```
const fileChangeHandler = async (e) => {
  const file = e.target.files?.[0];
  if (file) {
    setFile(file);
    const dataUrl = await readFileAsDataURL(file);
    setImagePreview(dataUrl);
  }
}
```

Line

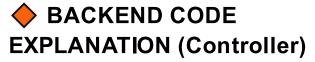
Explanation

```
e.target.files?.[0 Input se selected image file milti hai. [0] ka matlab first file.
 ]
                            File ko state me store kar liya (taaki formData me bhej sakein).
 setFile(file)
 readFileAsDataURL( File ko image preview ke liye base64 me convert kiya.
 file)
 setImagePreview(da
                            React state me preview image ka data store kiya. Ul me image
                            dikhane ke liye.
 taUrl)
const createPostHandler = async (e) => {
 const formData = new FormData();
 formData.append("caption", caption);
 if (imagePreview) formData.append("image", file);
 try {
  setLoading(true);
  const res = await axios.post('http://localhost:8080/api/v1/post/addpost', formData, {
   headers: { 'Content-Type': 'multipart/form-data' },
   withCredentials: true
  });
  if (res.data.success) {
   dispatch(setPosts([res.data.post, ...posts]));
   toast.success(res.data.message);
   setOpen(false);
 } catch (error) {
  toast.error(error.response.data.message);
 } finally {
  setLoading(false);
 }
}
```

Line

Explanation

FormData()	Ek special object banaya jo file + text dono bhej sakta hai.
<pre>append("caption", caption)</pre>	User ke likhe huye text ko bhejne ke liye.
<pre>append("image", file)</pre>	Selected image ko formData me bhej rahe hain.
<pre>axios.post()</pre>	API request backend ko bheji — controller trigger hua.
headers	Bataya ki yeh form-data hai (image/text dono hai).
withCredentials: true	Cookies bhejne ke liye (auth/session ke liye).
res.data.success	Backend se success response aaya toh post list update ho gaya.
<pre>res.data.success dispatch()</pre>	
	gaya.
dispatch()	gaya. Redux store me naye post ko daala.
<pre>dispatch() toast.success()</pre>	gaya. Redux store me naye post ko daala. Success toast dikhaya.
<pre>dispatch() toast.success() setOpen(false)</pre>	gaya. Redux store me naye post ko daala. Success toast dikhaya. Post modal close ho gaya.



js CopyEdit

```
export const addNewPost = async (req, res) => {
  try {
    const { caption } = req.body;
    const image = req.file;
    const authorId = req.id;

  if (!image) return res.status(400).json({ message: 'Image required' });

  const optimizedImageBuffer = await sharp(image.buffer)
    .resize({ width: 800, height: 800, fit: 'inside' })
```

```
.toFormat('jpeg', { quality: 80 })
   .toBuffer();
  const fileUri = `data:image/jpeg;base64,${optimizedImageBuffer.toString('base64')}`;
  const cloudResponse = await cloudinary.uploader.upload(fileUri);
  const post = await Post.create({
   caption,
   image: cloudResponse.secure_url,
   author: authorld
  });
  const user = await User.findByld(authorld);
  if (user) {
   user.posts.push(post._id);
   await user.save();
  }
  await post.populate({ path: 'author', select: '-password' });
  return res.status(201).json({
   message: 'New post added',
   post,
   success: true,
  });
 } catch (error) {
  console.log(error);
}
```

Line

Explanation

req.body.caption	Frontend se caption receive kiya.
req.file	Frontend se image receive ki.
req.id	Logged-in user ka ID (middleware ke through).
if (!image)	Agar image nahi mili toh error bhej diya.
sharp()	Image compress kiya: 800x800 max, JPEG format.
toBuffer()	lmage ko memory buffer me convert kiya.
base64	Cloudinary me base64 format me upload kiya.

```
cloudinary.uploader.uplo ad(...)

Post.create(...) MongoDB me nayi post banayi.

User.findById(...) User ko dhoonda jisne post banayi.

user.posts.push(...) Uske post list me new post add kiya.

post.populate(...) Author detail post ke saath embed kiya (except password).

res.status(201).json(... Frontend ko response bheja — post object ke saath.)
```