

함수형 코딩 Intro

2023.06.11 민경민

함수형 프로그래밍

프로그래밍 패러다임 중 하나로,
프로그램을 수학적 함수의 연속으로 간주하고,

상태와 가변 데이터보다는
불변성(Immutability)과 **순수 함수(Pure Function)**를 강조하는 방식



Functional Programming

불변성과 순수함수

- 불변성(Immutability) : 한 번 정의된 변수는 변경할 수 없고, 변수에 할당된 값은 변하지 않는 것을 의미
- 순수 함수(Pure Function) : 순수 함수는 입력값에 대해 항상 동일한 출력값을 반환하며, 외부의 상태를 변경하지 않고, 부작용이 없는 함수

불변성 + 순수 함수 =>

코드의 예측 가능성과 안정성 UP =>

디버깅 테스트 용이

사이드 이펙트?

#1 : *No Side-Effects*



- **사이드 이펙트(Side Effect)** : 함수나 표현식의 실행 결과가 함수 외부의 상태를 변경하거나 다른 외부 동작을 수행하는 것

사이드 이팩트 없는 프로그램?

- 이메일 보내기
- 파일 작성
- DB write
- Console.log



순수함수로 프로그램을 작성한다는 멘탈 모델은 전혀 현실적이지 않음

현실적인 함수형 프로그래밍

- 액션 / 계산 / 데이터로 코드를 구분하는 패러다임
- 일급추상을 적극 활용하는 패러다임



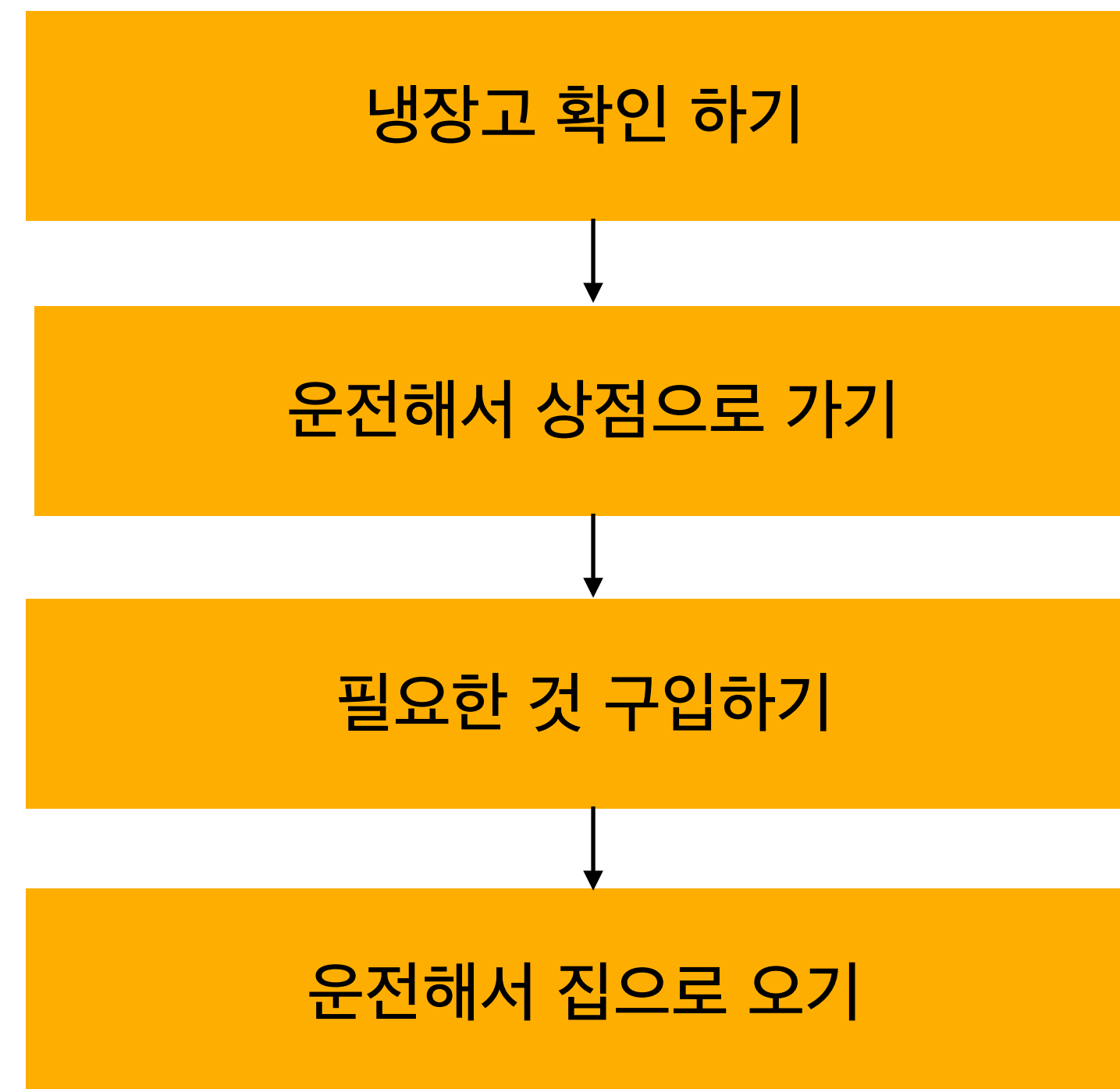
액션, 계산, 데이터 - 정의



- 액션 : 실행하는 시점, 순서에 의존하는 것
- 계산 : 입력값을 처리해서 출력값을 출력하는 것
- 데이터 : 이벤트에 대한 사실을 기록 한 것

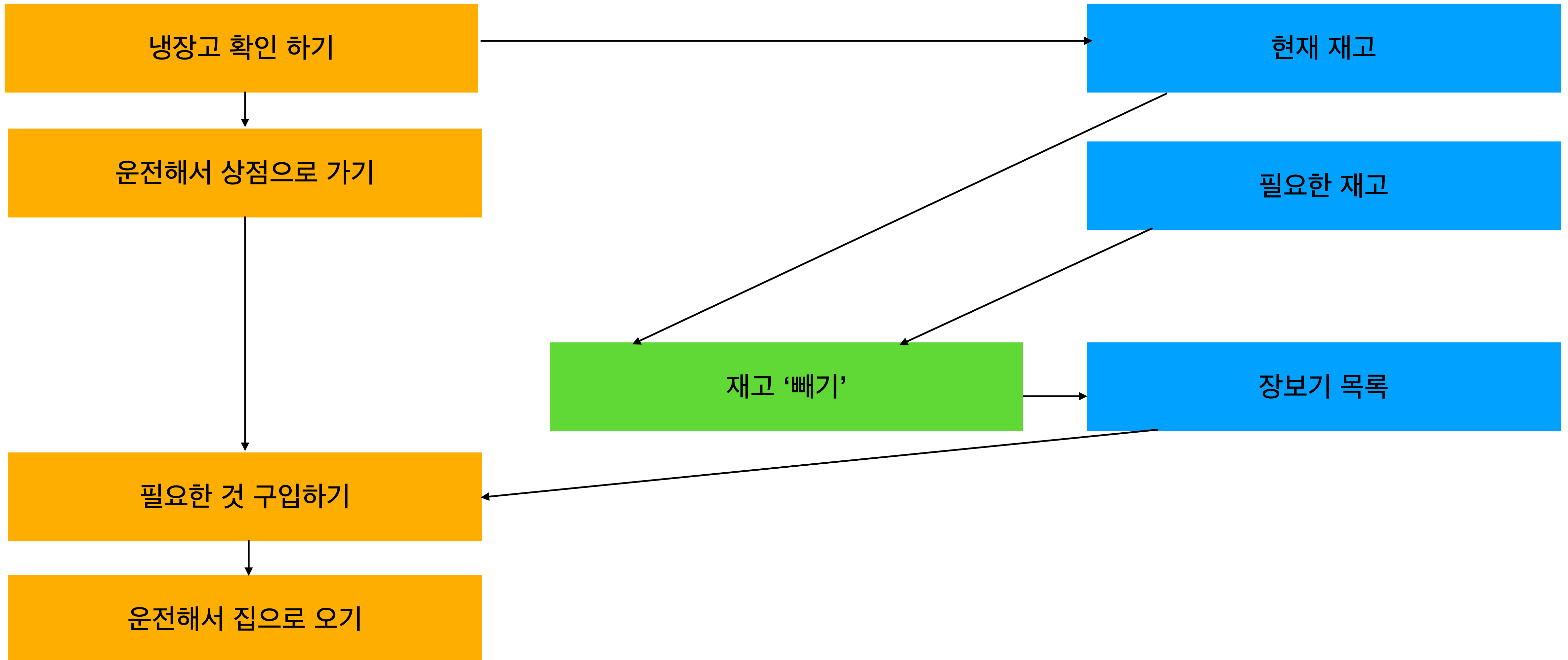
액션 계산 데이터는 어디에나 적용할 수 있다

장보기 과정



액션 계산 데이터는 어디에나 적용할 수 있다

장보기 과정



직관적으로 분류해보기

{ “firstname”: “Eric”, “lastname”: “Cartman”}

sendEmail(to, from ,subject, body)

sum(numbers)

saveUserDB(user)

string_length(str)

getCurrentTime()

[1, 10, 2, 45, 3, 98]

