aws re/start

컴퓨팅 기본 개념

학습 내용

강의의 핵심

학습 내용은 다음과 같습니다.

- 서버와 데이터 센터 설명
- 클라우드 컴퓨팅을 가능케 하는 컴퓨팅 기술에 대해 논의
- 소프트웨어의 개발 과정 설명

서버와 데이터 센터

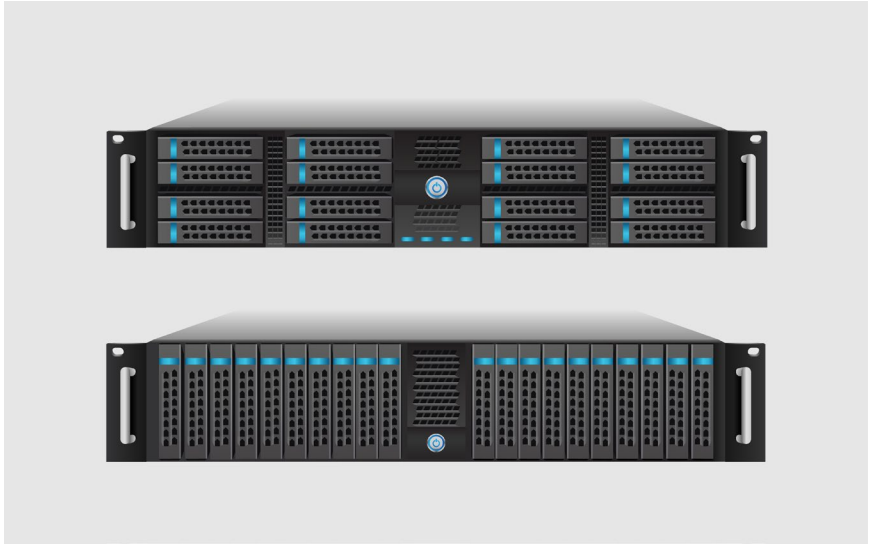
서버란?

다른 컴퓨터에 데이터 또는 서비스를 제공하는 하나의 컴퓨터

- 서버는 클라이언트 컴퓨터로부터 받은 요청에 대한 응답을 네트워크를 통해 전달합니다.

- 일반적으로 서버 하드웨어는 다음을 지원하기 때문에 데스크톱 하드웨어와는 차이가 있습니다.
  - 더 많은 메모리 및 다수의 CPU
  - 이중 전원 공급 장치 및 네트워크 인터페이스
  - 더 작은 폼 팩터

- 서버의 예
  - 웹 서버
  - 데이터베이스 서버
  - 메일 서버



클라이언트/서버의 예: 웹 애플리케이션

웹 서버에서 실행되며 데이터베이스 서버에 엑세스하는 웹 애플리케이션

서버는 어디에 상주하나요?

서버는 데이터 센터 내에 상주합니다.

데이터 센터는 다음을 포함한 한 조직의 컴퓨터 및 네트워크 장비 전체를 호스팅합니다.

- 서버
- 스토리지 장치
- 네트워크 장치(라우터, 스위치, 허브)
- 냉각 장비
- 무정전 전원 공급 장치(UPS)

데이터 센터는 누가 소유하나요?

기존 온프레미스 모델

- 사용자는 데이터 센터를 소유하며 자체 지정한 장소에서 호스팅
- 사용자는 자체 시설 내의 모든 하드웨어와 소프트웨어를 직접 구입, 설치, 구성 및 관리
- 사용자는 데이터 센터의 관리 및 유지 보수를 책임질 직원을 고용
- 사용자는 자체적 데이터 센터 리소스를 사용

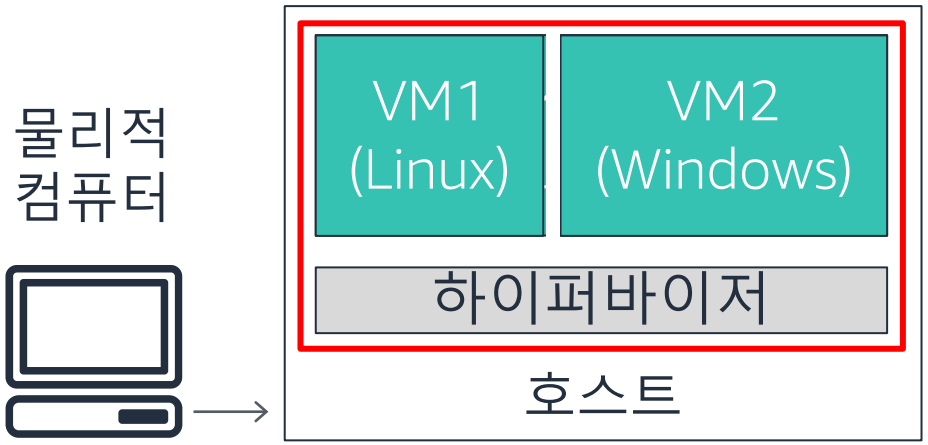
클라우드 모델

- 클라우드 서비스 공급자가 데이터 센터를 소유
- 클라우드 서비스 공급자가 자체 시설에 설치할 하드웨어와 인프라 소프트웨어를 구입
- 클라우드 서비스 공급자가 데이터 센터를 지원하기 위해 인원 고용
- 사용자가 클라우드 서비스 공급자의 데이터 센터 리소스를 사용하기 위해 비용을 지불

가상 머신

가상 머신이란?

