

ICT의 가치를 이끄는 사람들!!

126회

## 정보관리기술사 기출풀이 교시

## 국가기술자격 기술사 시험문제

정보처리기술사 제 126 회

제 3 교시

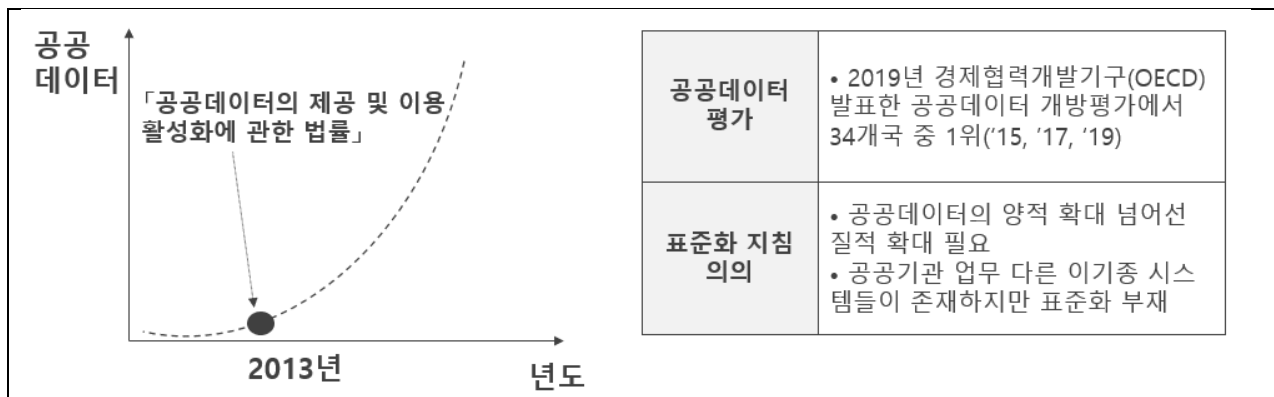
분야	정보처리	종목	정보관리기술사 컴퓨터시스템응용기술사	수험 번호		성 명	
----	------	----	------------------------	----------	--	--------	--

※ 다음 문제 중 4 문제를 선택하여 설명하십시오. (각 25 점)

1. 공공기관의 데이터베이스 표준화지침에서 정의하는 테이블 정의서에 기록될 항목과 항목의 작성 지침에 대해서 설명하십시오.
2. 전자정부 웹사이트 UI/UX 설계기준 7 가지와 웹접근성, 웹호환성, 웹개방성, 웹최적화에 대해 각각 설명하십시오.
3. 최근 SW 가 대부분 산업에 적용되면서 소프트웨어 안전이 중요하게 여겨지고 있다. 소프트웨어 안전진단가이드(공통분야)에 따른 소프트웨어 안전성 진단 항목 중 기능 동작 정확성 진단의 절차 및 주요활동 내용을 설명하십시오.
4. 앙상블(Ensemble) 모형의 Bagging 과 Boosting 알고리즘을 설명하십시오.
5. 소프트웨어 구조의 평가 및 개선을 위한 소프트웨어 아키텍처 분석 방법에 대하여 다음 내용을 설명하십시오.
  - 1) 소프트웨어 아키텍처 분석의 필요성
  - 2) 소프트웨어 아키텍처 정방향 분석과 역방향 분석 개념
  - 3) 소프트웨어 아키텍처 평가 기법중 ATAM(Architecture Trade-off Analysis Method)
6. 디지털 뉴딜 ‘데이터 댐’의 핵심인 인공지능 학습용 빅데이터 구축 사업의 성공적 추진을 위한 정책과 데이터 품질에 대한 관심도가 높아지고 있다. 다음의 내용을 설명하십시오.
  - 1) 인공지능 학습용 데이터의 특성
  - 2) 데이터 획득, 정제 방법과 기준
  - 3) 데이터 라벨링(Data Labeling) 및 어노테이션(Annotation) 방식

문 제	1. 공공기관의 데이터베이스 표준화지침에서 정의하는 테이블 정의서에 기록될 항목과 항목의 작성 지침에 대해서 설명하시오.		
출 제 영 역	데이터베이스	난 이 도	★★★★☆
출 제 배 경	- 인공지능, 빅데이터, IoT 등 지능정보기술 발전으로 데이터 거래,유통,융합 수요가 증가하면서 공공데이터 개방의 양적인 확대에서 질적인 측면 개선을 위해 지침 개정에 따른 출제		
출 제 빈 도	미출제		
참 고 자 료	- 공공데이터베이스 표준화 관리 매뉴얼(NIA, 2021.06)		
Key word	- 물리 DB 명, 소유자, 한/영문명, 유형, 엔티티명, 설명, 업무분류체계, 테이블 볼륨, 발생주기, 공개/비공개 사유		
풀 이	조승우(124 회 정보관리기술사/alectos@naver.com)		

## 1. 공공기관의 데이터베이스 표준화지침의 배경



- 「공공기관의 데이터베이스 표준화 지침」은 빅데이터 시대에 공공기관들이 보다 효율적으로 데이터 표준화 활동을 수행할 수 있도록 지원을 위해 제작함
- 공공기관에서 생성, 관리하고 있는 공공데이터 뿐만 아니라 국가차원에서 관리하고 있는 공공데이터를 포함하여 데이터 표준화를 이루기 위하여 가이드 제작함

## 2. 공공기관의 데이터 베이스 표준화 지침의 테이블 정의서 개요

### 가. 공공기관 데이터 베이스 표준화 지침의 테이블 정의서 개념 설명

개념	설 명
협의를 개념	- 공공기관의 장이 업무 담당자로 하여금 공공데이터베이스 설계, 구축, 운영 등 관리 시 반영토록 한 지침에 해당하는 테이블 정의 표준 양식
광의의 개념	- 공공기관이 표준관련 제도를 쉽게 적용하고 관리, 운영상의 편의성을 제고하여 데이터 융합 및 민간데이터 이용을 활성화 하도록 한 지침의 정의서

- 테이블 정의서는 「공공기관의 데이터베이스 표준화 지침」 별지 제 10 호 서식을 따라 가이드 제공함
- 테이블 정의서 적용 대상은 「공공데이터법」 제 2 조에 따라 공공기관이 법령 등에서 정하는 목적을 위하여 생성 또는 취득하여 관리하고 있는 모든 DB 가 표준화 대상으로, 공공기관에서 개별적으로 구축한 신규 및 고도화 시스템의 DB 가 공공데이터 표준화 대상됨

## 나. 공공기관 데이터베이스 표준화 지침의 테이블 정의서 항목

## 테이블정의서

물리DB명	테이블소유자	테이블 영문명	테이블 한글명	테이블 유형	관련엔티티명
테이블설명	업무분류체계	보존 기간	테이블 볼륨	발생주기	공개/비공개 사유

- 테이블 정의서에는 데이터베이스 명, 테이블 유형, 테이블 한글/영문명, 유형, 엔티티명, 업무분류체계, 보존기간, 볼륨, 발생주기 등 테이블 설계 시에 필요한 모든 정의항목을 포함하도록 함.
- 테이블 정의서는 「공공기관의 데이터베이스 표준화 지침」의 제 9 조 공공데이터베이스 산출물의 작성 및 관리의 표준산출물로 작성하도록 함

