

초격차 패키지 Online.

React Native 앱 개발

PART1 | React Native 개념 및 개요

CH1 | React Native 소개

CH2 | Expo 세팅

CH3 | React Native 학습 시 주요 개념

CH4 | 본격 학습 전 안내

PART 2 | 입문

CH1 | 카카오톡 친구목록 클론코딩

CH2 | 계산기

CH3 | 투두리스트 + 달력

CH4 | 나만의 갤러리

CH5 | 카카오퍼스 클론코딩

CH6 | 번역앱

CH1. React Native 소개

1 수강 안내

CH1. React Native 소개

2 React Native 소개

React Native란?

2.

React Native 소개

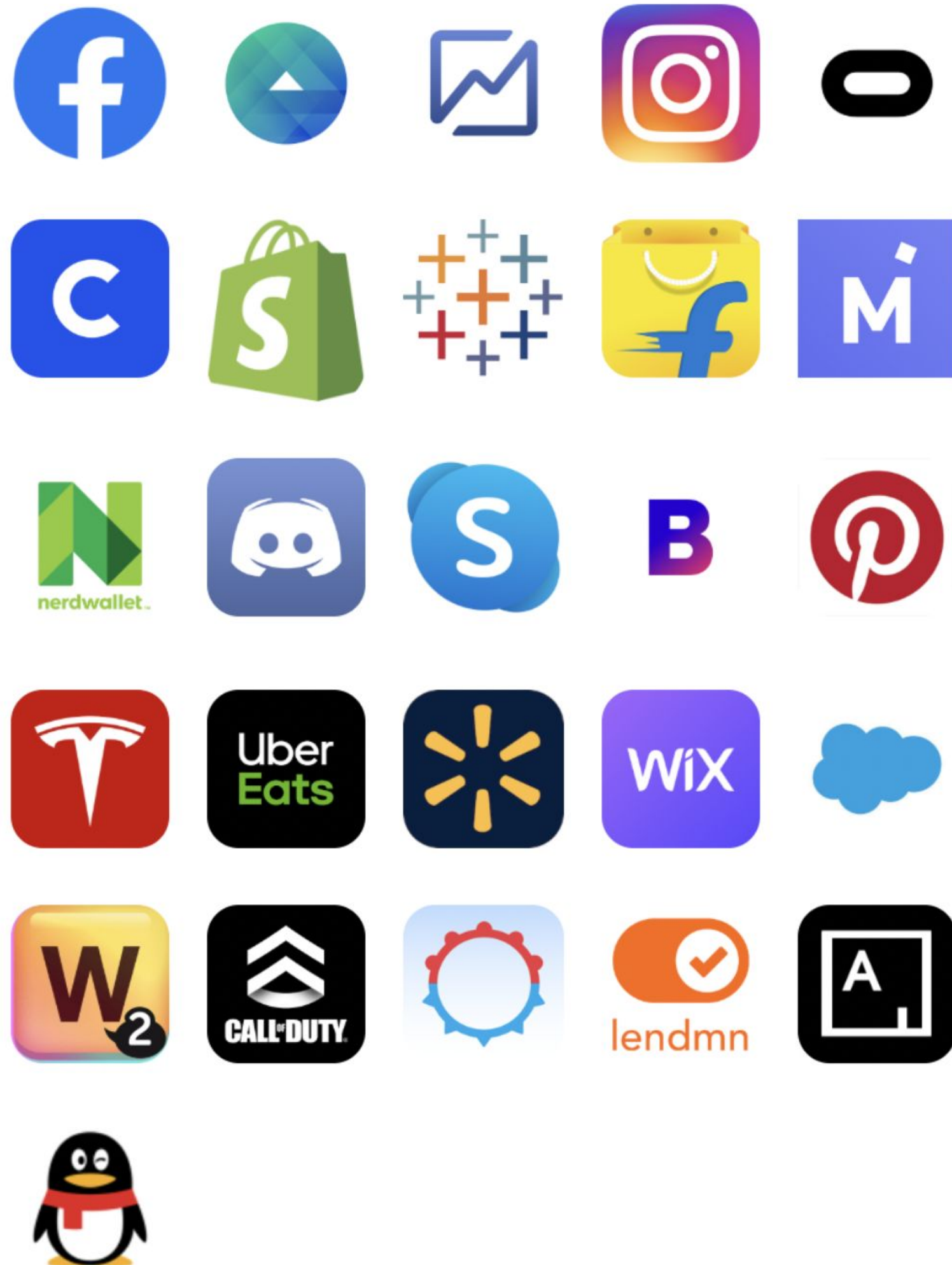
- 페이스북이 개발한 오픈 소스 모바일 애플리케이션 프레임워크
- Javascript 하나로 Android, iOS, Web 대응



React Native를 사용하는 앱

2.

