab3f540242a5abb0379')
title: "my first note"
content: "some content 1"
createdAt: 2026-02-11T04:50:59.240+00:00
updatedAt: 2026-02-11T04:50:59.240+00:00
__v: 0
```

```
_id: ObjectId('698c0c048123dba253ad879c')
title: "my second note"
content: "some content 2"
createdAt: 2026-02-11T04:56:36.167+00:00
updatedAt: 2026-02-11T04:56:36.167+00:00
__v: 0
```

```
_id: ObjectId('698de47c3d9b61901fc5d577')
title: "first title"
content: "first content"
createdAt: 2026-02-12T14:32:28.748+00:00
updatedAt: 2026-02-12T14:32:28.748+00:00
__v: 0
```

If these are commented:

```
// If title or content are not provided, even for
// if (!title.trim() || !content.trim()) {
//   toast.error("All fields are Required")
//   return
// }
```



← Back to Notes

## Create New Note

Title

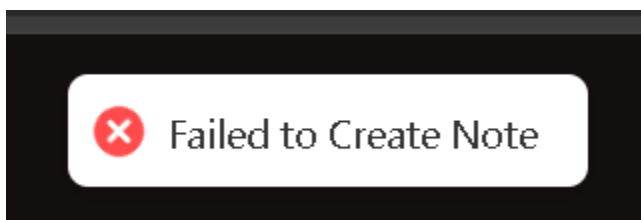
Note Title

Content

Write your notes here...

Create Note

Got toast error:



In console tab(after ctrl + shift + c):

```
✖ Failed to load resource: the server responded with a status of 500 :5001/api/notes:1 (Internal Server Error)
Error creating note: CreatePage.jsx:33
  ▼ AxiosError ⓘ
    code: "ERR_BAD_RESPONSE"
    ▶ config: {transitional: {...}, adapter: Array(3), transformRequest: Array(1), transformR
    message: "Request failed with status code 500"
    name: "AxiosError"
    ▶ request: XMLHttpRequest {onreadystatechange: null, readyState: 4, timeout: 0, withCre
    ▶ response: {data: {...}, status: 500, statusText: 'Internal Server Error', headers: Axio
    status: 500
    stack: "AxiosError: Request failed with status code 500\n    at settle (http://localh
    ▶ 55D...:1:1: Error
```



If not commented:

```
// If title or content are not provided,  
if (!title.trim() || !content.trim()){  
  toast.error("All fields are Required")  
  return  
}
```

← Back to Notes

## Create New Note

Title

Content

Write your notes here...

Create Note

Got toast error:

← Back to Notes

✕ All fields are Required

**What if user, clicks Create Note multiple times during empty fields.** Then its better to provide rate limiting:

### ***Code in CreatePage.jsx:***

```
import {useState} from 'react'
import {Link, useNavigate} from "react-router"
import { ArrowLeftIcon } from "lucide-react";
import toast from "react-hot-toast"
import axios from "axios"

const CreatePage = () => {
  const [title, setTitle] = useState("")
  const [content, setContent] = useState("")
  // After submission, it set to true
  const [loading, setLoading] = useState(false)
  const navigate = useNavigate()
  // Called after submitting the form
  // async to use await
  const handleSubmit = async (e) => {
    e.preventDefault()
    // console.log(title)
    // console.log(content)

    // If title or content are not provided, even for empty value(use .trim())
    // if (!title.trim() || !content.trim()){
    //   toast.error("All fields are Required")
    //   return
    // }

    setLoading(true)
    try {
      // Send body {title, content}
      await axios.post("http://localhost:5001/api/notes", {title, content})
      toast.success("Note Created Successfully!")
      navigate("/")
    } catch (error) {
      console.log("Error creating note:", error)
      // toast.error("Failed to Create Note")
      // 429 for rate limiting
      // 4000 milli Sec = 4 Sec
      if(error.response.status === 429){
        toast.error("Slow down! You're Creating Notes too fast", {
          duration: 4000,
```