## 3. 공공기관의 데이터 베이스 표준화 지침의 테이블 정의서 항목 작성 지침

작성분류	항목	항목 정의 및 작성 지침
테이블 정의	- 물리 DB 명	- 정보시스템에서 DB 를 식별하기 위하여 사용하는 물리 정보명(영문명)을 기재
	- 테이블 소유자	- 테이블 스키마에 대한 소유권(오너쉽)을 갖는 데이터베이스 사용자를 기재
테이블 명칭	- 테이블 영문명	- 물리적인 DB 생성할 테이블 이름을 기재 - '표준용어정의서'를 참조하여 테이블 명명규칙을 정의하고, 해당 명명규칙에 따라 테이블명을 정의 - 1 개의 엔티티가 다수의 테이블로 분리되어 구축되는 경우 일련번호를 포함 가능
	- 테이블 한글명	- 논리 DB 에 부여할 테이블 한글 명칭을 기재 - '표준용어정의서'를 참조하여 테이블 명명규칙을 정의하고, 해당 명명규칙에 따라 테이블명을 정의
테이블 유형	- 테이블 유형	- 테이블의 논리적, 물리적, 특성에 따른 테이블 유형을 기재 · 물리적 특성으로 테이블 유형을 정의하여 기재하는 경우: 일반 테이블, 파티션 테이블, 클러스터 테이블, 뷰 테이블 등으로 구분하여 기재 · 논리적 특성으로 테이블 유형을 정의하여 기재하는 경우: 코드테이블, 마스터테이블, 임시테이블, 통계테이블 등으로 구분하여 기재 ※ 테이블명에 테이블 유형이 반영된 경우 생략 가능
	- 관련 엔티티명	- 해당 테이블이 어떤 엔티티로부터 물리적으로 구현된 테이블인지, 엔티티정의서에 기록한 엔티티명(한글)을 기재 - 여러 엔티티가 한의 테이블로 통합되어 구현되는 경우 관련 엔티티들을 모두 기재
테이블	- 테이블 설명	- 테이블 물리적 특징에 대해 기재

관리설명		<ul style="list-style-type: none"> <li>· 파티션 테이블의 경우 분할 조건이나 관련 특징을 기재</li> <li>· 뷰 테이블의 경우 뷰를 구성하는 집합 결정 조건을 기재</li> </ul>
	- 업무분류 체계	- 정부기능분류체계(BRM) 또는 공공기관별 자체 BRM 을 참조하여 하위 분류레벨(4~6 단계)까지 기재
	- 보존기간	<ul style="list-style-type: none"> <li>- 해당 테이블의 데이터를 발생시점 기준으로 얼마나 오랫동안 보존하는지를 기재</li> <li>· 월 단위 또는 년 단위로 기재</li> </ul>
	- 테이블 볼륨	- 테이블의 보존기간으로 정의한 기간 동안 현재 테이블이 보유하고 있는 저장량을 저장 건수로 기재
	- 발생주기	<ul style="list-style-type: none"> <li>- 해당 테이블의 데이터가 작성되거나 수정되는 주기를 기재</li> <li>- 발생주기의 종류</li> <li>· 실시간: 수시로 데이터가 발생하는 온라인성 테이블</li> <li>· 일/주/월: 하루/일주일/한달에 한 번 발생</li> <li>· 분기/반기: 3 개월/6 개월에 한 번 발생</li> <li>· 년: 1 년에 한번 발생</li> </ul>
데이터개방 여부	- 공개/비공개 여부	<ul style="list-style-type: none"> <li>- 테이블이 보유한 원천데이터에 대한 공개, 부분공개 또는 비공개 여부를 기재</li> <li>· 부분 공개 또는 비공개인 경우 해당 사유를 기재</li> </ul>

- 공공기관 데이터베이스의 효율적인 메타데이터 수집·관리 및 표준화 활동등을 위해 기관별 메타시스템을 구축운영이 활발할 수 있도록 관련 기준 제시함

#### 4. 공공기관의 데이터 베이스 표준화 지침의 메타데이터 수집환경도



- 공공기관은 메타데이터의 등록, 운영과 관련하여 행정안전부와 공공데이터 활용지원센터가 구축한 기관 메타데이터 관리시스템(공동활용형 포함)을 도입하거나, 개별 메타데이터 관리시스템 구축 또는 상위기관의 기관 메타데이터 관리시스템을 사용하도록 가이드 함.

“끝”

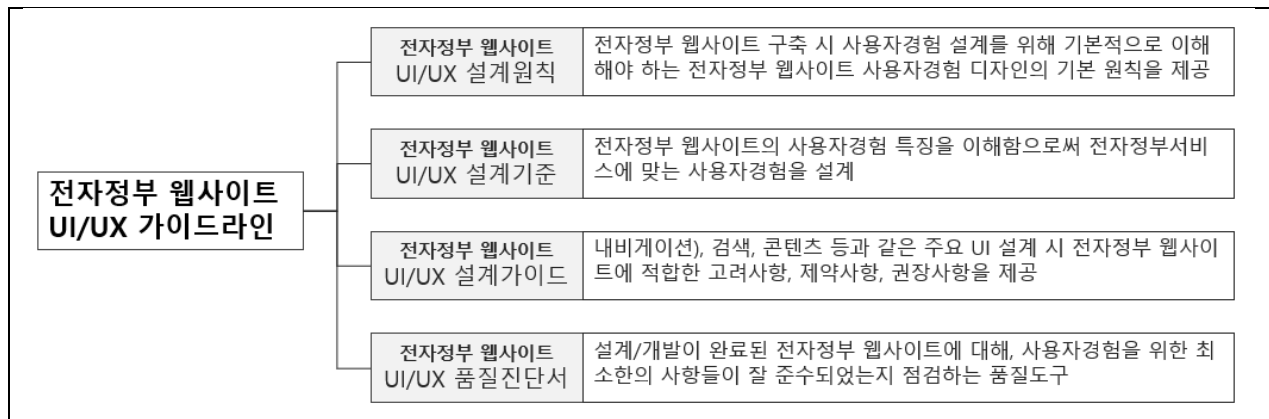
**기출풀이 의견**

1. 기본적인 테이블정의서 내용을 바탕으로 공공기관에서 중요데이터 평가지표인 공공데이터 오픈과 관련된 사항을 함께 기재를 한다면 해당 지침 가이드에 대한 설명을 수월하게 작성할 수 있었을 것입니다.



문 제	2. 전자정부 웹사이트 UI/UX 설계기준 7 가지와 웹접근성, 웹호환성, 웹개방성, 웹최적화에 대해 각각 설명하시오.		
출 제 영 역	디지털서비스	난 이 도	★★★★☆
출 제 배 경	- 전자정부 웹사이트 가이드에 대해서 알고 있는지 여부확인 - 사회적 이슈인 사회약자 보호, 취약계층 보호 위해 공공기관에서 준수해야 하는 웹사이트 가이드 체크		
출 제 빈 도	미출제		
참 고 자 료	- 전자정부 웹사이트 UI·UX 가이드라인(행정안전부, 2019.03)		
K e y w o r d	- 사용자관점, 작업시간, 직관성, 일관성, 이용성, 오류최소화, 모든유형이용, 정보검색		
풀 이	조승우(124 회 정보관리기술사/alectos@naver.com)		

### 1. 사용자 경험 가이드, 전자정부 웹사이트 UI/UX 가이드라인 개요



- 전자정부 웹사이트 사용자에게 대한 이해를 바탕으로 한 표준화된 사용자경험 가이드를 제공함
- 전자정부서비스를 효율적으로 제공할 수 있는 사용자경험 설계를 지원하고, 궁극적으로 사용자에게 최적의 사용경험을 제공하는 데 그 목적으로 함
- 전자정부 웹사이트 UI/UX 가이드라인에는 설계원칙, 설계기준, 설계가이드, 품질진단서를 가이드하고 있음.