React Native 소개

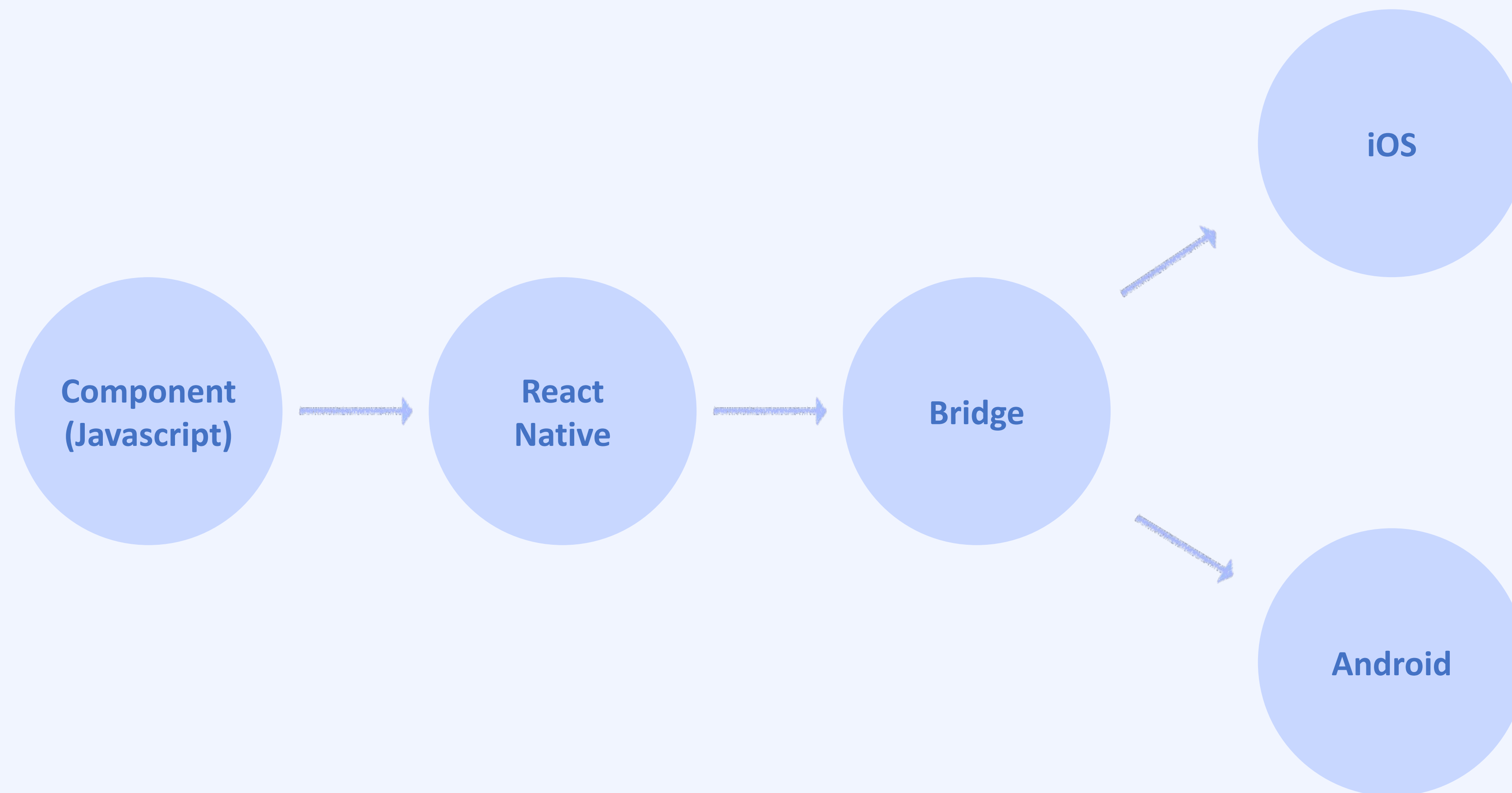


<https://reactnative.dev/showcase>

React Native 동작원리

2.

React Native 소개



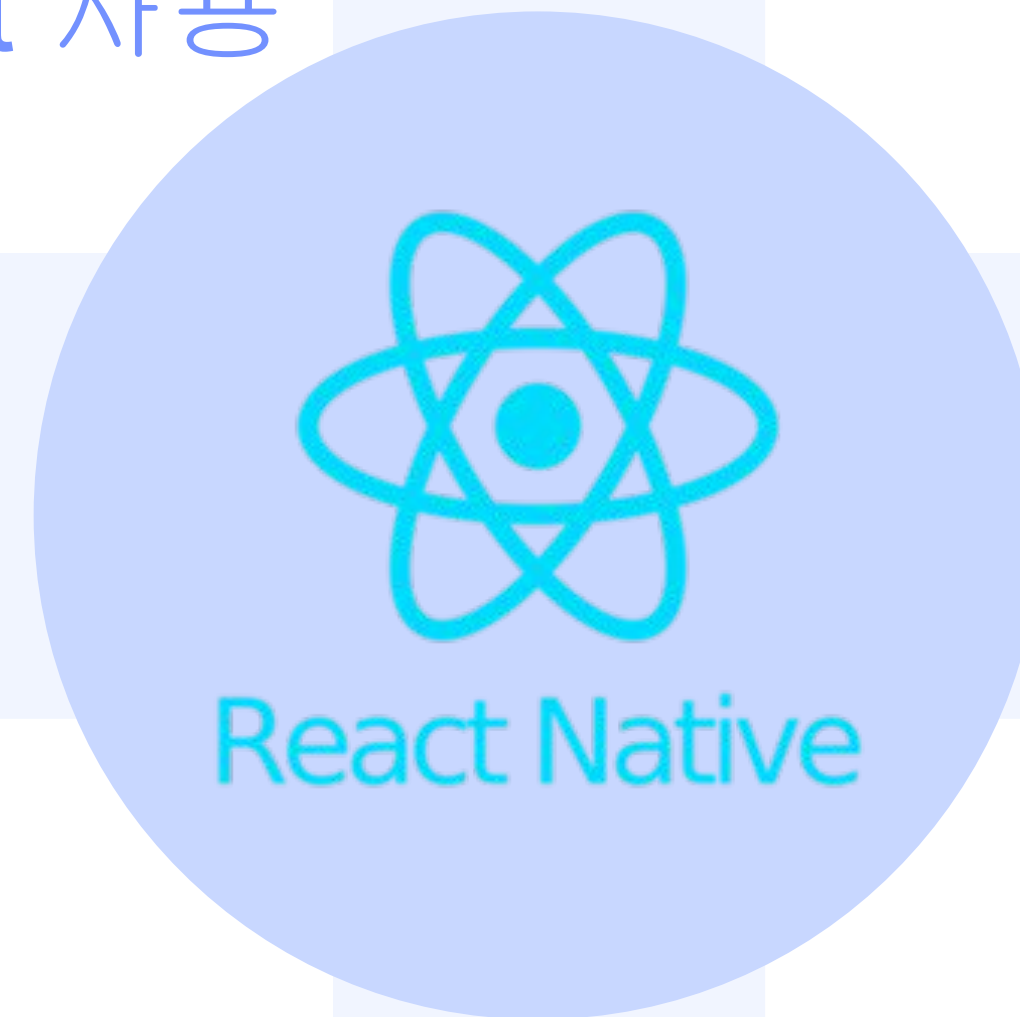
React Native 장점

2.