```

        icon: "💀"
      })
    }else{
      toast.error("Failed to Create Note")
    }
  } finally{
    setLoading(false)
  }
}

return <div className="min-h-screen bg-base-200">
  <div className="container mx-auto px-4 py-8">
    <div className="max-w-2xl mx-auto">
      <Link to={"/"} className="btn btn-ghost mb-6">
        <ArrowLeftIcon className="size-5" />
        Back to Notes
      </Link>

      <div className='card bg-base-100'>
        <div className='card-body'>
          <h2 className='card-title text-2xl mb-4'>Create New Note</h2>
          <form onSubmit={handleSubmit}>
            <div className='form-control mb-4'>
              <label className='label'>
                <span className='label-text'>Title</span>
              </label>
              <input type="text"
                placeholder='Note Title'
                className="input input-bordered"
                value={title}
                onChange={(e) => setTitle(e.target.value)}
              />
            </div>

            <div className='form-control mb-4'>
              <label className='label'>
                <span className='label-text'>Content</span>
              </label>
              <textarea
                placeholder='Write your notes here...'
                className='textarea textarea-bordered h-32'
                value={content}
                onChange={(e) => setContent(e.target.value)}
              />
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>

```

```

    </div>

    {/* justify-end <- to make the button available at right side
    disabled at loading state*/}
    <div className='card-actions justify-end'>
      <button type="submit" className='btn btn-primary' disabled={loading}>
        {loading? "Creating...": "Create Note"}
      </button>
    </div>
  </form>
</div>

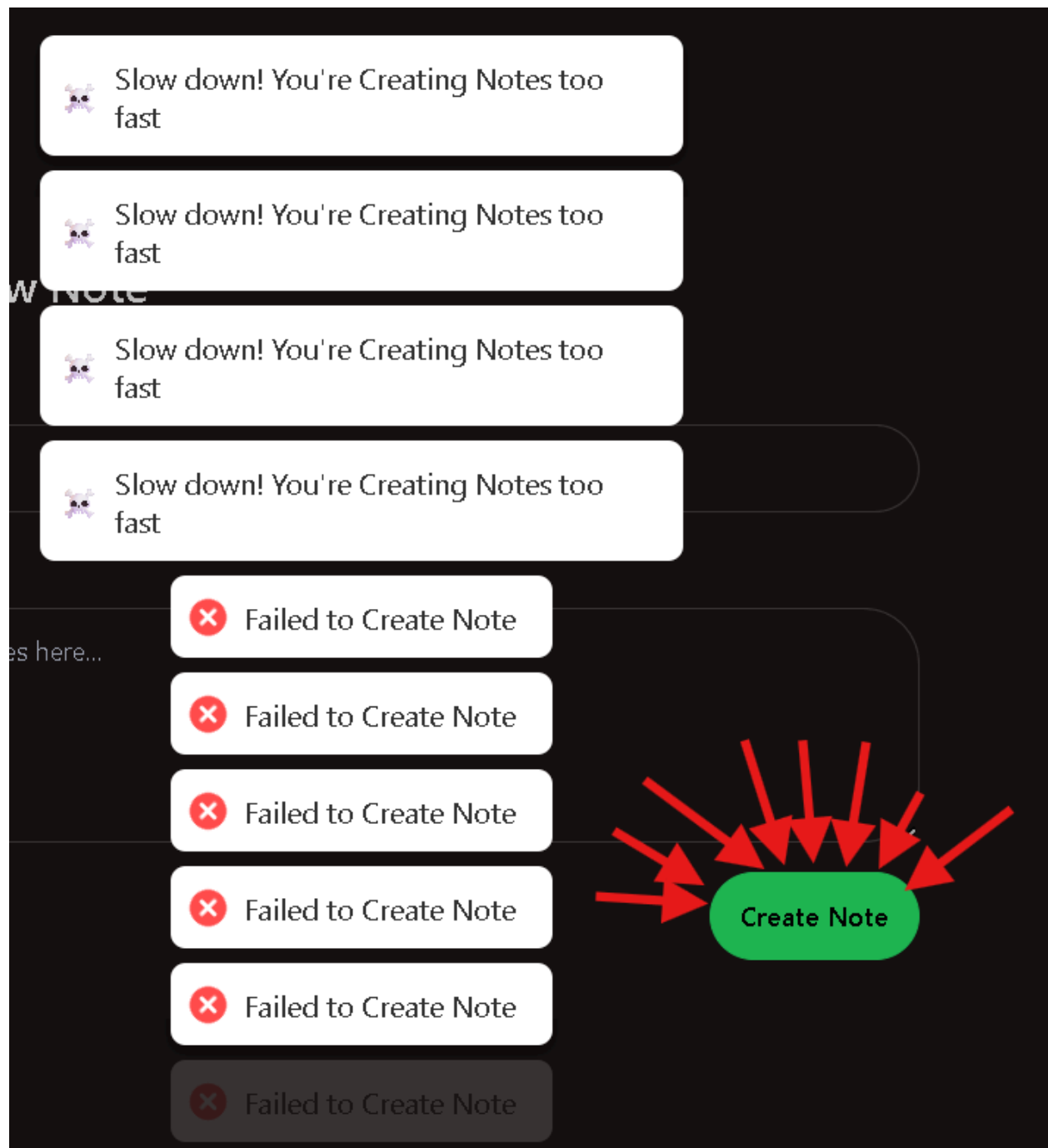
</div>
</div>
</div>
</div>
}

```

export default CreatePage

**Click multiple times:**

"Slow down! You're Creating Notes too fast" <- **Due to rate limiting**



Uncomment these:

```
// If title or content are not provided, even if (!title.trim() || !content.trim())){
  toast.error("All fields are Required")
  return
}
```

## Reason behind Creating [axios.js](#):

In CreatePage.jsx:



```
CreatePage.jsx x  HomePage.jsx
> frontend > src > pages > CreatePage.jsx > CreatePage > handleSubmit
7  const CreatePage = () => {
15  const handleSubmit = async (e) => {
    // Send body {title, content}
28    await axios.post("http://localhost:5001/api/notes", {title, content})
29  }
```

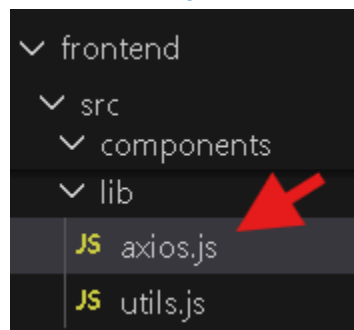
In HomePage.jsx:



```
HomePage.jsx x
> HomePage.jsx > HomePage > useEffect() callback > fetchNotes
const HomePage = () => {
  useEffect(() => {
    const fetchNotes = async () => {
      const res = await axios.get("http://localhost:5001/api/notes")
      // create data and update page
    }
  })
}
```

Creating a common thing among files as instance, and use it as a global instance. Most of the big projects use these.

Create [axios.js](#):



## Code in [axios.js](#):

import axios from "axios"

```
const api = axios.create({
  baseURL: "http://localhost:5001/api",
})
```

```
export default api
```

### ***Code in HomePage.jsx:***

```
import { useEffect } from 'react'
import { useState } from 'react';
import Navbar from '../components/Navbar'
import RateLimitedUI from '../components/RateLimitedUI';
import toast from "react-hot-toast";
import NoteCard from "../components/NoteCard"
import api from "../lib/axios"

const HomePage = () => {
  const [isRateLimited, setIsRateLimited] = useState(false);
  // fetching the notes
  const [notes, setNotes] = useState([])
  // to keep track of loading state, set to true, soon
  // as we load the page, we need to fetch the notes
  const [loading, setLoading] = useState(true)
  // To able to fetch that
  useEffect(() => {
    const fetchNotes = async () => {
      try {
        // For get method
        // const res = await axios.get("http://localhost:5001/api/notes")
        // For post method
        // const res = await axios.post("http://localhost:5001/api/notes")
        const res = await api.get("/notes")
        // const data = await res.json()
        // Which is lot more comfortable then fetch API
        console.log(res.data)// instead of console logging the data
        // We want to update the notes state
        setNotes(res.data)
        // If we can get the data, ratelimited to false
        setIsRateLimited(false)
      } catch (erro) {
        console.log("Error fetching notes")
        console.log(erro.response)
        // We want to check for the state in:
```