### 2. 전자정부 웹사이트 UI/UX 설계기준 7 가지 설명

분류	설계기준 7 가지	설 명
사용자관점	1. 사용자에게 필요한 기능과 정보를 제공한다.	- 사용자 관리를 포함한 관리 및 접근 제어 제공 - 사용자가 목적을 달성할 수 있도록 충분한 양의 정보와 기능을 제공 - 중요한 정보와 기능을 효과적으로 이해하기 쉽게 표현
작업시간	2. 작업에 소요되는 시간과 단계를 최소화 한다.	- 웹사이트는 사용자가 요청한 작업에 대해 적절한 속도로 반응 - 사용자가 원하는 정보 및 기능에 빠르게 접근 - 하나의 과업을 완료하기 위해, 동일한 작업을 여러 번 반복하지 않아야 함. - 작업 수행에 불필요한 단계가 포함되어 있지 않아야 함.
직관성,	3. 직관적이고 일관성	- 모든 정보와 기능은 논리적 흐름에 적합하도록 구성

일관성	있게 만든다	<ul style="list-style-type: none"> <li>- 사용자가 이해하기 쉬운 단어와 문장을 사용</li> <li>- 용어, 스타일, 레이아웃, 기능의 작동방식 등은 전체 페이지에서 일관성 있게 적용</li> <li>- 정보의 표현과 기능의 작동방식은 일반적인 사용자의 기대에 부합</li> <li>- 사용자의 행동에 도움이 되는 단서 및 정보를 적절한 시기와 위치에 제공</li> </ul>
사용자 이용성	4. 사용자가 원하는 방식으로 이용할 수 있게 만든다.	<ul style="list-style-type: none"> <li>- 사용자의 행동과 웹사이트의 반응 사이에 불필요한 요소제거</li> <li>- 사용자는 원하는 입출력 도구를 자유롭게 사용</li> <li>- 사용자가 선호에 따라 웹사이트의 상태 및 설정을 변경가능</li> <li>- 사용자에게 다양한 맞춤형 정보를 제공</li> </ul>
사용자 오류 최소화	5. 사용자가 실수하지 않게 만든다.	<ul style="list-style-type: none"> <li>- 사용자 실수를 유발하는 요소 유무</li> <li>- 심각한 결과를 유발하는 기능의 실행여부를 사용자에게 확인</li> <li>- 사용자가 오류를 빠르게 인지할 수 있도록 표현</li> <li>- 동작을 취소하거나 이전 상태에 돌아갈 수 있는 수단 제공</li> </ul>
모든 유형 이용	6. 모든 유형의 사용자가 이용할 수 있게 만든다.	<ul style="list-style-type: none"> <li>- 장애가 있는 사용자에게 비장애 사용자와 동등한 수준의 정보와 기능을 제공</li> <li>- 다양한 웹 브라우저에서 모든 기능을 동등하게 이용</li> <li>- 초보 사용자와 숙련된 사용자가 웹사이트에서 제공되는 모든 기능을 편리하게 이용</li> </ul>
정보검색	7. 원하는 서비스와 정보를 쉽게 찾을 수 있게 만든다.	<ul style="list-style-type: none"> <li>- 사용자가 원하는 페이지나 위치로 이동할 수 있는 여러가지 수단을 제공</li> <li>- 일반적으로 사용되는 내비게이션 모형을 적절하게 구현</li> <li>- 중요한 정보와 다른 정보 간 위계를 명료하게 구분</li> </ul>

- 전자정부 웹사이트 UI/UX 설계기준 7 가지를 바탕으로 웹 설계 및 개발 시 웹접근성, 웹호환성, 웹개방성, 웹최적화를 적용해야함 .

### 3. 웹접근성, 웹호환성, 웹개방성, 웹최적화 설명

#### 가. 웹접근성, 웹호환성 설명

분류	구분	설 명
웹접근성	- 정의	- 장애인과 비장애인이 동등하게 전자정부 웹사이트 콘텐츠에 접근할 있는 가를 보는 방법
	- 진단지표	- 한국형 웹 콘텐츠 접근성 지침 기반 24 개 지표, 24 개 진단기준
	- 진단지표 및 진단기준 예시	<ul style="list-style-type: none"> <li>- 대체 텍스트 제공: 텍스트 아닌 콘텐츠에는 대체 텍스트를 제공하는가?</li> <li>- 자동 재생 금지: 3 초 이상 자동으로 소리가 재생되는 콘텐츠를 제공하지 않는가?</li> <li>- 키보드 사용보장: 모든 기능은 키보드만 사용할 수 있는가?</li> </ul>
	- 진단방법	- 수동진단 혹은 자동진단
웹호환성	- 정의	- 전자정부 웹사이트가 이용환경에 상관없이 모든 사용자를 수용 할



		수 있는지를 평가
	- 진단지표	- 웹호환성 진단지표 기반의 3 개 지표, 6 개 진단기준
	- 진단지표 및 진단기준	- 웹 표준 문법 준수: 표준 HTML 문법을 준수하는가? - 웹호환성 확보: 다양한 웹브라우저에 따른 기능 호환성을 확보하였는가? - 비표준 기술제거: 비표준기술(액티브 X 등) 없이 웹사이트의 모든 기능이 동작하는가?
	- 진단방법	- 수동진단 혹은 자동진단

- 웹접근성은 장애인과 비장애인이 동등하게 전자정부 웹사이트 콘텐츠에 접근할 수 있는가 평가함
- 웹호환성은 전자정부 웹사이트가 이용환경에 상관없이 모든 사용자를 수용할 수 있는가 평가함

#### 나. 웹개방성, 웹최적화 설명

분류	구분	설 명
웹개방성	- 정의	- 전자정부 웹사이트에 공개된 모든 정보에 아무런 제한 없이 접근할 수 있는지를 평가
	- 진단지표	- 4 개 진단지표, 4 개 진단기준
	- 진단지표 및 진단기준 예시	- 특정 검색 엔진 접근 차단: 특정 검색엔진에 대해 해당 페이지를 노출하지 않거나 웹사이트에서 웹 방화벽 등을 사용하여 검색 로봇을 차단하고 있는가? - 페이지별 URL 차단: 검색 결과에 대한 외부 공유를 제한하고 있는가?
	- 진단방법	- 자동진단
웹최적화	- 정의	- 사용자의 요청에 의해 웹사이트가 브라우저에 표시되는 데 까지의 응답시간의 적절성과 링크의 관리를 평가
	- 진단지표	- 2 개 진단지표, 3 개 진단기준
	- 진단지표 및 진단기준 예시	- 웹사이트 응답속도 최소화: 웹사이트 메인 페이지는 적절한 속도(3 초 이내)로 로딩되는가? - 웹사이트 링크관리: 웹사이트 링크정보는 링크가 단절(Dead link)되는 곳이 없도록 고나리하고 있는가?
	- 진단방법	- 자동진단

- 웹개방성은 사용자가 전자정부 웹사이트에 공개된 모든 정보에 아무런 제한 없이 접근할 수 있는지를 평가함
- 사용자의 요청에 의해 웹사이트가 브라우저에 표시되는 데까지의 응답 시간의 적절성과 링크의 관리를 평가

“끝”

## 기출풀이 의견

2. 전자정부 웹사이트 UI/UX 가이드의 전체적인 개념을 바탕으로 작성이 필요한 문제였습니다. 특히 장애인, 비장애인 등 UI/UX가 일반사용자 부분 뿐만아니라 대국민 서비스를 위한 차별 없는 웹사이트를 제공한다는 개념으로 작성해야 합니다. 최근 전자정부 웹사이트 품질관리 지침(행정안전부 고시 제2020-38호)가 고시가 되어 함께 학습하시면 좋을 것 같습니다.

## ※ 전자정부 웹사이트 자가 진단지표

구분	진단지표	내용	진단방법
웹 접근성	한국형 웹 콘텐츠 접근성 지침 기반의 24개 지표, 24개 진단기준	장애인과 비장애인이 동등하게 전자정부 웹사이트 콘텐츠에 접근할 수 있는가	자동진단 및 수동진단
웹 호환성	웹 호환성 진단지표 기반의 3개 지표, 6개 진단기준	전자정부 웹사이트 이용환경에 상관없이 모든 사용자를 수용할 수 있는가	자동진단 및 수동진단
웹 개방성	4개 진단지표, 4개 진단기준	사용자가 전자정부 웹사이트에 공개된 모든 정보에 아무런 제한 없이 접근할 수 있는가	자동진단
웹 최적화	2개 진단지표, 3개 진단기준	웹사이트가 브라우저에 표시되는데 걸리는 시간이 적절하고, 링크가 정확하게 연결되고 있는가	자동진단