React Native 소개

하나의 코드로 관리,
러닝커브 높은 React 사용

코드푸시로 빠른 업데이트
-> 비용절감



Fast Refresh

오픈소스 플랫폼

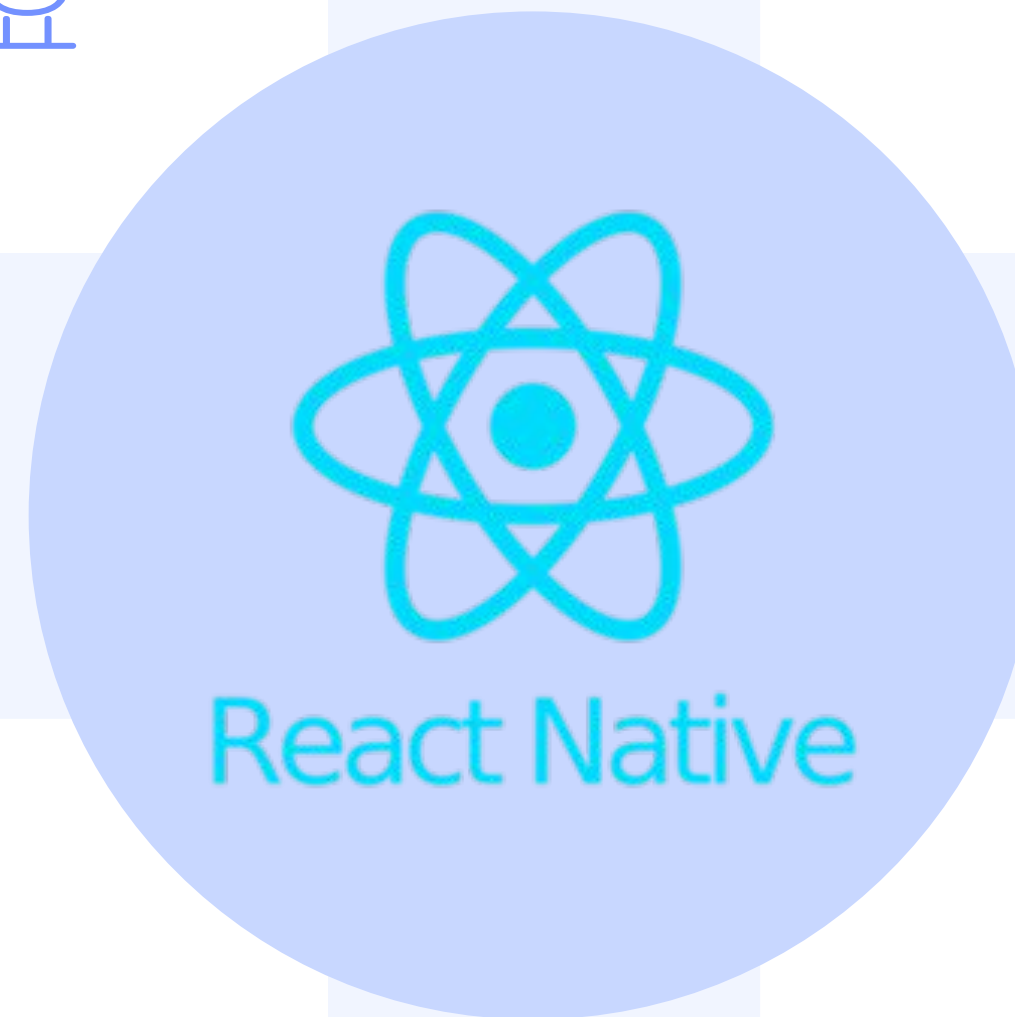
React Native 단점

2.

React Native 소개

일부 기능
Native 접근 필요

오픈소스, 라이브러리
의존도



성능

잦은 업데이트

CH1. React Native 소개

3 Expo CLI와 React Native CLI의
개념 및 비교설명

Expo Cli

3.

Expo CLI와
React Native CLI
개념 및 비교설명

- React Native 앱을 쉽고 빠르게 구축하는 종합선물세트
- 장점
 1. 기본제공되는 API, 라이브러리 -> 초반 앱 개발 단순화
 2. Expo Go 어플만 있으면 기기 상관없이 프로젝트 실행 가능
- 단점
 1. 추가 네이티브 모듈 사용 불가

React Native Cli

3.

Expo CLI와
React Native CLI
개념 및 비교설명

- 고도화된 기능 개발 및 높은 개발 자유도를 위해 선택한다!
- 장점
 1. 네이티브 모듈 연결 가능
 - > 다양한 라이브러리 사용으로 높은 자유도
- 단점
 1. 기본 제공되는 라이브러리가 적어, 필요한 것이 있다면 직접 설치
 2. Mac 개발 필수, Native 폴더 구조에¹¹대한 기본 지식 필요

Expo Cli vs React Native Cli 선택 시 고려사항

3.

Expo CLI와
React Native CLI
개념 및 비교설명

구현하고자 하는 모든 기능이 Expo에서 지원되는가?

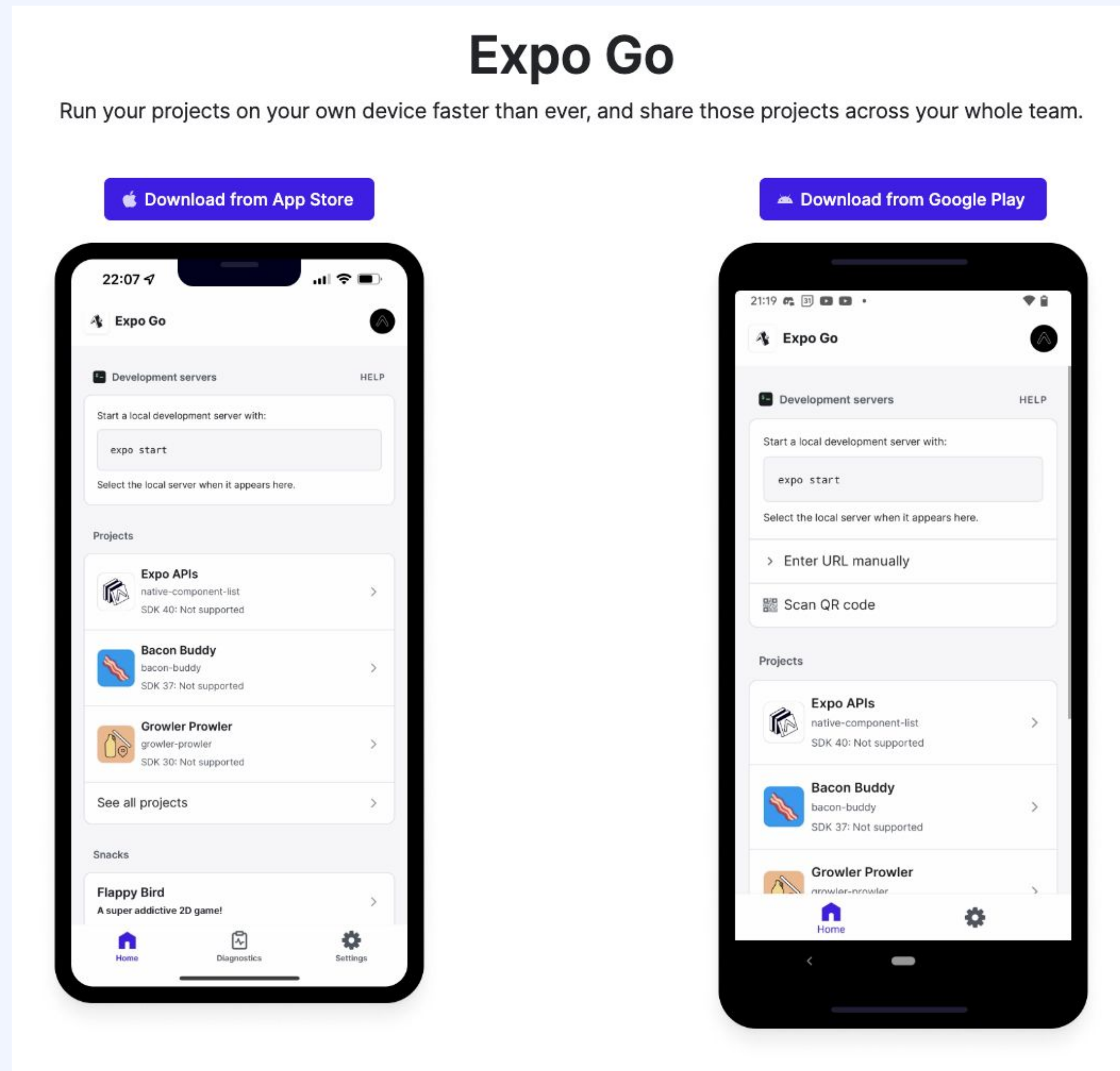
CH2. Expo 세팅

1 운영 체제에 따른 강의 안내

Expo Go 앱 설치

1.

