

Chat Room 정보 (1) 방 생성자, 방 설명

누가 생성한 방?

방에 대한 설명

가장 많은 글을 쓴 사람

누가 생성한 방?

```
<Card.Body>
  <img
    width={48}
    height={48}
    className="mr-3"
    src={chatRoom && chatRoom.createdBy.image}
    alt={chatRoom && chatRoom.createdBy.name}
  />
  {chatRoom && chatRoom.createdBy.name}
</Card.Body>
```

방에 대한 설명

```
<Card>
  <Card.Header style={{ padding: '0 1rem' }}>
    <Accordion.Toggle as={Button} variant="link" eventKey="0">
      Description
    </Accordion.Toggle>
  </Card.Header>
  <Accordion.Collapse eventKey="0">
    <Card.Body> {chatRoom && chatRoom.description}</Card.Body>
  </Accordion.Collapse>
</Card>
```

Chat Room 정보 (2) 각 사람당 글 쓴 개수

가장 글을 많이 쓴 사람

원하는 데이터 생성

```
addMessagesListeners = (chatRoomId) => {  
  let messagesArray = [];  
  this.setState({ messages: [] });  
  this.state.messagesRef.child(chatRoomId)  
    .messagesArray.push(DataSnapshot.val(  
      this.setState({  
        messages: messagesArray,  
        messageLoading: false  
      })  
    ));  
  this.userPostsCount(messagesArray);  
};
```

```
userPostsCount = messages => {  
  let userPosts = messages.reduce((acc, message) => {  
    if (message.user.name in acc) {  
      acc[message.user.name].count += 1;  
    } else {  
      acc[message.user.name] = {  
        image: message.user.image,  
        count: 1  
      };  
    }  
    return acc;  
  }, {});  
  console.log('userPosts :', userPosts)  
  this.props.dispatch(setUserPosts(userPosts));  
};
```

```
userPosts :  
▼ {123333: {...}, John Ahn: {...}, jaewon2: {...}} ⓘ  
  ► 123333: {image: "http://gravatar.com/avatar...",  
    ▼ John Ahn:  
      count: 1  
      image: "https://firebasestorage.googleapis.com/v1/b/chatroom-123333.appspot.com/o/John%20Ahn.jpg?alt=media&token=...",  
      ► __proto__: Object  
    ▼ jaewon2:  
      count: 1
```

```
image: "http://gravatar.com/avatar/24484d7
  ▶ __proto__: Object
  ▶ __proto__: Object
```

원하는 데이터 생성 후
그 데이터 보여주기

```
<Accordion.Collapse eventKey="0">
  <Card.Body>
    {userPosts && renderUserPosts(userPosts)}
  </Card.Body>
</Accordion.Collapse>
```

Count 큰 것부터 위에
서 아래로

```
const renderUserPosts = userPosts =>
  Object.entries(userPosts)
    .sort((a, b) => b[1].count - a[1].count)
    .map(([key, val], i) => (
      <Media key={i}>
        <img
          style={{ borderRadius: '25px' }}
          width={48}
          height={48}
          className="mr-3"
          src={val.image}
          alt={val.name}
        />
        <Media.Body>
          <h6>{key}</h6>
          <p>
            {val.count} 개
          </p>
        </Media.Body>
      </Media>
    ))
```

```
1 let numbers = [4, 2, 5, 1, 3];
2 numbers.sort((a, b) => a - b);
3 console.log(numbers);
4
5 // [1, 2, 3, 4, 5]
```

```
a
▼ (2) ["user2", {...}] ⓘ
  0: "user2"
  ▼ 1:
    count: 2
    image: "http://gravata"
    ► __proto__: Object
    length: 2
    ► __proto__: Array(0)
▼ (2) ["user1", {...}] ⓘ
  0: "user1"
  ▼ 1:
    count: 1
    image: "http://gravata"
    ► __proto__: Object
    length: 2
    ► __proto__: Array(0)
```

채팅방을
만든 사람 정보는
단체 방에서만 보여주기

```
{!isPrivateChatRoom &&
  <Col>
    <Accordion >
      <Card>
        <Card.Header style={{ paddin
          <Accordion.Toggle as={Bu
            Created By
        </Accordion.Toggle>
```