## 웹접근성 자가진단표

구분	진단지표	진단기준	진단방법
웹 접근성	대체 텍스트 제공	텍스트 아닌 콘텐츠에는 대체 텍스트를 제공하는가?	수동진단
	멀티미디어 자막 제공	동영상, 음성 등 멀티미디어 콘텐츠를 이해할 수 있도록 자막, 대본 또는 수화를 제공하는가?	수동진단
	색에 무관한 콘텐츠 인식	콘텐츠(그래프, <u>차트</u> , 지도, 필수 입력항목 등)는 <u>색에 관계없이</u> 인식될 수 있도록 제공하는가?	수동진단
	명확한 지시사항 제공	지시사항은 모양, 크기, 위치, 방향, 색, 소리 등에 <u>관계없이</u> 인식될 수 있도록 제공하는가?	수동진단
	텍스트 콘텐츠의 명도 대비	텍스트 콘텐츠와 배경 간의 명도 대비는 4.5:1 이상을 만족하는가?	수동진단
	자동 재생 금지	3초 이상 자동으로 소리가 재생되는 콘텐츠를 제공하지 않는가?	수동진단
	콘텐츠 간의 구분	이웃한 콘텐츠가 <u>여백</u> , <u>테두리</u> , <u>구분선</u> , <u>명도대비</u> 등을 이용하여 구별되는가?	수동진단
	키보드 사용 보장	모든 기능은 키보드만으로도 사용할 수 있는가?	수동진단
	초점 이동	키보드에 의한 초점은 논리적으로 이동해야 하며 시각적으로 구별할 수 있는가?	수동진단
	조작 가능	사용자 입력 및 <u>컨트롤은 조작 가능하도록</u> 제공되는가?	수동진단
	응답시간 조절	시간제한이 있는 콘텐츠는 응답시간을 조절할 수 있는가?	수동진단

구분	진단지표	진단기준	진단방법
	정지 기능 제공	시간에 따라 자동으로 변경되는 콘텐츠는 정지, 이전, 다음 기능 등 움직임을 제어할 수 있는가?	수동진단
	깜빡임과 번쩍임 사용 제한	초당 3~50회 주기로 깜빡이거나 번쩍이는 콘텐츠를 제공하지 않는가?	수동진단
	반복 영역 건너뛰기	콘텐츠의 반복되는 영역은 건너 뛸 수 있는가?	수동진단
	제목 제공	페이지, 프레임, 콘텐츠 블록에는 적절한 제목을 제공하는가?	수동진단
	적절한 링크 텍스트	링크 텍스트는 용도나 목적을 이해할 수 있도록 제공하는가?	수동진단
	기본 언어 표시	기본 언어 표시 속성을 사용하여 주로 사용하는 언어를 명시하고 있는가?	수동진단
	사용자 요구에 따른 실행	사용자가 의도하지 않은 기능(새 창, 초점 변화 등)은 실행되지 않는가?	수동진단
	콘텐츠의 선행 구조	콘텐츠는 논리적인 순서로 제공하는가?	수동진단
	표의 구성	표는 이해하기 쉽게 구성되어 있는가?	수동진단
	레이블 제공	입력 서식에 대응하는 레이블을 제공하고 있는가?	수동진단
	오류 정정	입력 오류를 <u>정정할 수 있는 방법</u> 을 제공하는가?	수동진단
	마크업 오류 방지	마크업 언어의 요소는 <u>연고</u> , <u>단음</u> , 중첩 관계 및 속성 선언에 오류 없이 제공하는가?	수동진단
	웹 애플리케이션 접근성 준수	웹 애플리케이션은 자체 접근성을 준수하였거나 대체 콘텐츠를 제공하고 있는가?	수동진단

### 웹호환성 자가진단표

구분	진단지표	진단기준	진단방법
웹 호환성	웹 표준 문법 준수	표준 (X)HTML문법을 준수하는가?	수동진단
		표준 CSS문법을 준수하는가?	
	웹 호환성 확보	다양한 웹브라우저에 따른 기능 호환성을 확보하였는가?	다양한 웹브라우저를 활용한 수동진단
		다양한 웹브라우저에서 화면표시가 동등하게 구현 되었는가?	
	비표준 기술제거	비표준기술(액티브X 등) 없이 웹사이트의 모든 기능이 동작하는가?	자동진단
		최신 웹 표준 기술(HTML5)을 활용하여 웹사이트를 운영하고 있는가?	수동진단

## 웹개방성 자가진단표

구분	진단지표	진단기준	진단방법
웹 개방성	특정 검색엔진 접근 차단	특정 검색엔진에 대해 해당 페이지를 노출하지 않거나 웹사이트에서 웹 방화벽 등을 사용하여 검색 로봇을 차단하고 있는가?	수동 진단
	검색 로봇 접근 차단	웹사이트에 검색 로봇을 차단하여 외부 검색 사이트에 정보를 노출하지 못하도록 검색을 통제하고 있는가?	자동 진단
	페이지 정보 수집 거부	meta 태그에 noindex, nofollow 속성을 설정하여 페이지 정보를 공유하지 못하도록 차단하고 있는가?	수동 진단
	페이지별 URL 차단	검색 결과에 대한 외부 공유를 제한하고 있는가?	수동 진단

## 웹최적화 자가진단표

구분	진단지표	진단기준	진단방법
웹 최적화	웹사이트 응답 속도 최소화	웹사이트 메인 페이지는 적절한 속도(3초 이내)로 로딩되는가?	자동 진단
		웹사이트 메인 페이지의 용량은 적절한 크기(3MB)로 제공되고 있는가?	자동 진단
	웹사이트 링크 관리	웹사이트 링크 정보는 링크가 단절(Dead link)되는 곳이 없도록 관리하고 있는가?	자동 진단

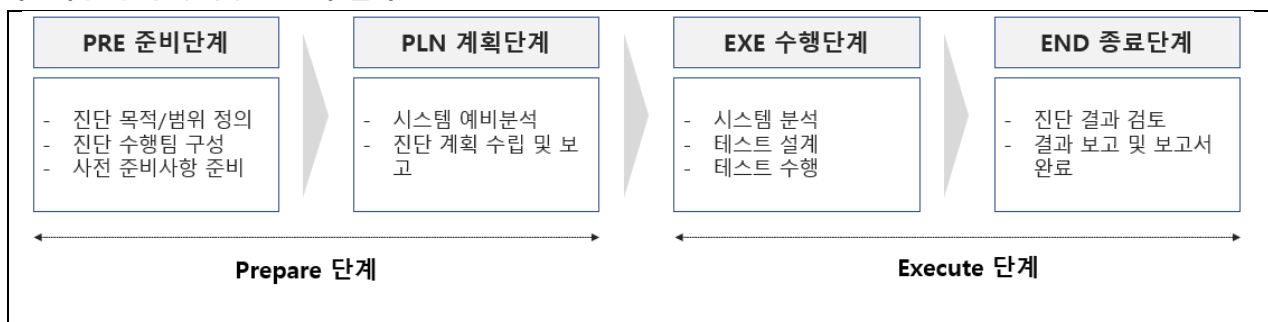
문 제	3. 최근 SW 가 대부분 산업에 적용되면서 소프트웨어 안전이 중요하게 여겨지고 있다. 소프트웨어 안전진단가이드(공통분야)에 따른 소프트웨어 안전성 진단 항목 중 기능 동작 정확성 진단의 절차 및 주요활동 내용을 설명하시오.		
출 제 영 역	소프트웨어	난 이 도	★★★★☆
출 제 배 경	- 4 차 산업혁명에 따른 IoT, 빅데이터, AI 등의 소프트웨어가 도입이 증가하면서 자동화된 서비스를 이용하여 편리한 Life 를 즐길 수 있지만 도입증가 따른 각종 보안, 오류사고등이 증가하면서 소프트웨어 안전이 더욱더 중요해지고 있음		
출 제 빈 도	미출제		
참 고 자 료	- 소프트웨어 안전진단 가이드-공통분야(한국정보통신기술협회, 2017.12)		
Key word	- 준비단계, 계획단계, 수행단계, 종료단계, 진단결과서, 잠재결함		
풀 이	조승우(124 회 정보관리기술사/alectos@naver.com)		