운영체제에 따른
강의 안내



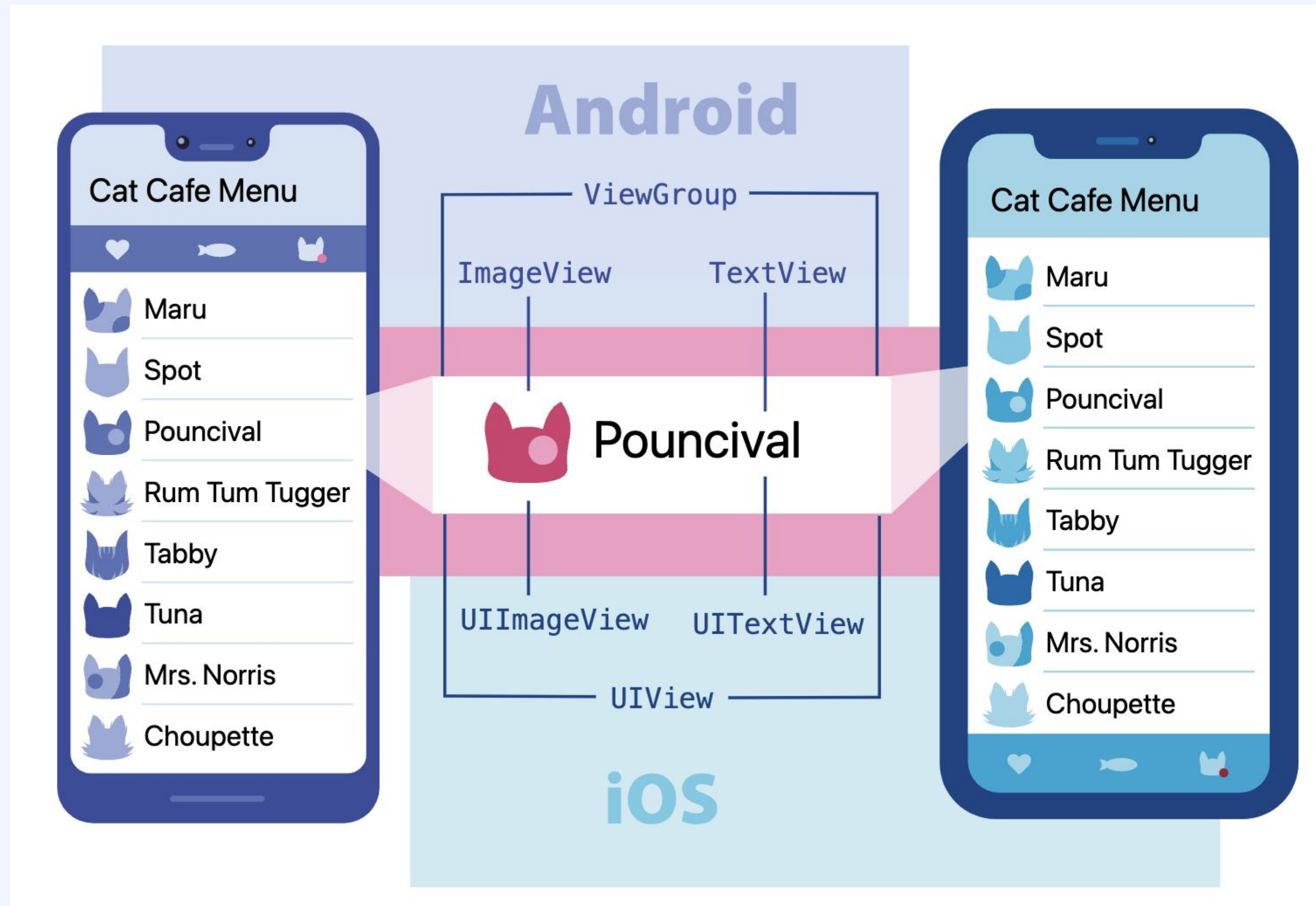
CH3. React Native 학습 시 주요 개념

1 Core Components

React Native에서 자주 쓰이는 Core Components

1.

Core Components



CH3. React Native 학습 시 주요 개념

2 컴포넌트 및 prop: 이론

컴포넌트의 의미

2.

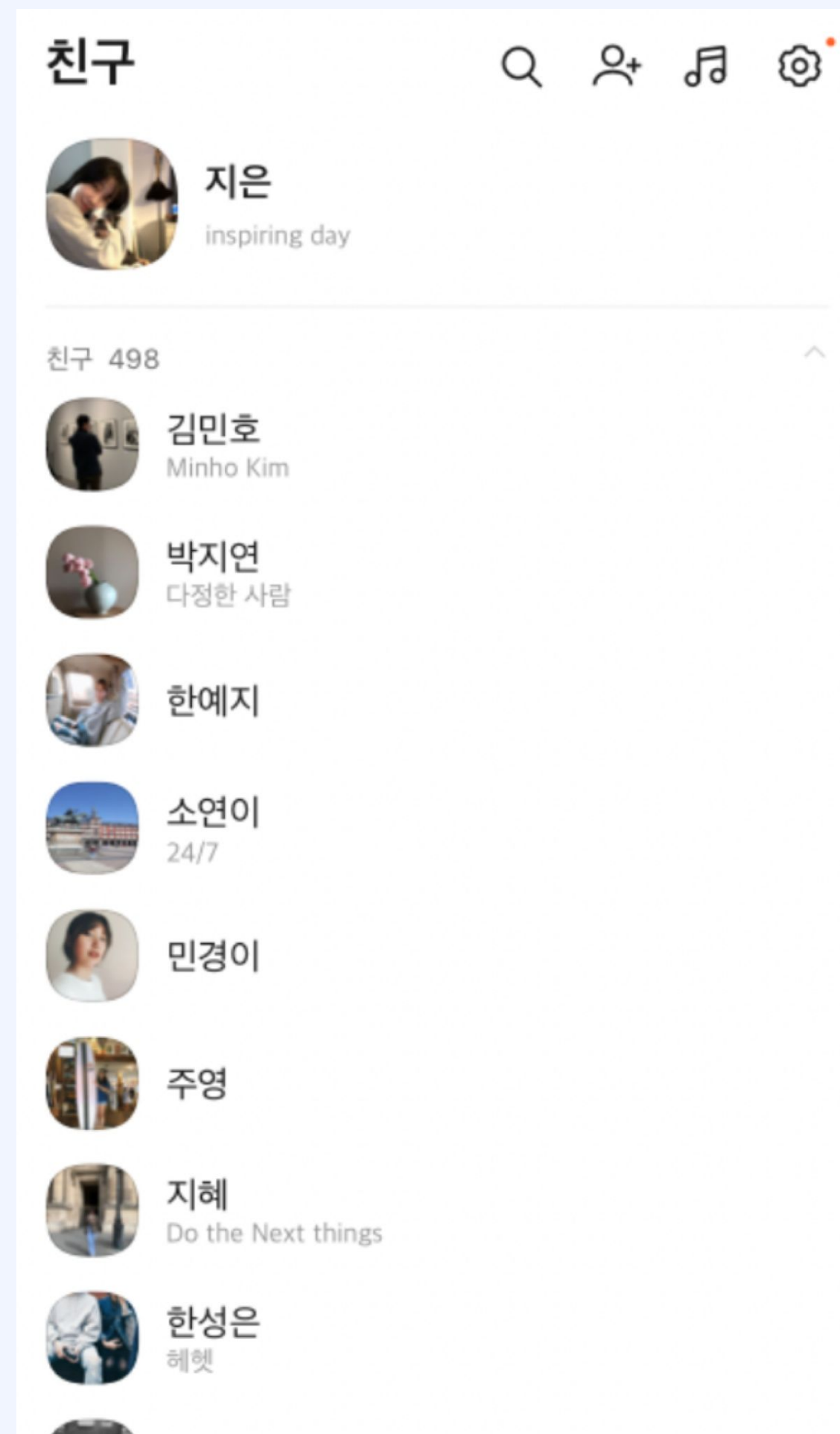
컴포넌트 및 prop
: 이론

“재사용 가능한 개별적인 여러 조각”

화면 예시) 카카오톡 친구목록

2.