가상 머신(VM)은 소프트웨어 기반의 컴퓨터



- VM은 **호스트**라고 하는 물리적인 컴퓨터 내에서 작동
- **하이퍼바이저**라고 하는 소프트웨어 계층이 물리적인 컴퓨터의 리소스(CPU, 메모리, 디스크, 네트워크)에 대한 액세스를 VM에 제공
- VM은 **자체 운영 체제(OS)**를 실행하며 하이퍼바이저를 통해 호스트와 상호작용
- 하나의 호스트에서 **다수의 VM**에 프로비저닝 가능

가상화를 사용하면 한 대의 물리적 기계 내에서 각각 자체적인 OS와 애플리케이션을 갖춘 VM을 다수 생성할 수 있습니다.

AWS의 이점

비용 절감

- 다수의 VM을 하나의 물리적인 장치에서 실행하면 새 컴퓨터를 구입할 필요가 줄어듭니다.

효율성

- 다수의 VM을 하나의 물리적인 컴퓨터에서 실행하면 활용도가 높아집니다.

재사용성 및 휴대성

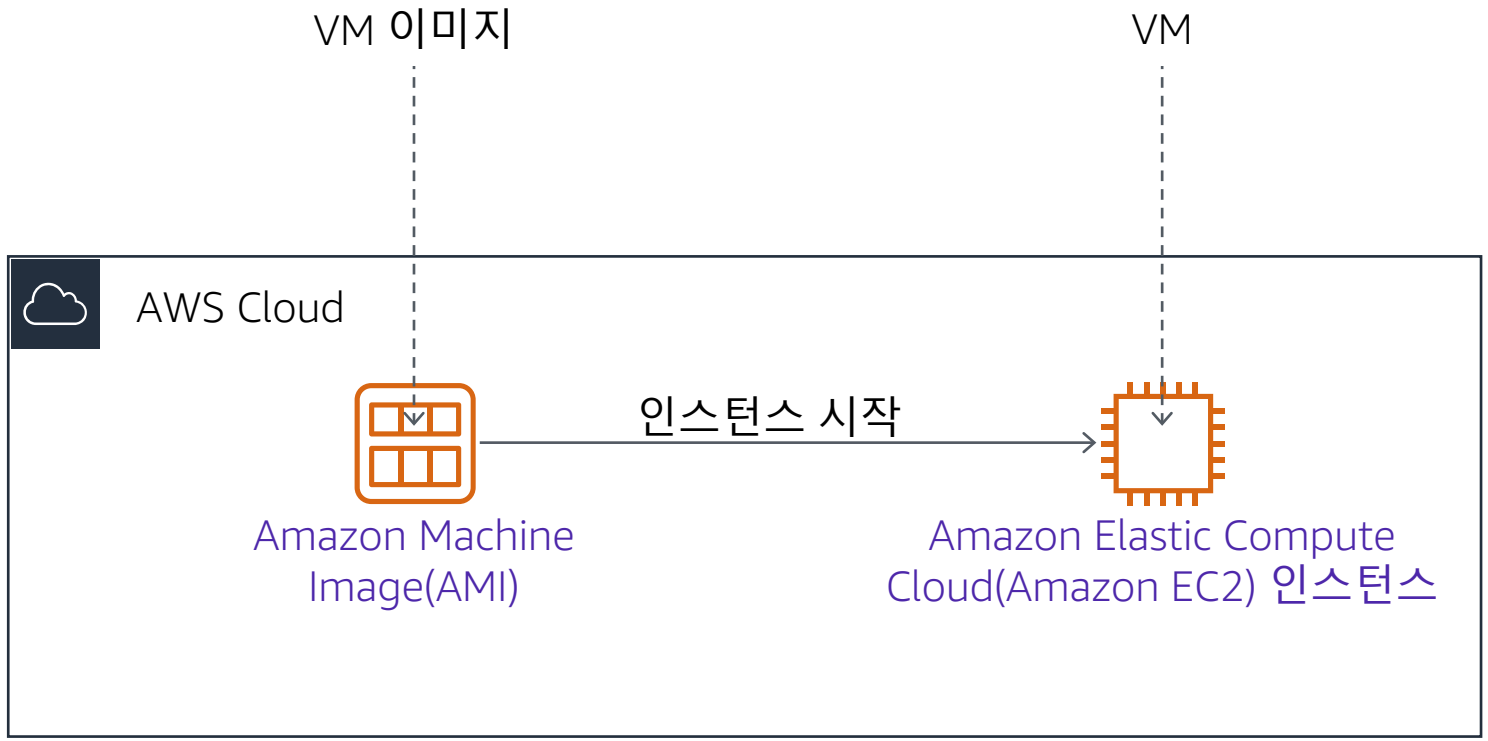
- 사용자는 같은 물리적 호스트에서 VM 이미지를 복사하거나 다른 호스트로 이동시켜 VM의 컴퓨팅 환경을 복제할 수 있습니다.