### 1. SW 안전성 향상을 위한 소프트웨어 안전진단 가이드의 개요

소프트웨어 안전진단 가이드 내용 및 범위	적용대상 시스템
<ul style="list-style-type: none"> <li>- SW 안전진단에 대한 <u>기본 개념 및 체계</u> 제시</li> <li>- SW 안전 <u>진단분야, 진단절차, 진단방법, 진단기법</u> 등 제시</li> <li>- 진단에 활용할 수 있는 <u>템플릿, 예제, 실무적 고려사항</u> 등 제시</li> </ul>	<ul style="list-style-type: none"> <li>- <u>국민 안전에 직결되거나 국민 안전 증진</u>을 위해 운영되고 있는 시스템</li> <li>- 시스템 장애 또는 오작동 시 국민 신체, 생명, 재산 피해나 사회적 혼란을 야기 또는 야기할 수 있는 시스템</li> </ul>
<ul style="list-style-type: none"> <li>- 소프트웨어 안전진단 가이드는 국가·공공기관 등에서 SW 안전진단을 기획, 추진 및 수행하는데 참고할 수 있는 실무 지침서</li> <li>- 「재난 및 안전관리 기본법」 26 조 따른 시스템, 「정보통신기반 보호법」 8 조 따른 주요정보통신기반시설 혹은 시스템 대상</li> </ul>	

### 2. 소프트웨어 안전성 진단 중 기능 동작 정확성 진단의 절차

#### 가. 기능 동작 정확성 진단의 절차



- 기능동작 정확성을 진단하기 위한 절차는 준비 단계, 계획 단계, 진단 수행 단계, 종료단계로 구분

#### 나. 기능 동작 정확성 진단의 절차

구분	단계	설 명
----	----	-----

Prepare 단계	- PRE 준비단계	- 기능동작 정확성 진단 목적 및 진단 수행 범위 등을 정의하고, 진단 이해관계자 식별 및 진단 수행팀을 구성하는 등 진단에 필요한 제반사항을 준비
	- PLN 계획단계	- 시스템 예비분석을 수행하여 테스트 전략, 방법, 일정 등에 대한 진단 세부계획을 수립하고 진단내용을 공유 및 보고
Execute 단계	- EXE 수행단계	- 테스트 대상 시스템을 상세히 분석하고 테스트가 필요한 기능 및 상황을 식별 - 테스트가 필요한 기능에 대해 테스트 및 시나리오를 설계하여 테스트를 수행하고 발견된 결함을 포함한 테스트 결과를 정리
	- END 종료단계	- 진단을 마무리하는 단계로 소프트웨어 테스트 결과 검출된 결함이 시스템 안전에 미치는 영향을 분석하고 해당 결과를 상호 검토하여 정리

- 소프트웨어 안전성 진단 중 기능동작의 정확성을 확인하기 위해서는 최초 진단목적 및 범위 정의에서부터 명확한 설정에서부터 시작해야함.

### 3. 소프트웨어 안전성 진단 중 기능 동작 정확성 주요 활동 내용

구분	상세 절차	설 명
(PRE) 준비단계	- 진단목적 및 범위 정의	- 진단 목적 정의 - 안전관련 표준, 지침 등 준수여부 확인 - 진단 환경 확인 및 범위(상위주순) 협의
	- 진단 이해관계자 식별 및 진단 수행팀 구성	- 진단 이해관계자 식별 - 진단 지원 인력 협조 요청 - 진단 수행팀 구성
	- 사전준비사항 협의 및 준비	- 사전 준비사항 협의 - 진단 환경, 산출물 등 사전준비 - 사전 준비사항 준비 상태 확인
(PLN) 계획단계	- 시스템 예비분석	- 시스템 현황 분석 - 시스템 형상변경 범위 식별 - 진단 범위 조정 및 재협의
	- 진단 계획 수립	- 진단 전략 및 방법 수립 - 진단 참여인력 구성 확정 및 역할 분담 - 진단 계획서(초안) 작성
	- 진단 계획 보고	- 계획보고 회의 준비 및 개최 - 진단계획서(초안) 보완 - 진단계획서 배포
(EXE) 수행단계	- 시스템 분석	- 시스템 상세분석 및 세부 기능리스트 도출 - 안전 관련기능 식별 - 위험 시나리오 도출 - 위험 시나리오 도출 및 안전관련 기능 식별 결과 검토
	- 테스트 설계	- 테스트케이스 및 테스트 시나리오 설계

		<ul style="list-style-type: none"> <li>- 테스트케이스 별 테스트 지원가능 여부 검토</li> <li>- 테스트케이스 별 테스트 수행 범위, 수행 일자 등 확정</li> </ul>
	- 테스트 수행	<ul style="list-style-type: none"> <li>- 테스트 환경 점검</li> <li>- 테스트 수행 및 결과 기록</li> <li>- 테스트 결과</li> </ul>
(END) 종료단계	- 진단결과 분석 및 검토	<ul style="list-style-type: none"> <li>- 진단결과 정리 및 공유</li> <li>- 진단결과 검토</li> <li>- 진단결과 보고서(초안) 작성</li> </ul>
	- 결과보고 및 보고서 완료	<ul style="list-style-type: none"> <li>- 결과 보고 회의 준비 및 개최</li> <li>- 진단 결과서(초안) 보완</li> <li>- 진단 결과 최종보고서 완료</li> </ul>

- 소프트웨어 안전성 진단을 통하여 잠재결합을 찾아내어 이를 개선하고 관리를 통해 안전한 소프트웨어를 만들도록 해야함.

"끝"



## 기출풀이 의견

3. 소프트웨어 안전진단 관련 부분은 4차산업에서는 앞으로 중요한 이슈로 계속 대두가 될 주제이기 때문에 관련 고시 및 시행령 등을 중점으로 확인할 필요가 있습니다. 추가적으로 소프트웨어 개발 보안가이드도 같이 참고해서 보시면 좋을 것 같습니다.



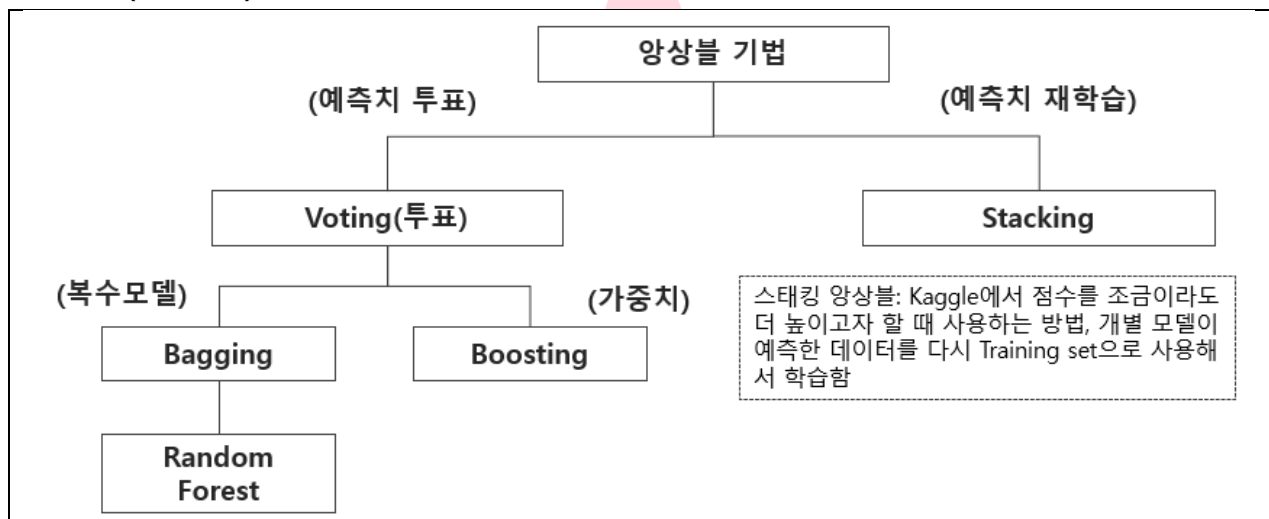
문 제	4. 앙상블(Ensemble) 모형의 Bagging 과 Boosting 알고리즘을 설명하시오.		
출 제 영 역	인공지능	난 이 도	★★☆☆☆
출 제 배 경	- 인공지능이 점차 다양한 알고리즘이 나오면서 여러 알고리즘을 혼합하여 보다 높은 정확도를 낼 수 있는 Needs 가 점차 증가함		
출 제 빈 도	120 회 정보관리 3 교시 출제		
참 고 자 료	- 도리의 디지털라이프 배경과 부스팅		
K e y w o r d	- Bootstrap, Classification, 랜덤샘플링, 변동성, 정확도		
풀 이	조승우(124 회 정보관리기술사/alectos@naver.com)		