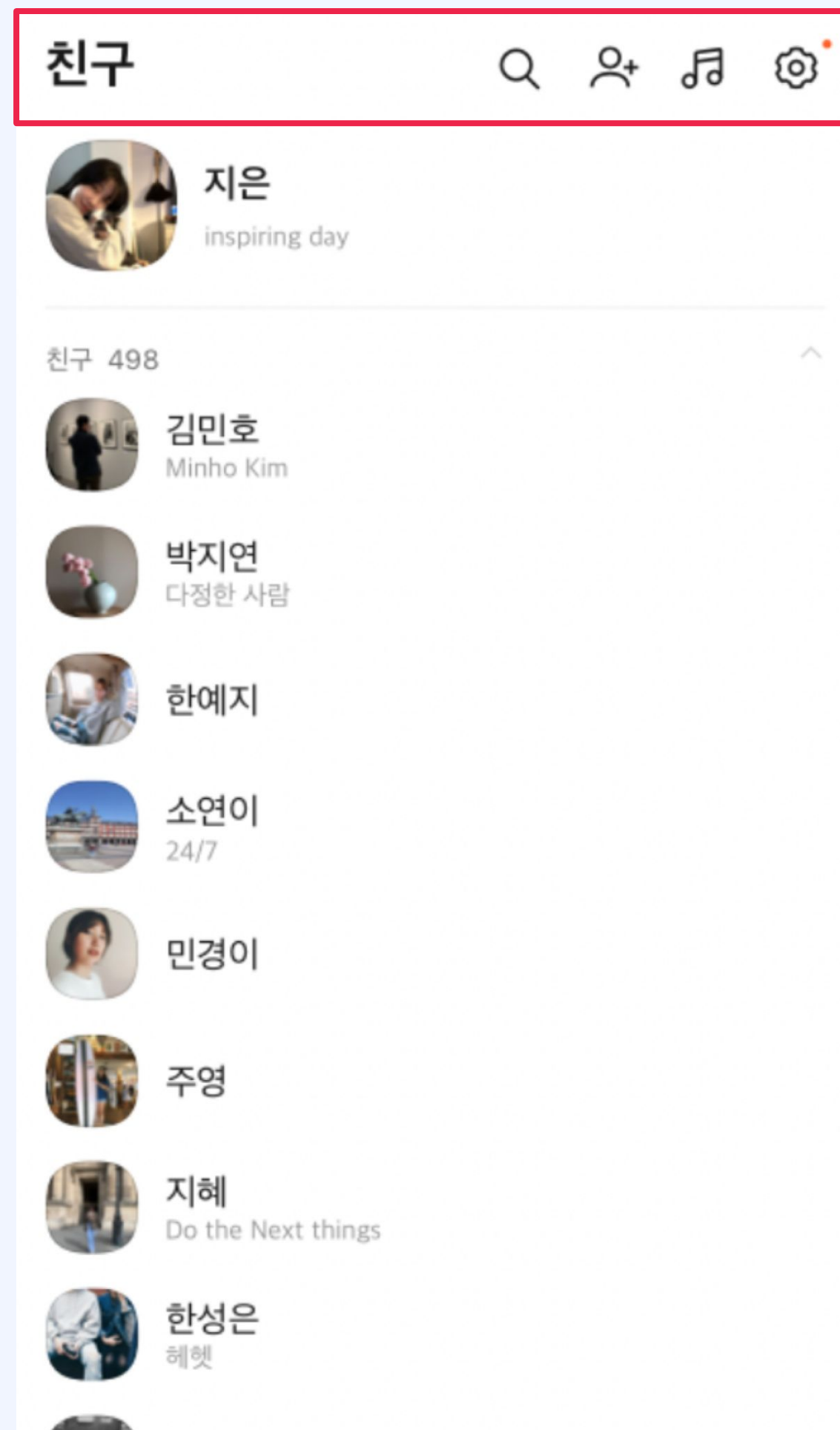
컴포넌트 및 prop
: 이론



화면 예시) 카카오톡 친구목록

2.

컴포넌트 및 prop
: 이론

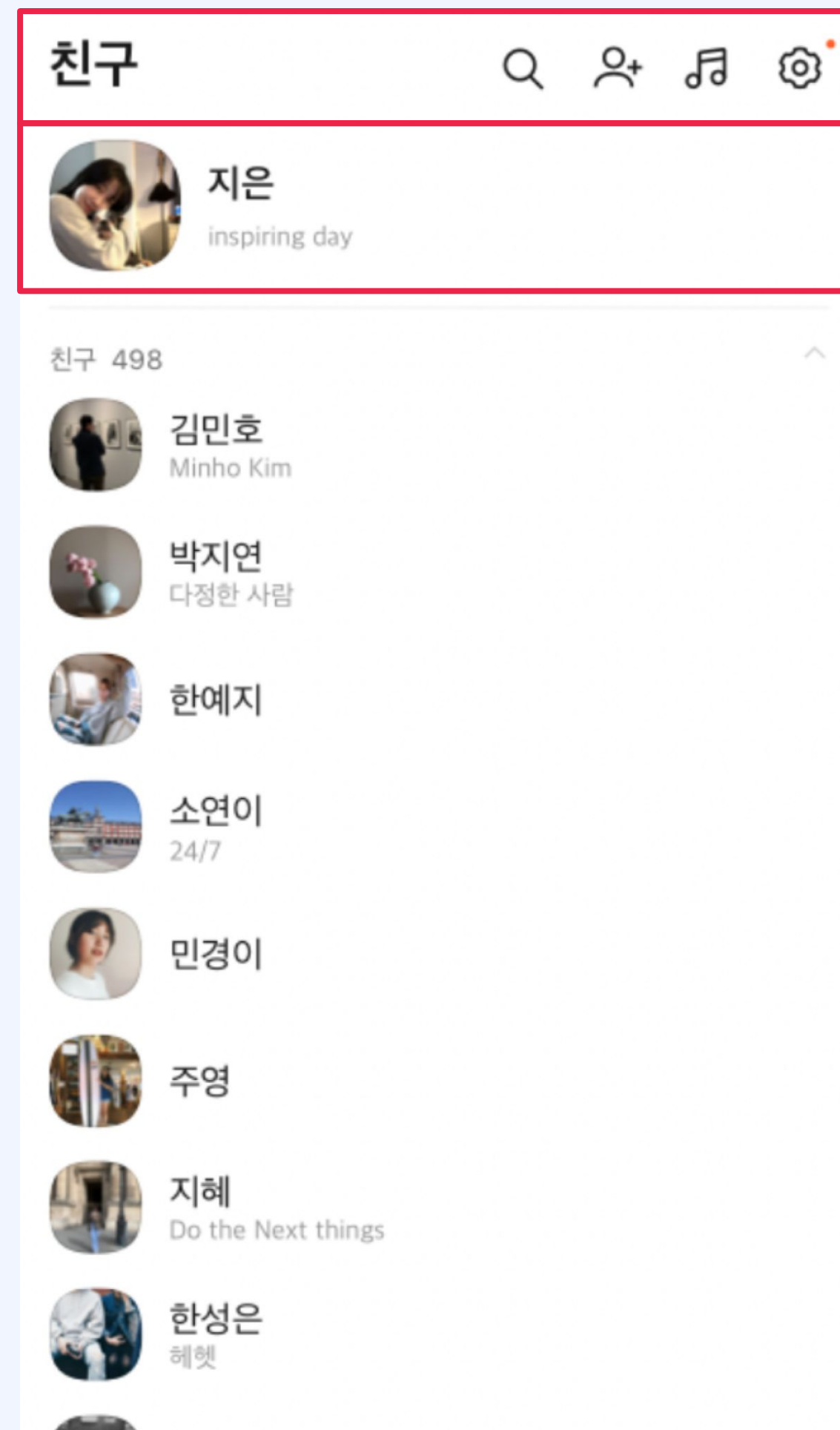


<Header />

화면 예시) 카카오톡 친구목록

2.

