

# LLM 활용 인공지능 서비스 개발자 양성과정

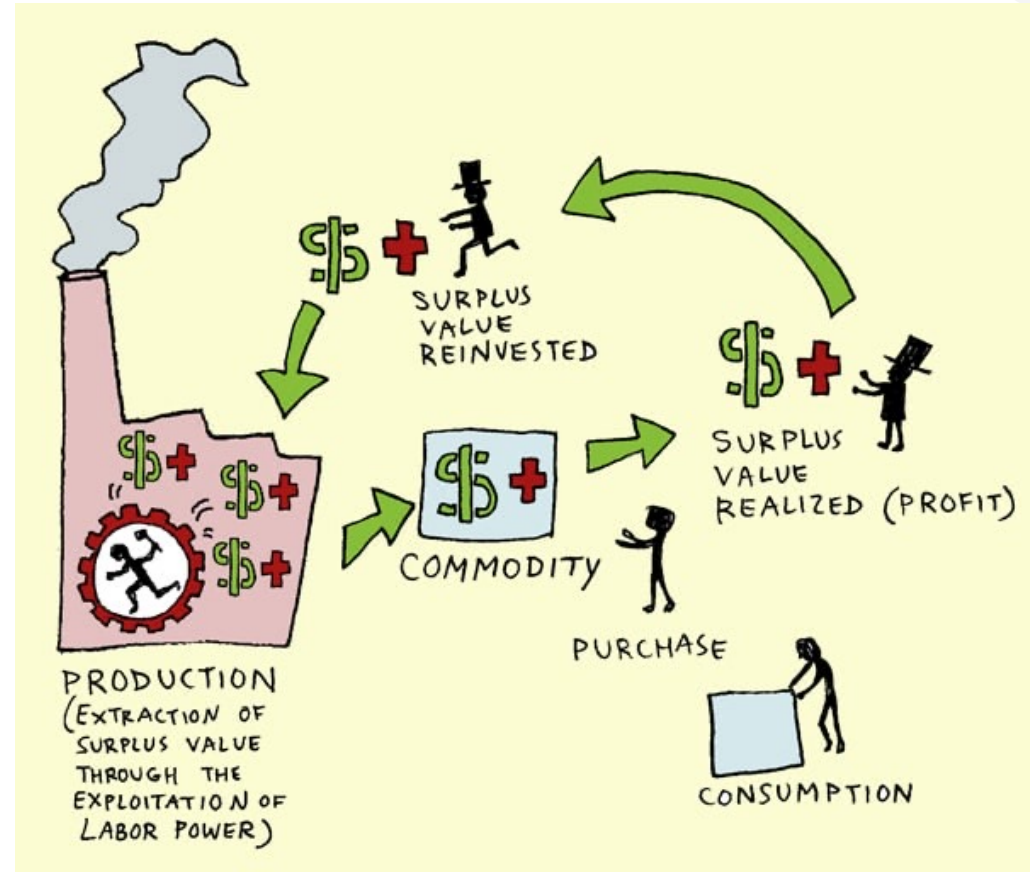
도봉 SeSAC 캠퍼스 X **Saltlux**

강사 최동혁

# 인공지능 개발자로 살아남기

### 문제는 속도

- 윤리문제
  - 코드를 바꾸거나 데이터를 바꾸도록 규제하면 된다.
- 일자리문제
  - 다른 일을 찾거나 일자리를 만들면 된다.
- 초과공급과 양극화 문제
  - 인공지능이 대체하는 것은 인간의 지식 노동
  - 초과공급을 소비가 받아주지 못할 정도로 파괴적인 속도
  - '풍요로운 종말'
  - 속도를 늦추거나(샘 알트먼 해고 사건) 소비를 늘리는 방법(ex. 양적 완화, 기본소득)



## 현실적인 조언들

1. 네이버 검색 금지(적어도 개발만큼은)
2. 개발 문서를 잘 읽자(되도록이면 영어로)
3. 카오스를 받아들이자
4. 개발일지를 쓰자
  - Notion 또는 obsidian 추천
5. 알고리즘이 나를 돕게 하자
  - 3b1b: <https://www.youtube.com/@3blue1brown>
  - Jack Herrington: <https://www.youtube.com/@jherr>
  - 코딩애플: <https://www.youtube.com/@codingapple>
  - 노마드코더: <https://www.youtube.com/@nomadcoders>
  - 골빈해커: <https://twitter.com/golbin>
  - 드림코딩: <https://www.youtube.com/@dream-coding>

# 개발문서 읽는 방법

## 1. Getting Started 정독

- 핵심 개념 숙지
- 그림으로 그려 놓은 것은 반드시 이해하자

## 2. 각 기능 또는 시나리오에 대한 설명 중 내가 만들어야 하는 부분을 먼저 정독

- 모든 기능을 정독할 필요 없음

## 3. 튜토리얼이 있다면 반드시 진행

## 4. 개발 시작

## 5. 개발 중간 필요한 내용 발췌독

# 개발문서 읽는 방법

## 6. Github은 유용한 정보로 가득

- Example 코드 있는지 체크할 것
- Stackoverflow에 없다면 Issues를 보자
- Wiki를 만들어 봤다면 살펴보자
- Star가 몇개 있는지 보자
- 마지막 업데이트가 언제 인지 보자 (너무 오래됐다면 그 라이브러리는 사용하지 말자)
- 필요하면 기여하자. 약간의 기여도 포트폴리오에 도움