## 1. 최적의 알고리즘 조합, 앙상블 모형의 개요

### 가. 앙상블(Ensemble) 모형의 정의

- 주어진 데이터로부터 여러 개의 모델을 학습한 다음, 예측 시 여러 모델의 예측 결과들을 종합하여 정확도를 높이는 기법

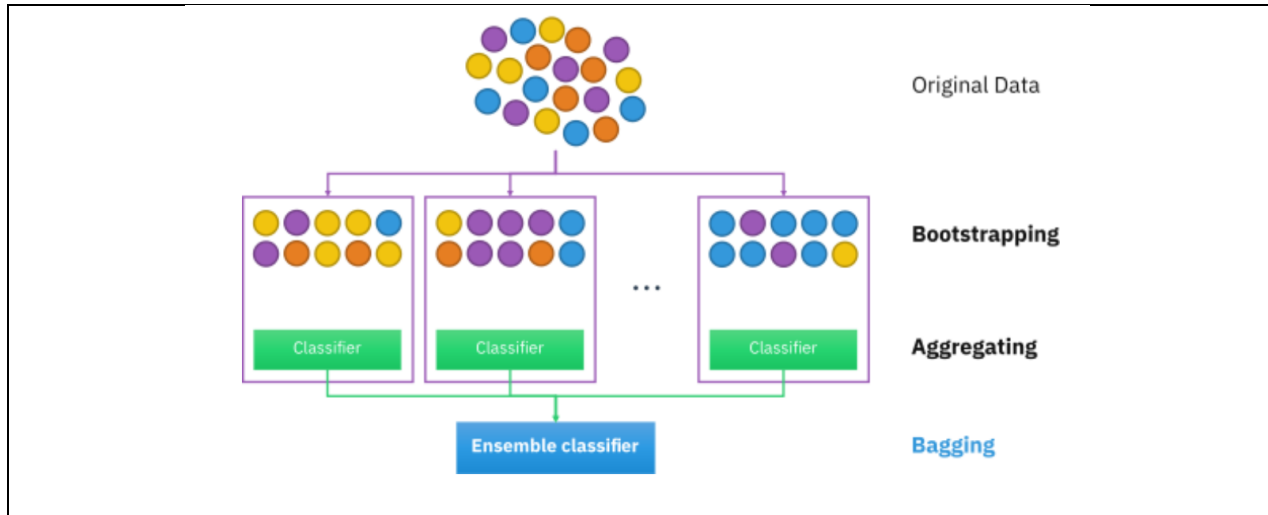
### 나. 앙상블(Ensemble) 모형의 유형



- Bagging 과 Boosting 알고리즘은 Voting 알고리즘에서 발전된 알고리즘.

## 2. 앙상블(Ensemble) 모형의 Bagging 알고리즘 설명

### 가. 복수모델 결합, Bagging 알고리즘의 개념



- Bagging 은 주어진 데이터에서 여러 개의 Bootstrap 자료를 생성하고, 각 자료를 모델링한 결합하여 최종 예측 모델을 만드는 알고리즘
- Bagging 은 부트스트랩(Bootstrap)을 집계(Aggregating)하여 모델을 만들어 냄.

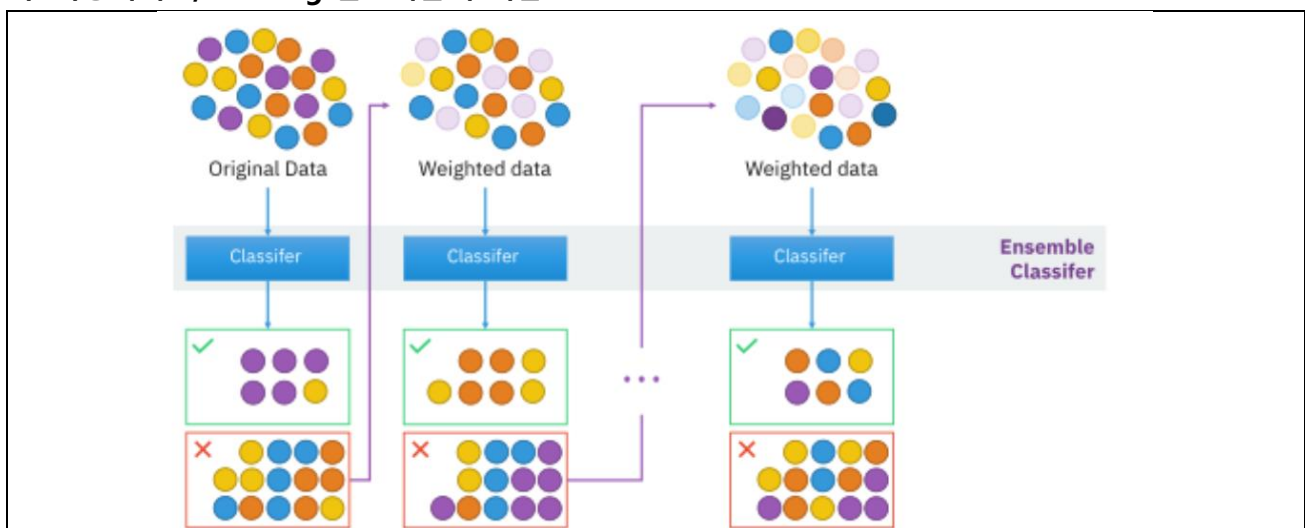
#### 나. Bagging 알고리즘의 절차

순서	단계	설 명
1	- Bootstrap 추출	- 원시데이터(Row Data)에서 bootstrap 데이터 추출 - bootstrap: 통계에서 모집단에서 표본을 추출한다는 개념으로 랜덤샘플링 (중복추출가능)
2	- 데이터 셋 생성	- 추출을 반복하여 n 개의 데이터 셋 생성
3	- 모델링	- 각 데이터를 각각 모델링하여 모델 생성
4	- Bagging 모델 생성	- 단일 모델을 결합하여 Bagging 모델 생성

- Bagging 은 부트스트랩(Bootstrap)을 집계(Aggregating) 하는 과정을 하는 알고리즘
- Bagging 은 Boosting 과 다르게 병렬로 진행함

### 3. 앙상블(Ensemble) 모형의 Boosting 알고리즘 설명

#### 가. 가중치기준, Boosting 알고리즘의 개념



- Boosting 은 잘못 분류된 개체들에 가중치를 적용, 새로운 분류규칙을 만들고, 이 과정을 반복해 최종 모델을 만드는 알고리즘

#### 나. Boosting 알고리즘의 절차

순서	단계	설 명
1	- 모델생성	- 원시데이터(Row Data)에서 동일가중치로 모델 생성
2	- 오분류 데이터	- 생성된 모델로 인한 오분류 데이터 수집
3	- 높은 가중치	- 오분류 데이터에 높은 가중치 부여
4	- 반복	- 과정 반복을 통하여 모델의 정확도 향상

- Boosting 은 Bagging 과 다르게 순차적으로 진행되며 복원추출로 가중치를 부여함
- Bagging 은 여러 번의 샘플링을 통해 분산을 줄여 모델 변동성을 감소시킴, Boosting 은 잘못 분류된 데이터에 집중해 모델의 정확도를 향상시킴

#### 4. Bagging 과 Boosting 알고리즘 기술적 특징 비교

항목	Bagging	Boosting
수행원리	- 샘플링에 의한 결합	- 가중치 재조정 반복
수행목적	- 모델 변동성 감소	- 모델 정확도 향상
적용연산	- 평균 및 다수결 원칙	- 가중치 선형 결합
초기모델	- Bootstrap 모델	- Weak Classification
최종모델	- Bagging 모델	- String Classification
분류성능	- 결측치 존재시 우수	- 데이터 다수 시 우수

- 데이터 마이닝의 분류 문제 해결 시 목표 변수 예측 모델 생성, 특성 별 모델 생성 알고리즘 선택 적용 필요

끝"

#### 기출풀이 의견

4. Bagging과 Boosting이 120회에 이어 재출제가 되었습니다. 그렇기 때문에 기타 Stacking 및 Random Forest, Voting 등의 알고리즘도 추가로 학습하여 재 출제에 대비할 필요가 있습니다.