컴포넌트 및 prop
: 이론



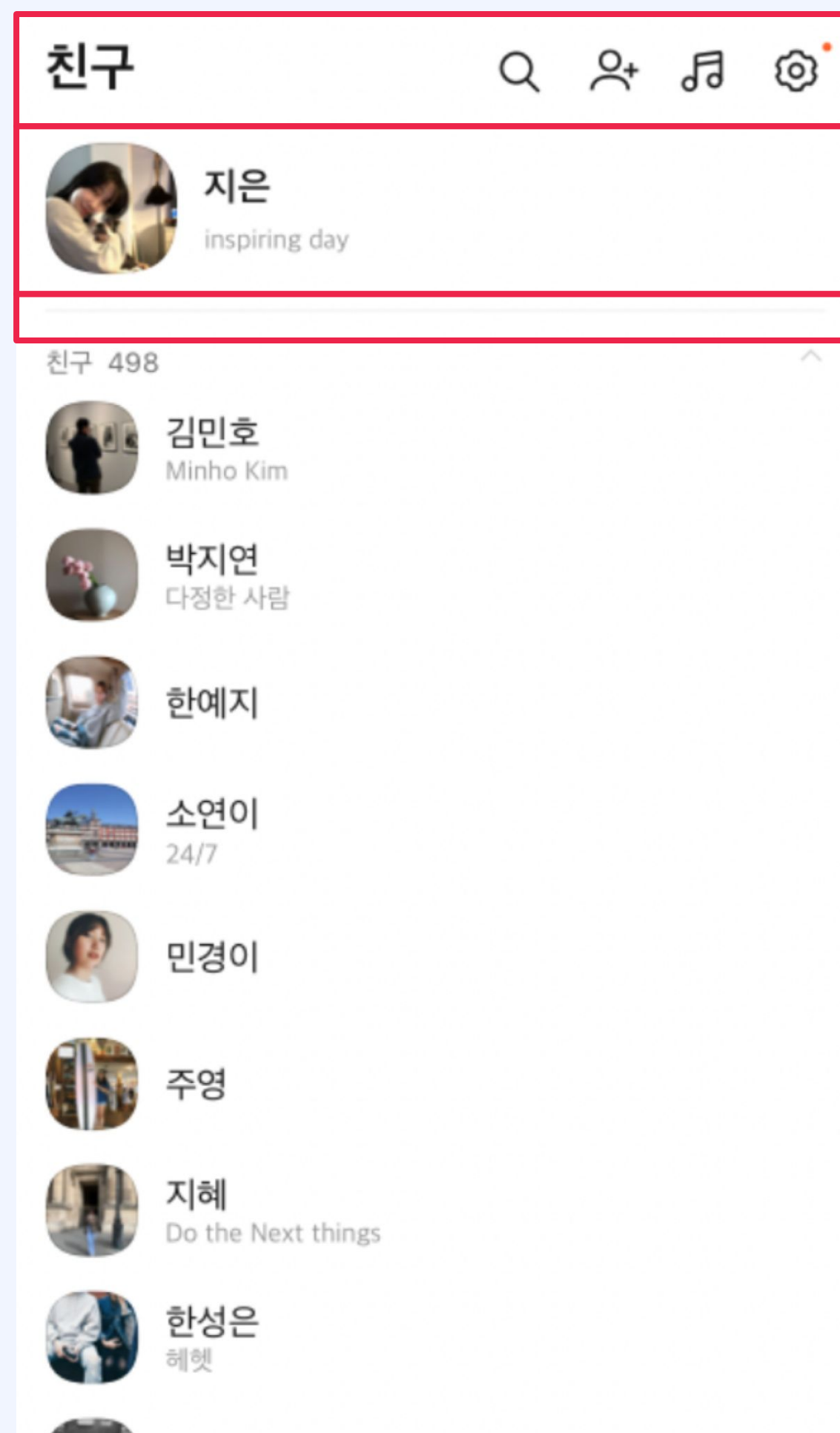
<Header />

<MyProfile />

화면 예시) 카카오톡 친구목록

2.