클라우드 내의 VM

VM은 클라우드 내 컴퓨팅의 기본 단위

VM은 다음을 가능하게 함

- 셀프 서비스
- 사용량에 따라 요금 지불
- 확장성

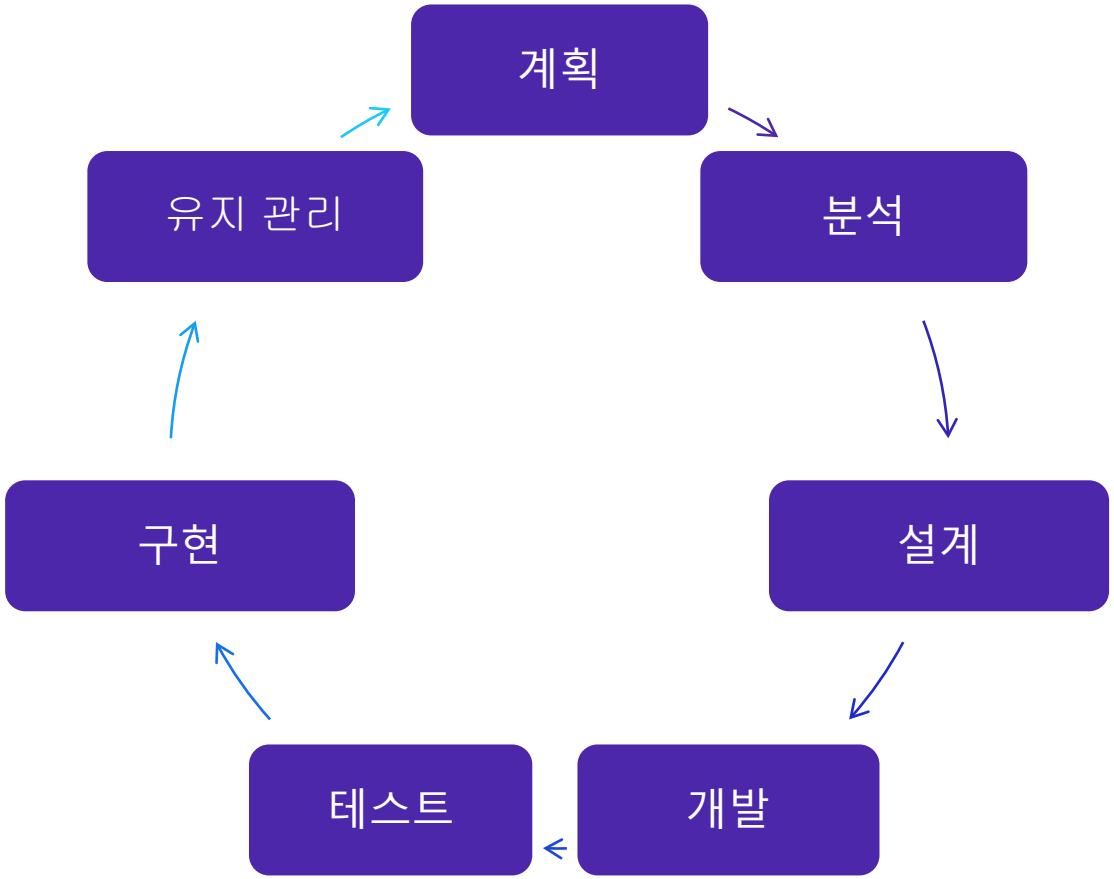


aws re/start

소프트웨어 개발 수명 주기

소프트웨어의 개발 과정

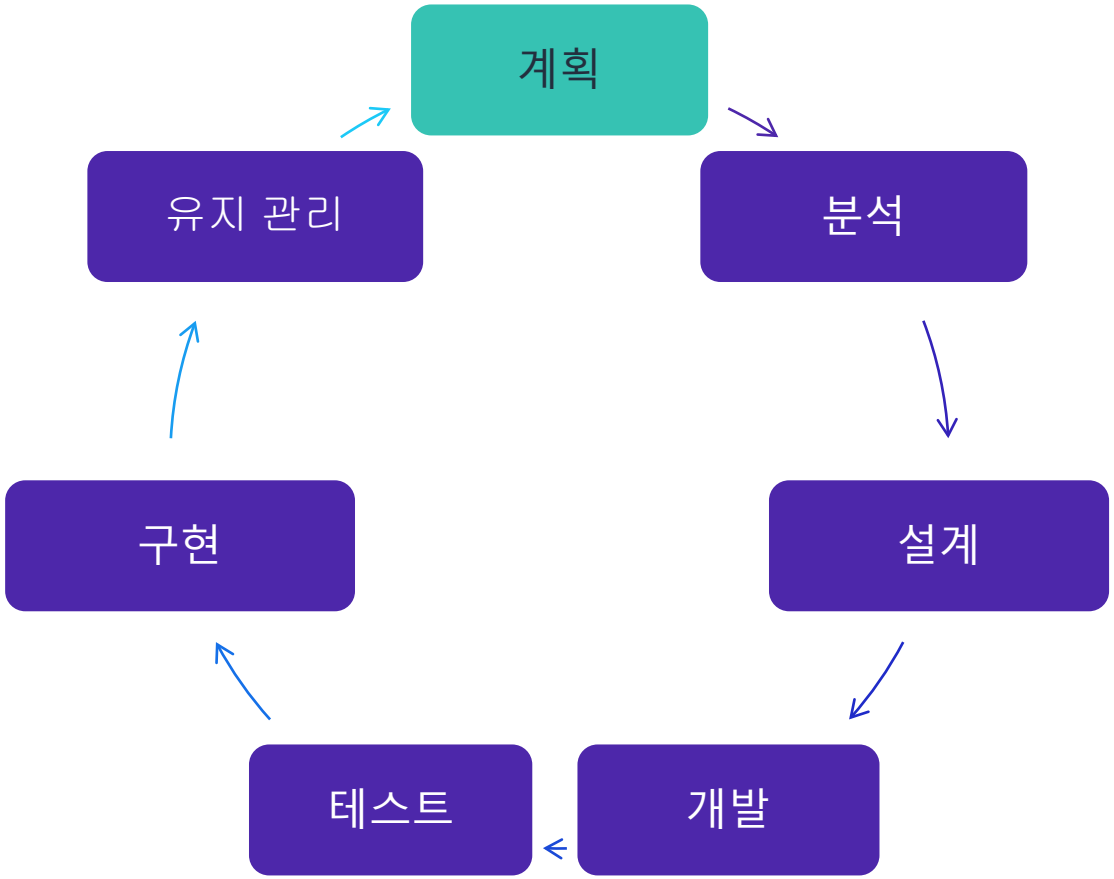
소프트웨어 개발 수명 주기(SDLC)