문 제	5. 소프트웨어 구조의 평가 및 개선을 위한 소프트웨어 아키텍처 분석 방법에 대하여 다음 내용을 설명하시오.		
	1) 소프트웨어 아키텍처 분석의 필요성 2) 소프트웨어 아키텍처 정방향 분석과 역방향 분석 개념 3) 소프트웨어 아키텍처 평가 기법중 ATAM(Architecture Trade-off Analysis Method)		
출 제 영 역	소프트웨어	난 이 도	★★★★☆
출 제 배 경	- 점차 소프트웨어가 복잡해지면서 소프트웨어 아키텍처를 통한 의사소통 및 소프트웨어 Risk 감소, 재사용성 촉진 통한 비용 감소가 중요해짐에 따라 출제됨		
출 제 빈 도	80 회 응용 출제		
참 고 자 료	- SW 아키텍처 참조모델(NIPA SW 공학센터, 2014)		
K e y w o r d	- 의사소통, 비용절감, 유틸리티트리, 시나리오, CBAM, SAAM, ADR, ARID, 정적분석, 동적분석		
풀 이	조성준(120 회 정보관리기술사/alectos@naver.com)		

### 1. 소프트웨어 아키텍처 분석의 필요성

구분	필요성	설 명
요구사항 관리	- 의사소통 도구	- 이해당사자간의 다양한 관점을 충족시키는 일관된 내용 제공
	- 비용절감	- 변경발생 가능성이 줄어들어 비용절감 효과 발생
	- 요구사항 체크	- 아키텍처의 의사결정이 시스템의 진화에 따라 변경 타당성을 파악하고 관리하게 도움 - 기술, 플랫폼 독립적인 모형에 기반하여 변경되는 IT 환경 변경 유연성 확보
품질관리	- 문제점 해결	- 비논리적 요구사항, 소프트웨어 성능 등 품질 속성문제, 잠재적 위험요소 문제를 조기 발견하고 대응 가능
	- 품질예측	- 시스템 품질 예측에 도움을 줌 - 시스템 품질 속성을 실현 지원

- 소프트웨어 아키텍처는 품질 속성에 미치는 영향을 파악 및 판단하여 전체 아키텍처의 적합성을 평가하는 표준 절차
- 소프트웨어 아키텍처 분석하기 위한 방법으로 정방향 및 역방향 분석방법이 있음

### 2. 소프트웨어 아키텍처 정방향 분석과 역방향 분석 개념

#### 가. 소프트웨어 아키텍처 정방향 분석과 역방향 분석 개요

구분	정방향 분석	역방향 분석
정의	- 제시된 소프트웨어 아키텍처가 개발될 소프트웨어에 대해서 요구되는 품질특성을 충족시킬 수 있는 가를 아키텍처 수준에서	- 프로그램 소스를 분석하여 다시 아키텍처 구조로 재생산하는 아키텍처 분석기법

	분석하는 기법	
시기	- 프로젝트 초반 혹은 개발초기, 소프트웨어 구체적으로 실체화/구현화/테스트 되기 이전	- 실제 개발 진행 중인 시점 중간마다 실시함
목적	- 소프트웨어 아키텍처 적합성 여부 분석해 향후 소프트웨어 아키텍처 채택함으로 인한 위험 최소화 목적	- 개발 중인 소프트웨어가 요구사항에 맞게 개발이 되고 있는지 확인 목적
종류	- ATAM, CBAM	- 리버스 엔지니어링, 소스코드분석, 클래스 다이어그램 분석
장점	- 소프트웨어 Risk 관리 용이함 - 이해관계자 간의 의사소통 도구로 활용함 - 높은 품질 이끌어 낼 수 있음	- 의존성이 적은 코드 구성가능 - 정량적으로 문제가 표시되어 관리 가능 - 여러 사람들의 코드 이해도를 높일 수있음
단점	- 이해관계자 간의 접근 방향 선정에 많은 시간 소요됨	- 소스 코드에 한정된 단편적인 뷰만을 제공 - 실제 현장에 발생하는 다양한 Trade-off 들을 제대로 반영하지 못할 수 있음