컴포넌트 및 prop
: 이론



<Header />

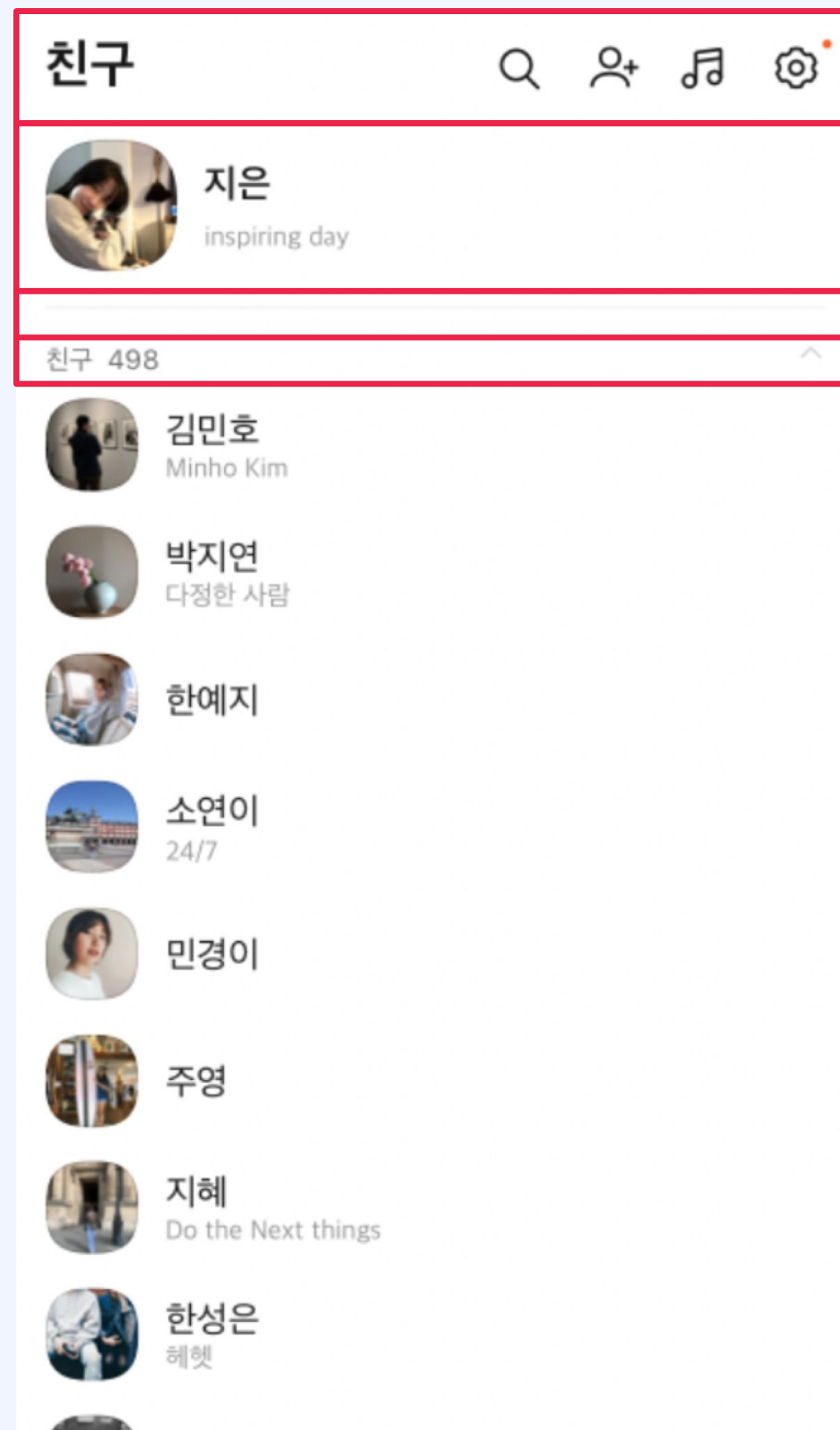
<MyProfile />

<Division />

화면 예시) 카카오톡 친구목록

2.

컴포넌트 및 prop
: 이론



<Header />

<MyProfile />

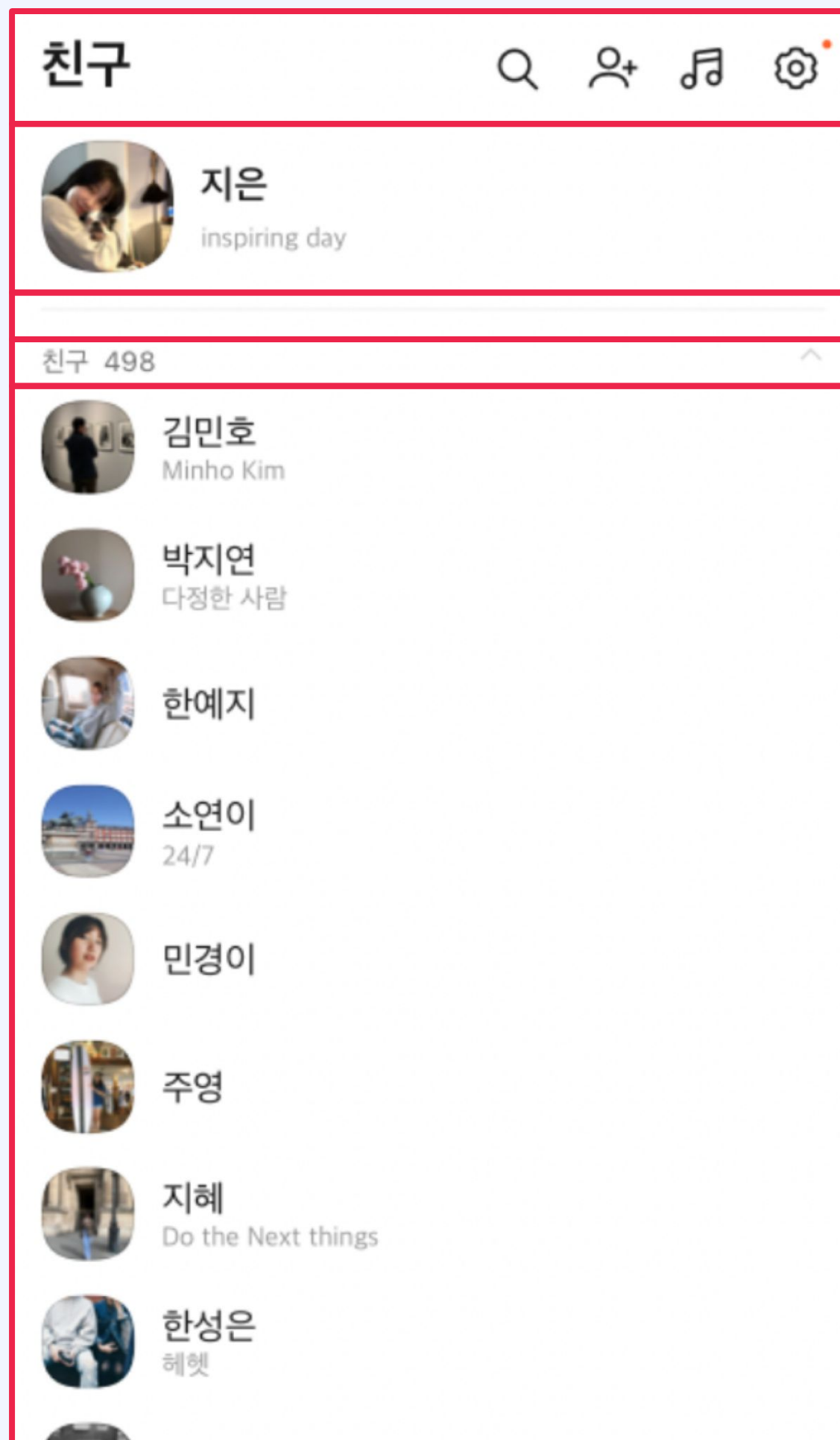
<Division />

<FriendSection />

화면 예시) 카카오톡 친구목록

2.

컴포넌트 및 prop
: 이론



<Header />

<MyProfile />

<Division />

<FriendSection />

<FriendList />

컴포넌트의 종류

2.

컴포넌트 및 prop
: 이론

- 클래스형 컴포넌트
- 함수형 컴포넌트

클래스형 컴포넌트

2.

컴포넌트 및 prop
: 이론

```

63  /**
64   * 클래스형 컴포넌트
65   */
    You, 1 second ago | 1 author (You)
66  class FriendList extends React.Component {
67      render() {
68          return (
69              <View>
70                  <Friend name="김민호" />
71                  <Friend name="박지연" />
72                  <Friend name="한예지" />
73                  <Friend name="소연이" />
74                  <Friend name="민경이" />
75                  <Friend name="주영" />
76                  <Friend name="지혜" />
77                  <Friend name="한성은" />
78              </View>
79          )
80      }
81  }
82  export default FriendList;
    
```

- class 키워드 필요
- Component로 상속 받아야 함
- render() 메소드 반드시 필요
- state, lifeCycle 관련 기능 사용 가능
- 함수형보다 메모리 자원을 더 사용

함수형 컴포넌트

2.