- 계획: 문제가 무엇이며 해결을 위해 어떤 리소스가 필요한가?
- 분석: 솔루션을 통해 얻기 원하는 것은 무엇인가?
- 설계: 원하는 것을 어떻게 구축할 것인가?
- 개발: 설계한 대로 구축.
- 테스트: 원하는 결과를 얻었는가?
- 구현: 구축한 결과물을 사용하기 시작.
- 유지 관리: 구축한 결과물을 개선.

계획

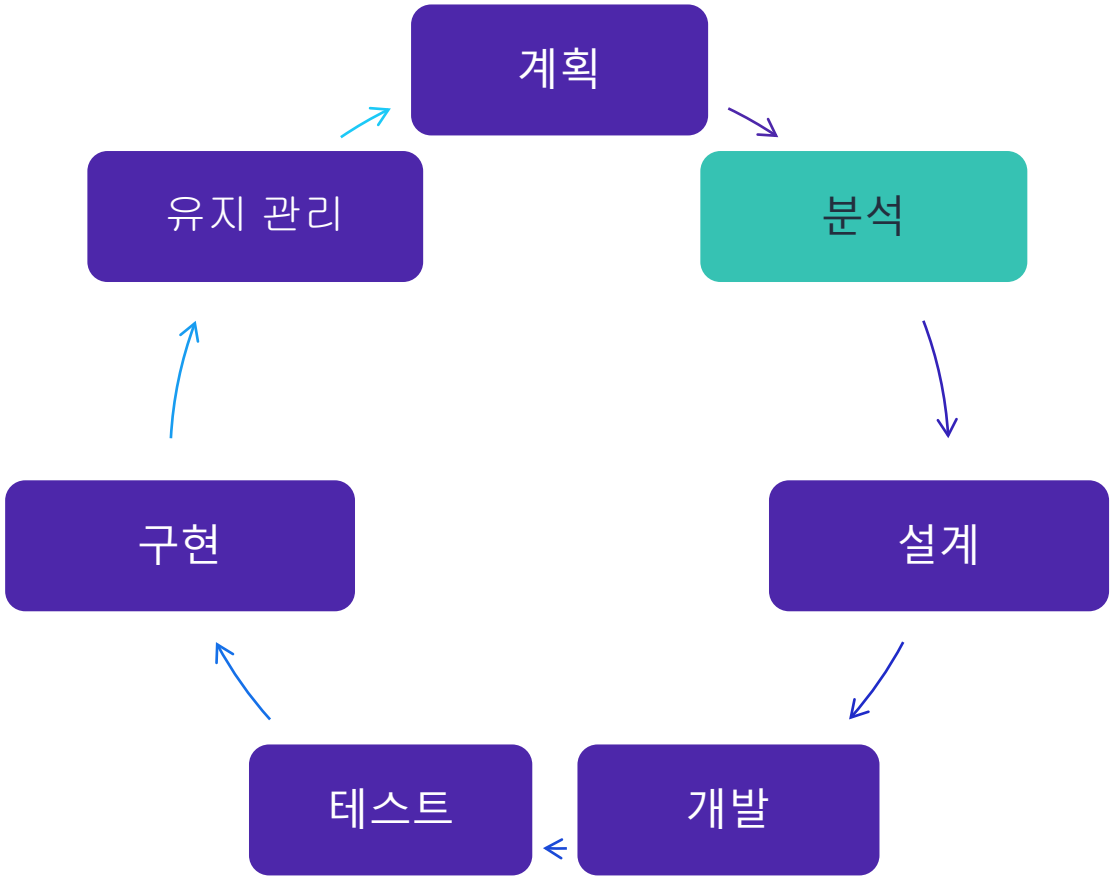
소프트웨어 개발 수명 주기(SDLC)