```

    // const [isRateLimited, setIsRateLimited] = useState(false);
    // 429 for rateLimited
    if(error.response?.status === 429){
      setIsRateLimited(true)
    }else{
      toast.error("Failed to load Notes")
    }
  } finally{
    setLoading(false)
  }
}

fetchNotes()
}, [])
return (
  // min-h-screen = minimum height will be screen
  // takes entire screen
  <div className="min-h-screen">
    {/* First it will have navbar component */}
    <Navbar />
    {isRateLimited && <RateLimitedUI />}

    <div className="max-w-7xl mx-auto p-4 mt-6">
      {loading && <div className="text-center text-primary py-10">Loading notes...</div>}
      {notes.length > 0 && !isRateLimited && (
        <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">
          {notes.map(note => (
            <NoteCard key={note._id} note={note} />
          ))}
        </div>
      )}
    </div>
  </div>
);
};

export default HomePage;

```

## ***Code in CreatePage.jsx:***

```

import {useState} from 'react'
import {Link, useNavigate} from "react-router"
import { ArrowLeftIcon } from "lucide-react";
import toast from "react-hot-toast"

```



**import api from "../lib/axios"**

```
const CreatePage = () => {
  const [title, setTitle] = useState("")
  const [content, setContent] = useState("")
  // After submission, it set to true
  const [loading, setLoading] = useState(false)
  const navigate = useNavigate()
  // Called after submitting the form
  // async to use await
  const handleSubmit = async (e) => {
    e.preventDefault()
    // console.log(title)
    // console.log(content)

    // If title or content are not provided, even for empty value(use .trim())
    if (!title.trim() || !content.trim()){
      toast.error("All fields are Required")
      return
    }

    setLoading(true)
    try {
      // Send body {title, content}
      // await axios.post("http://localhost:5001/api/notes", {title, content})
      await api.post("/notes", {title, content})
      toast.success("Note Created Successfully!")
      navigate("/")
    } catch (error) {
      console.log("Error creating note:", error)
      // toast.error("Failed to Create Note")
      // 429 for rate limiting
      // 4000 milli Sec = 4 Sec
      if(error.response.status === 429){
        toast.error("Slow down! You're Creating Notes too fast", {
          duration: 4000,
          icon: "💀"
        })
      }else{
        toast.error("Failed to Create Note")
      }
    } finally{
      setLoading(false)
    }
  }
}
```

```
}
```

```
return <div className="min-h-screen bg-base-200">
  <div className="container mx-auto px-4 py-8">
    <div className="max-w-2xl mx-auto">
      <Link to={"/"} className="btn btn-ghost mb-6">
        <ArrowLeftIcon className="size-5" />
        Back to Notes
      </Link>

      <div className='card bg-base-100'>
        <div className='card-body'>
          <h2 className='card-title text-2xl mb-4'>Create New Note</h2>
          <form onSubmit={handleSubmit}>
            <div className='form-control mb-4'>
              <label className='label'>
                <span className='label-text'>Title</span>
              </label>
              <input type="text"
                placeholder='Note Title'
                className="input input-bordered"
                value={title}
                onChange={(e) => setTitle(e.target.value)}
              />
            </div>

            <div className='form-control mb-4'>
              <label className='label'>
                <span className='label-text'>Content</span>
              </label>
              <textarea
                placeholder='Write your notes here...'
                className='textarea textarea-bordered h-32'
                value={content}
                onChange={(e) => setContent(e.target.value)}
              />
            </div>

            {/* justify-end <- to make the button available at right side
            disabled at loading state*/}
            <div className='card-actions justify-end'>
              <button type="submit" className='btn btn-primary' disabled={loading}>
                {loading? "Creating...": "Create Note"}
              </button>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>
```

```
    </div>
  </form>
</div>
```

```
  </div>
</div>
</div>
</div>
}
```

```
export default CreatePage
```