컴포넌트 및 prop
: 이론

```

4  /**
5   * 함수형 컴포넌트
6   */
7  const Friend = (props) => {
8    return <Text>{props.name}</Text>;
9  };
10
11 export default () => {
12   return (
13     <View>
14       <Friend name="김민호" />
15       <Friend name="박지연" />
16       <Friend name="한예지" />
17       <Friend name="소연이" />
18       <Friend name="민경이" />
19       <Friend name="주영" />
20       <Friend name="지혜" />
21       <Friend name="한성은" />
22     </View>
23   );
24 };

```

- state, lifeCycle 관련 기능 사용 불가능 -> hook으로 해결
- 클래스형보다 메모리 자원을 덜 사용
- 컴포넌트 선언이 편함
- 공식문서에서도 함수형 컴포넌트 + hook 사용을 권장

CH3. React Native 학습 시 주요 개념

3 컴포넌트 및 prop: 실습

CH3. React Native 학습 시 주요 개념

4 React Hooks (1):
useState 이론&실습

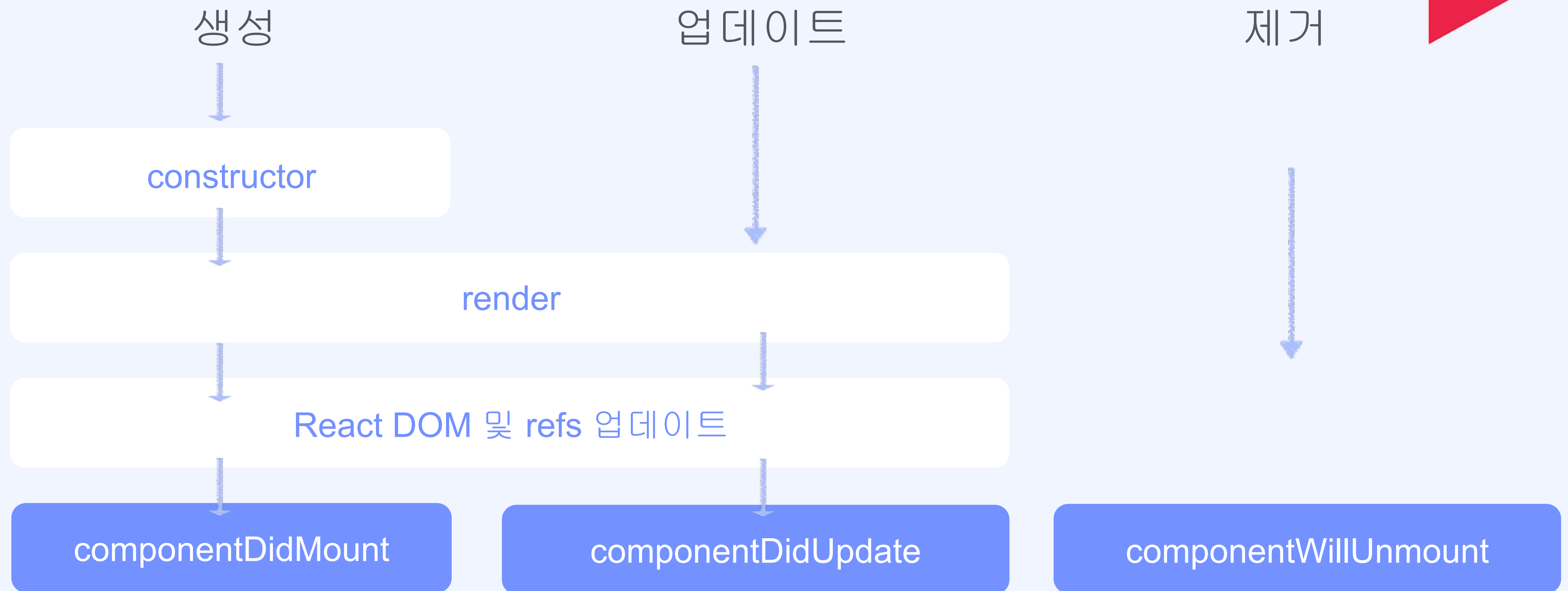
CH3. React Native 학습 시 주요 개념

5 React Hooks (2):
useEffect 이론&실습

클래스 컴포넌트의 생명주기

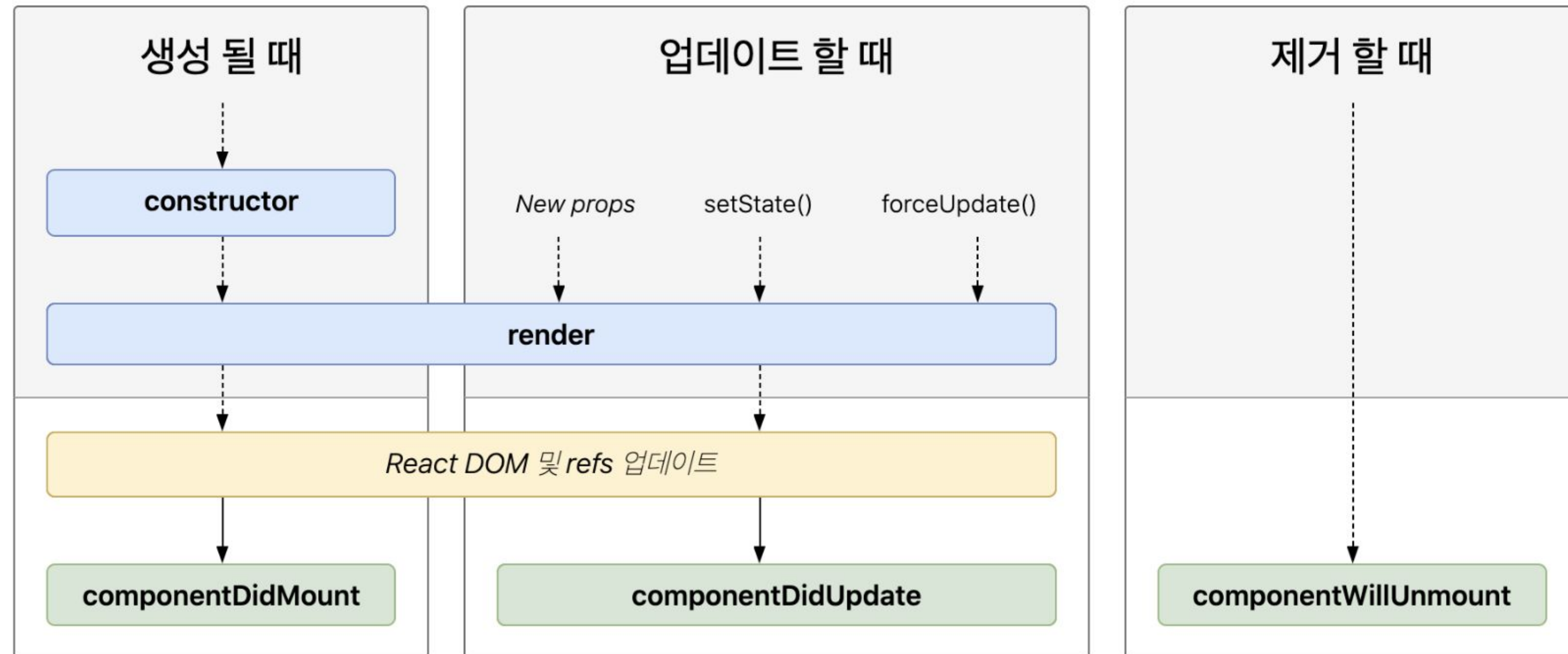
3.

맞춤 보고서 및
세그먼트 활용



“Render 단계”
순수하고 부작용이 없습니다. React에 의해 일시 중지, 중단 또는 재시작 될 수 있습니다

“Commit 단계”
DOM을 사용하여 부작용을 실행하고 업데이트를 예약 할 수 있습니다.



CH3. React Native 학습 시 주요 개념

6 React Hooks (3):
custom