- 프로젝트의 목표를 수립하고 구현을 위해 필요한 리소스를 파악합니다. 이 단계의 결과물은 프로젝트 계획서입니다.
- 이 단계에서는 프로젝트 구현의 경제적, 운영적, 기술적 측면을 비롯하여 많은 요소를 고려합니다.
- 품질 보장을 위한 계획도 이 단계에서 수립합니다.

분석

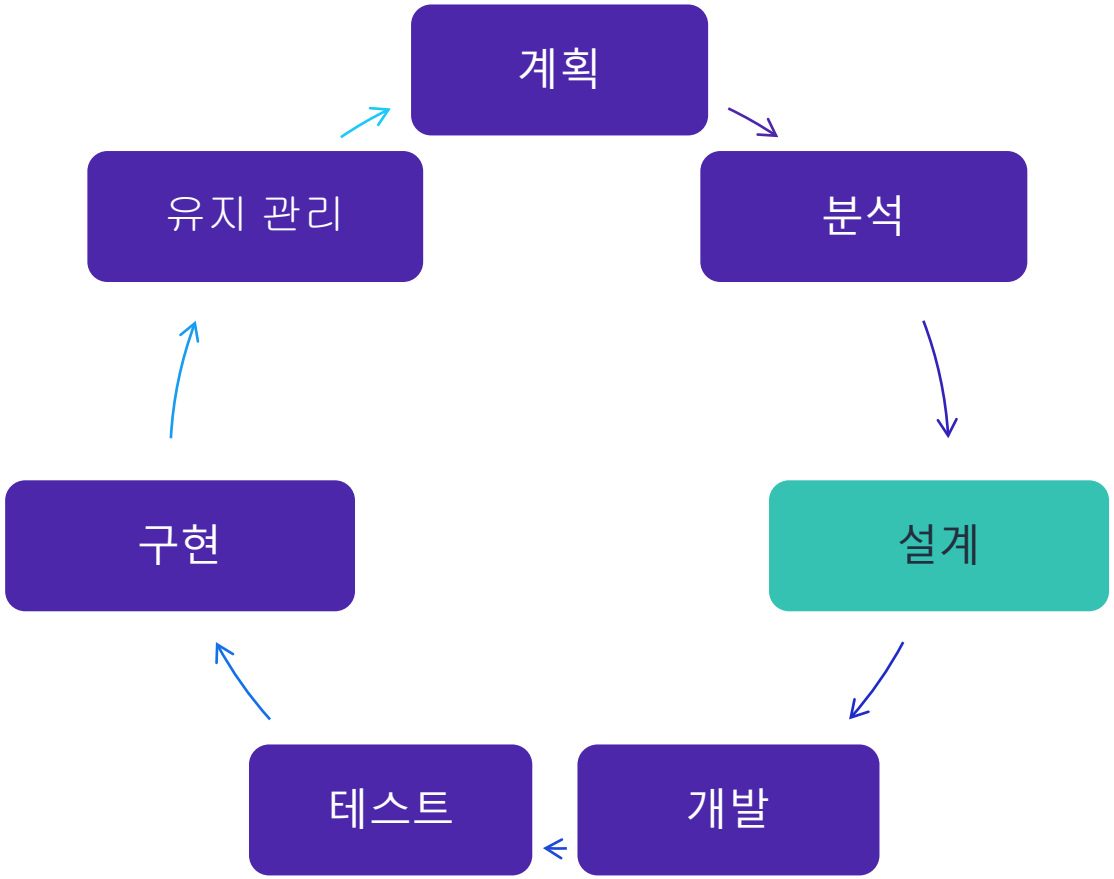
소프트웨어 개발 수명 주기(SDLC)



- 제품 요구 사항을 명확하게 정의해 소프트웨어 요구 사양서(SRS)로 문서화합니다.
- 다음으로 고객이 요구 사항을 승인합니다.
- SRS는 이어지는 SDLC의 각 단계에서 참조 도구로서 사용됩니다.

