

```

1 import React, { useState, useCallback } from 'react';
2 import { Card, Button, Avatar, Popover, List, Comment } from 'antd';
3 import PropTypes from 'prop-types';
4 import { RetweetOutlined, HeartTwoTone, HeartOutlined, MessageOutlined, EllipsisOutlined }
  from '@ant-design/icons';
5 import styled from 'styled-components';
6 import Link from 'next/link';
7
8 import CommentForm from './CommentForm';
9 import PostCardContent from './PostCardContent';
10 import PostImages from './PostImages';
11 import FollowButton from './FollowButton';
12
13 const dummyComments = [{
14   User: {
15     nickname: 'nero',
16   },
17   content: '우와 개정판이 나왔군요~',
18 }, {
19   User: {
20     nickname: 'hero',
21   },
22   content: '얼른 사고싶어요~',
23 }];
24
25 const CardWrapper = styled.div`
26   margin-bottom: 20px;
27 `;
28
29 const PostCard = ({ post }) => {
30   const [commentFormOpened, setCommentFormOpened] = useState(false);
31   const [liked, setLiked] = useState(false);
32
33   const onToggleLike = useCallback(() => {
34     setLiked((prev) => !prev);
35   }, []);
36
37   const onToggleComment = useCallback(() => {
38     setCommentFormOpened((prev) => !prev);
39   }, []);
40
41   return (
42     <CardWrapper key={post.id}>
43       <Card
44         cover={post.Images[0]} && <PostImages images={post.Images} />
45         actions={[
46           <RetweetOutlined key="retweet" />,
47           liked
48             ? <HeartTwoTone twoToneColor="#eb2f96" key="heart" onClick={onToggleLike} />
49             : <HeartOutlined key="heart" onClick={onToggleLike} />,
50           <MessageOutlined key="message" onClick={onToggleComment} />,
51           <Popover
52             key="ellipsis"
53             content={(
54               <Button.Group>
55                 <Button>신고</Button>
56                 <Button>수정</Button>
57                 <Button danger>삭제</Button>
58               </Button.Group>

```

```

59     })
60   >
61     <EllipsisOutlined />
62   </Popover>,
63 ]}
64 extra={<FollowButton post={post} />}
65 >
66   <Card.Meta
67     avatar={<Avatar>{post.User.nickname[0]}</Avatar>}
68     title={post.User.nickname}
69     description={<PostCardContent postData={post.content} />}
70   />
71 </Card>
72 {commentFormOpened && (
73   <>
74     <CommentForm post={post} />
75     <List
76       header={` ${dummyComments.length} 댓글`}
77       itemLayout="horizontal"
78       dataSource={dummyComments}
79       renderItem={(item) => (
80         <li>
81           <Comment
82             author={item.User.nickname}
83             avatar={
84               <Link href={{ pathname: '/user', query: { id: item.User.id } }} as=
85               `/user/${item.User.id}`>
86                 <a><Avatar>{item.User.nickname[0]}</Avatar></a>
87               </Link>
88             )}
89             content={item.content}
90           </li>
91         )}
92       />
93     </>
94   )}
95 </CardWrapper>
96 );
97 };
98
99 PostCard.propTypes = {
100   post: PropTypes.shape({
101     id: PropTypes.number,
102     User: PropTypes.object,
103     content: PropTypes.string,
104     Images: PropTypes.arrayOf(PropTypes.shape({
105       src: PropTypes.string,
106     })),
107     createdAt: PropTypes.object,
108   }),
109 };
110
111 export default PostCard;
112

```