1. 파트 1: Amazon Elastic Block Store (Amazon EBS)
2. 파트 2: Amazon Simple Storage Service (Amazon S3)
   1. Amazon S3
      1. 아마존형 웹하드, 모든 서비스가 S3와 연동가능
      2. 객체 수준 스토리지
      3. 원하는 만큼 많은 수의 객체 저장 가능
      4. 선택한 리전 내에 여러 AWS 시설에 중복 저장
      5. AWS Management Console, AWS SDK 혹은 타사 솔루션을 통해 엑세스
      6. 완전 관리형이기 때문에, 워크로드나 가용성을 신경쓰지 않아도 됨
      7. 객체 업로드나 삭제는 알림이나 워크플로 혹은 스크립트를 트리거 가능 (Lambda와 연동 사용 가능)
      8. 전송 데이터와 저장데이터는 자동으로 암호화 됨
   2. 데이터 업로드
      1. AWS 리전 중 한 곳에 버킷을 생성하고
      2. 원하는 만큼 객체를 버킷에 업로드
      3. 버킷 주소: 리전 코드.amazonaws.com/버킷이름
      4. 객체 주소: 리전 코드.amazonaws.com/버킷이름/키(객체명)
      5. 버킷 단위로 저장 / 한 리전에만 저장 / 버킷 이름 중복 불가
   3. Amazon S3 스토리지 계층
      1. Amazon S3 Standard
         1. 저장은 비교적 비싸나, 읽고 쓰는 비용이 낮음
         2. 접근을 자주하는 경우에 적합
      2. Amazon S3 Standard – IA
         1. 엑세스 빈도가 낮지만, 필요할 때 빠르게 엑세스해야 하는 경우
      3. Amazon S3 One Zone – IA
         1. Amazon S3 Standard - IA에서 단일 가용 영역(AZ)에 저장한 다는 점이 다름.
      4. Amazon Glacier
   4. Amazon S3: 스토리지 요금 (요금이 부과되는 항목)
      1. 요금 = ∑(얼마나 저장? + 읽고 쓰기 + 리전 밖 이동)
      2. 스토리지 클래스 유형
         1. Standard Storage: 이 경우가 더 비싸다
         2. Standard-Infrequent Access (SIA)
      3. 스토리지 용량
         1. 객체의 숫자와 사이즈
         2. 스토리지 유형
      4. 요청
         1. 요청(GET,PUT,COPY)의 숫자
         2. GET은 다른 요금율을 가지고 있음
      5. 데이터 전송
         1. 리전 외부로 전송되는 데이터의 양에 따라 요금부과
         2. 가용영역은 하나의 리전으로 연결되기에 내부로 간주함
   5. 사용 사례
      1. 응용 프로그램의 자산(이미지등)을 저장
      2. 정적 웹 호스팅
      3. 백업(ex) EC2의 스냅샷) 및 재해복구
3. 파트 3: Amazon Elastic File System (Amazon EFS)
4. 파트 4: Amazon Glacier