설계

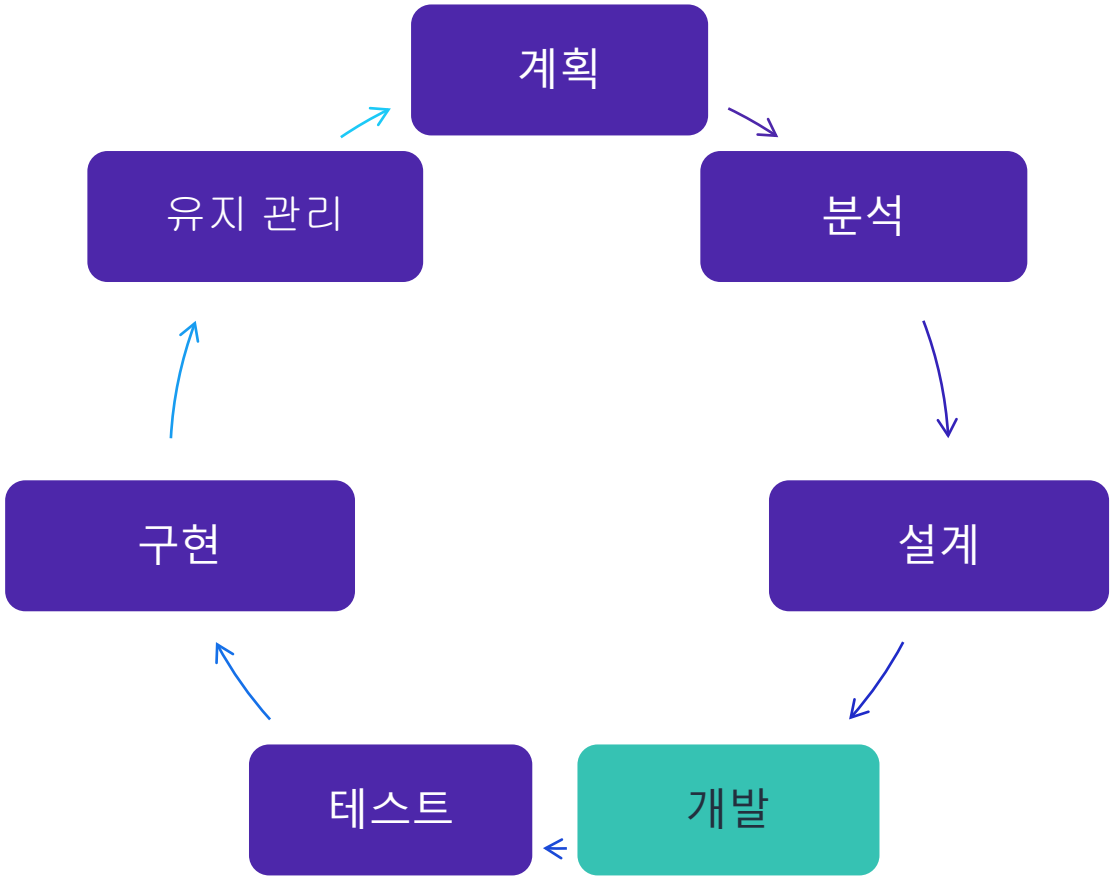
소프트웨어 개발 수명 주기(SDLC)



- SRS를 기반으로 여러 유형의 아키텍처를 평가하여 프로젝트에서 가장 효과적으로 사용될 수 있게 합니다.
- 설계 사양서를 통해 하나 이상의 설계 방법을 수립 및 제안합니다.
- 설계 사양서는 자세한 기능 설명과 사용자 인터페이스 설명과 같은 추가 정보를 포함합니다.
- 위험도, 예산, 시간 제약 등을 고려하며 설계 옵션을 검토합니다. 가장 적절한 설계 옵션을 선택합니다.

개발

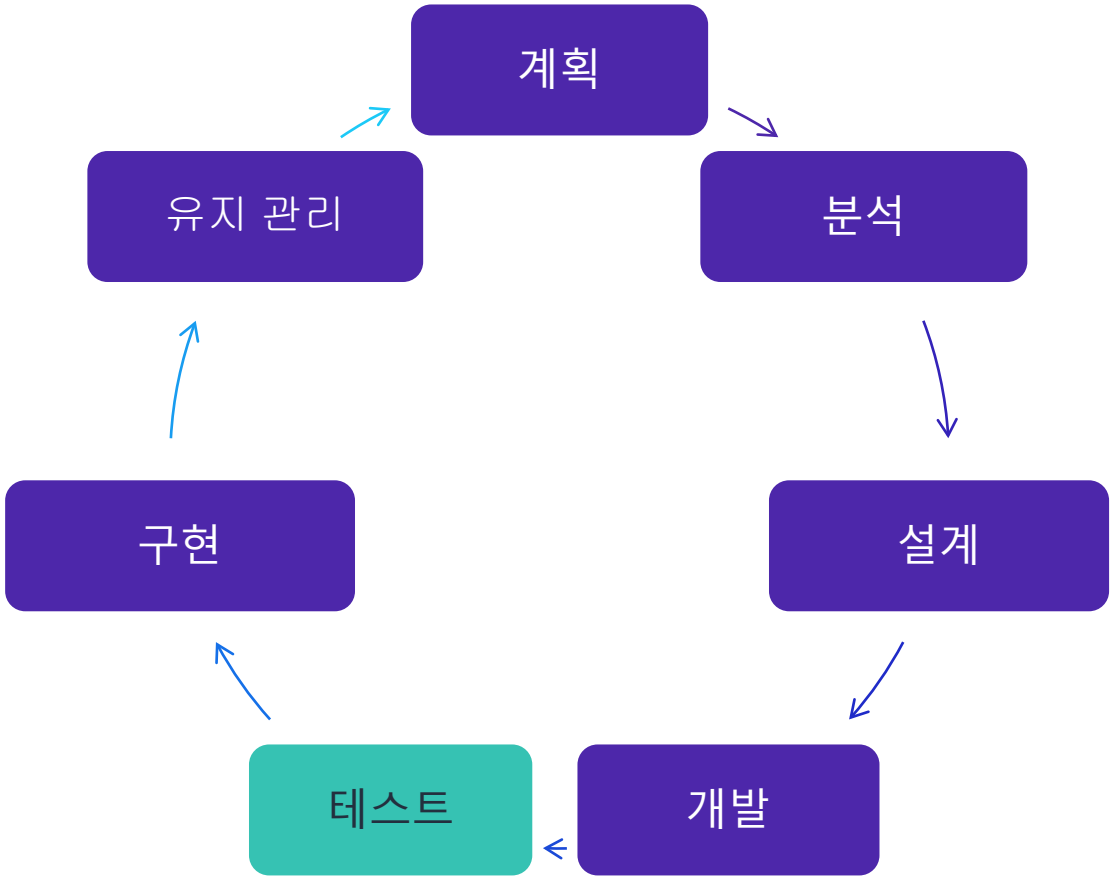
소프트웨어 개발 수명 주기(SDLC)



- 실제 컴퓨터 코드의 작성은 이 단계에서 수행하며 제품을 구축합니다.
- 설계 사양서와 조직의 소프트웨어 개발 표준 및 지침에 따라 코드를 작성합니다.
- 제작될 소프트웨어의 유형에 따라 프로그래밍 언어를 선택합니다.

테스트

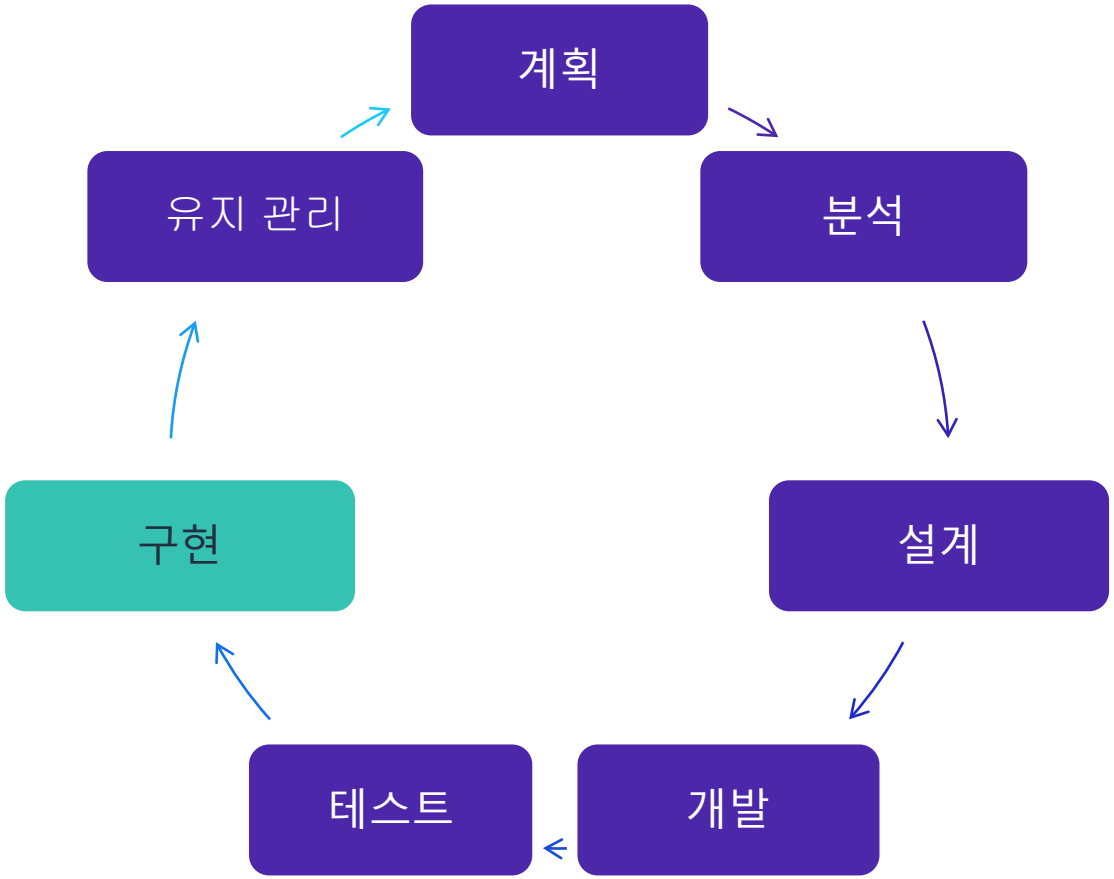
소프트웨어 개발 수명 주기(SDLC)



- SDLC에서 가장 중요한 단계 중 하나
- 다른 코드를 테스트하기 위한 코드 작성 가능 이 과정을 **자동화 테스팅**이라고 함
- 대표적인 테스팅 유형 -
  - 단위 테스트
  - 통합 테스트
  - 보안 테스트
  - 성능 테스트

구현

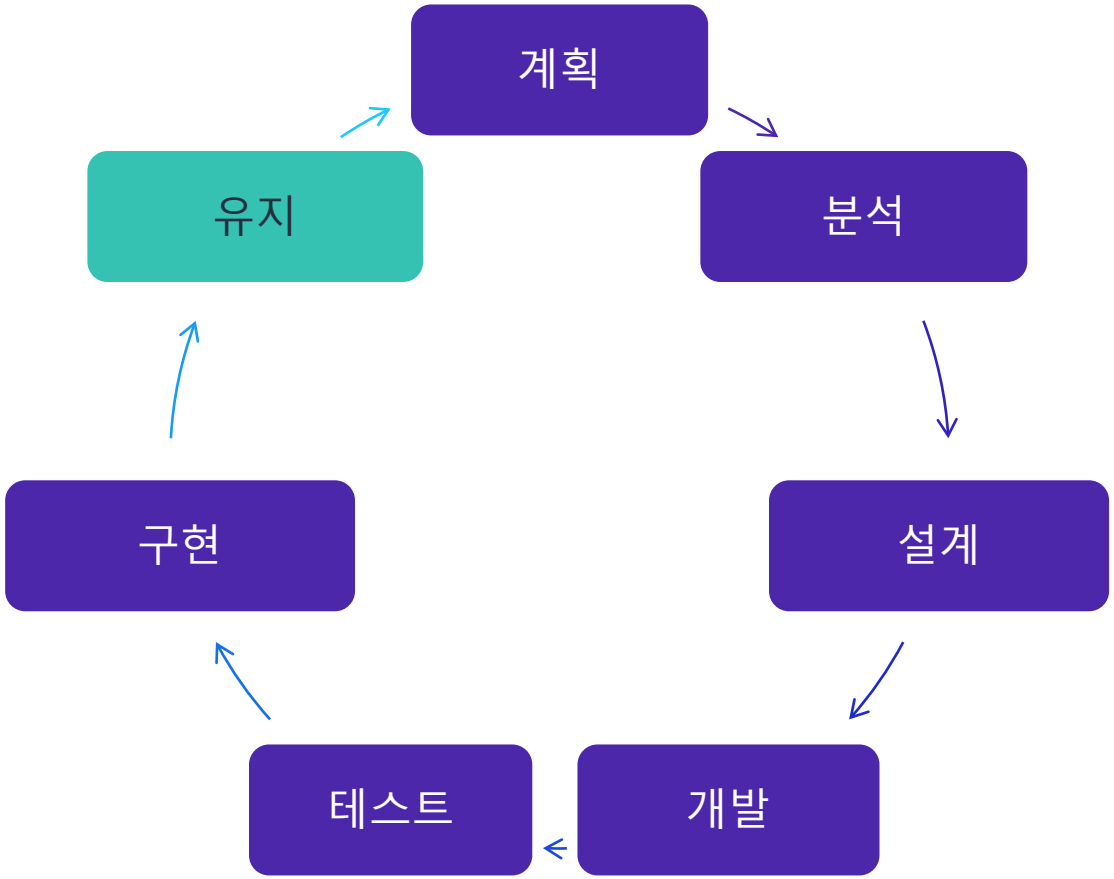
소프트웨어 개발 수명 주기(SDLC)



- 구현은 종종 배포라고도 합니다.
- 고객은 애플리케이션의 완성을 승인 및 확정합니다.
- 애플리케이션이 출시되고 생산에 사용됩니다.

유지 관리

소프트웨어 개발 수명 주기(SDLC)



- 생산 과정에서 애플리케이션을 지속적으로 모니터링해 올바르게 동작하는지 확인해야 합니다.
- 다음의 여러 다른 이유로 유지 관리가 필요합니다.
  - 결함 또는 오류 파악 → 교정적 유지 관리
  - 애플리케이션 환경의 변화 → 적응적 유지 관리
  - 애플리케이션 요구 사항의 변경 → 완벽 유지 관리
  - 오류 발생 방지 → 예방적 유지 관리

핵심 요점

- 서버란 다른 컴퓨터에 데이터 또는 서비스를 제공하는 하나의 컴퓨터를 말합니다.
- 데이터 센터란 하나의 조직이 컴퓨터와 네트워킹 장비를 보관 및 운영하는 물리적 위치를 말합니다.
- 하드웨어 가상화는 물리적 컴퓨터에서 VM을 생성하는 것을 가능케 합니다. 이는 클라우드 컴퓨팅에서 사용되는 기본적인 기술입니다.
- 소프트웨어 개발 수명 주기의 단계 –
  - 계획
  - 분석
  - 설계
  - 개발
  - 테스트
  - 구현
  - 유지 관리

aws re/start