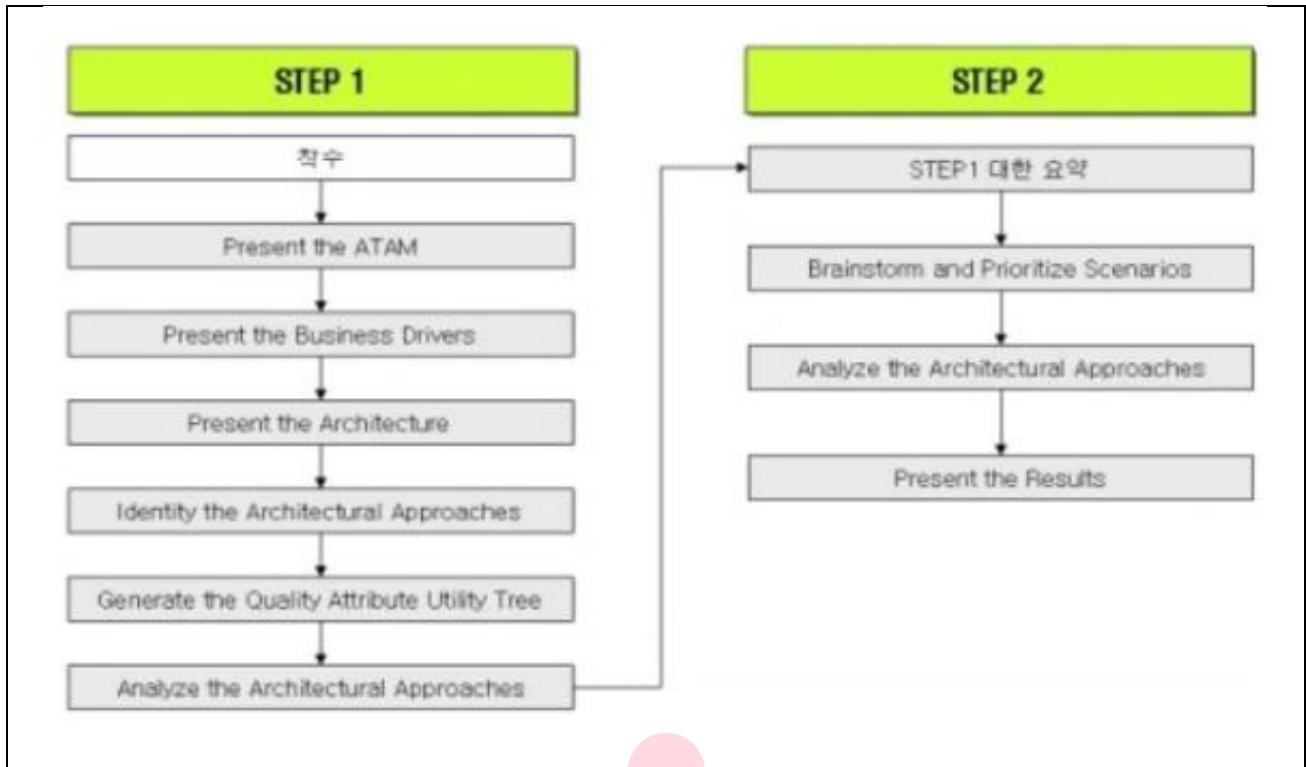
#### 나. 소프트웨어 아키텍처 정방향 분석과 역방향 분석의 종류

분류	분석 기법	설 명
정방향 분석	- ATAM	- 품질 목표 간의 Trade-off 가 있는지 파악가능한 아키텍처 평가방법 - SAAM 을 계승하여 발전
	- CBAM	- 경제적 의사결정에 대한 요구를 충족시키기 위해 ATAM 을 바탕으로 소프트웨어 아키텍처 분석에 중점을 둔 경제적 모델링 기법 - ATAM 에 부족한 경제성 평가 보강
	- SAAM	- 변경 용이성, 기능 집중에 부분 평가, 평가가 용이함
	- ADR	- 아키텍처 구성요소간 응집도 평가
	- ARID	- 특정 부분에 대한 품질요소 집중
역방향 분석	- 정적분석	- 소프트웨어가 실행되지 않는 환경 하, 소스 코드 의미를 분석, 결함을 찾아내는 분석 기법
	- 동적분석	- 소프트웨어가 실행중인 환경 하, 다양한 입/출력 데이터, 사용자 상호작용의 변화들을 점검
	- 바이너리 구조 분석	- 역공학 분석 도구를 이용
	- 시멘틱 분석	- 인터페이스 & 함수 호출의 구조적 취약점을 분석
	- PMD	- 응용 프로그램 코드에서 발견 된 문제를 보고 하는 오픈 소스 정적 JAVA 소스코드 분석기
	- 리버스 엔지니어링	- 소스코드를 분석하여 소프트웨어 아키텍처를 역으로 분석하는 기법

- 위의 기법 들 중에서 프로젝트 초기 품질목표들 가의 Trade-off 가 요구사항에 맞게 적절하게 되었는지 파악하기 위하여 ATAM 기법을 주로 활용함

### 3. 소프트웨어 아키텍처 평가기법 ATAM(Architecture Trade-off Analysis Method)

## 가. 소프트웨어 아키텍처 평가기법 ATAM의 개념



- 소프트웨어 아키텍처가 특정 품질 목표를 만족하는지 여부와 품질 목표들 간에 발생하는 충돌에 대해서 분석하는 평가 기법

## 나. 소프트웨어 아키텍처 평가기법 ATAM의 개념

단계	단계	설 명
Step1	- Present the ATAM	- 평가팀이 모든 이해 관계자에게 평가 절차에 대해서 설명하는 단계
	- Present the Business Drivers	- 프로젝트 PM 이 비즈니스 관점에서 평가팀에게 시스템에 대한 설명 - 가장 중요한 기능적 요구사항, 기술적 - 기술적, 관리적, 정치적 제약사항 - 비즈니스 목표, 중요 이관계자
	- Present the Architecture	- 아키텍트가 평가팀에 아키텍처에 관한 설명하는 단계 - OS, 하드웨어, 미들웨어 등 이미 결정된 기술적인 제약사항 - 연동해서 사용해야 하는 다른 시스템 - 품질 요소를 만족시키기 위해 사용한 아키텍처 접근방법
	- Identity the Architectural Approaches	- 아키텍트가 소개한 아키텍처 고유의 접근 방법과 아키텍처 스타일 파악하는 단계
	- Generate the Quality Attribute Utility Tree	- 평가팀, 아키텍처 팀, 프로젝트 매니저 등이 협력하여 유틸리티 트리를 작성하는 단계 - 유틸리티 트리를 통해 평가팀이 아키텍처의 어느 부분에 집중해야 할지 파악하게 됨

		<div><div>Utility</div><div><div>Performance</div><div><div>Data latency</div><div><div>(M,L) Minimize storage latency on customer DB to 200 ms.</div><div>(H,M) Deliver video in real time</div></div></div><div>Transaction throughput</div><div><div>(M,M) Maximize average throughput to the authentication server</div></div></div><div><div>Modifiability</div><div><div>New product categories</div><div><div>(L,H) add CORBA middleware in &lt; 20 person-months</div><div>(H,L) Change Web user interface in &lt; 4 person-weeks</div></div></div><div>Change COTS</div></div><div><div>Availability</div><div><div>Hardware failure</div><div><div>(L,H) Power outage at Site 1 requires traffic redirect to Site 3 in &lt; 3 seconds</div><div>(M,M) Restart after disk failure in 5 minutes</div></div></div><div>COTS software failure</div><div><div>(H,M) Network failure is detected and recovered in &lt; 1.5 minutes</div><div>(L,H) Credit card transactions are secure 99.999% of time</div></div></div><div><div>Security</div><div><div>Data confidentiality</div><div><div>(L,H) Customer database authorization works 99.999% of time</div></div></div><div>Data integrity</div></div></div> <tr><td rowspan="2">- Analyze the Architectural Approaches</td><td><div><div>- 유틸리티 트리를 통해 평가 범위 결정한후, 아키텍처 접근 방법과 결정사항을 평가하는 단계</div><div><div><div>아키텍처 접근법 분석서</div><table><tr><td>시나리오 번호</td><td>A05</td><td>시나리오</td><td colspan="2">프라이머리 CPU에 발생하는 하드웨어 장애를 감지해서 복구한다.</td></tr><tr><td>품질속성</td><td>가용성(Availability)</td><td></td><td></td><td></td></tr><tr><td>환경</td><td>일반 운영 환경</td><td></td><td></td><td></td></tr><tr><td>자극</td><td>CPU 가운데 하나가 장애를 일으킴</td><td></td><td></td><td></td></tr><tr><td>대응</td><td>장애CPU가 처리하던작업을 다른CPU로 넘긴다 이때 99.9999% 가용성을유지해야한다</td><td></td><td></td><td></td></tr><tr><td>아키텍처 판단</td><td>민감점</td><td>결충점</td><td>위험</td><td>무위험</td></tr><tr><td>CPU백업</td><td>S2</td><td></td><td>R8</td><td></td></tr><tr><td>데이터채널을백업하지않음</td><td>S3</td><td>T3</td><td>R9</td><td></td></tr><tr><td>Watchdog</td><td>S4</td><td></td><td></td><td>N12</td></tr><tr><td>Heartbeat</td><td>S5</td><td></td><td></td><td>N13</td></tr><tr><td>Fallover Routing</td><td>S6</td><td></td><td></td><td>N14</td></tr></table><div><div>근거</div><div><div><div>❖ 다른 하드웨어와운영시스템을도입해서같은원인으로동시에 장애에빠지지 않도록한다 (위험R8 완화)</div><div>❖ 최악의경우라도4초안에작업을넘길 수 있다</div><div>❖ Heartbeat 과watchdog 방식을둘 때2초 간격으로장애를감지할수 있다</div><div>❖ Watchdog은 간단하고신뢰성이높을것이다</div><div>❖ 데이터채널을백업하지않으면가용성요구사항을지키지못할수 있다.</div></div></div><div><div>아키텍처 다이어그램</div><div><div><div>프라이머리CPU (OS1)</div><div>↓ 1초 간격 heartbeat</div><div>Watchdog을 가진백업 CPU (OS2)</div><div>↓</div><div>스위치CPU (OS3)</div></div></div></div></div></div><tr><td></td><td></td></tr></div></div></td></tr>	- Analyze the Architectural Approaches	<div><div>- 유틸리티 트리를 통해 평가 범위 결정한후, 아키텍처 접근 방법과 결정사항을 평가하는 단계</div><div><div><div>아키텍처 접근법 분석서</div><table><tr><td>시나리오 번호</td><td>A05</td><td>시나리오</td><td colspan="2">프라이머리 CPU에 발생하는 하드웨어 장애를 감지해서 복구한다.</td></tr><tr><td>품질속성</td><td>가용성(Availability)</td><td></td><td></td><td></td></tr><tr><td>환경</td><td>일반 운영 환경</td><td></td><td></td><td></td></tr><tr><td>자극</td><td>CPU 가운데 하나가 장애를 일으킴</td><td></td><td></td><td></td></tr><tr><td>대응</td><td>장애CPU가 처리하던작업을 다른CPU로 넘긴다 이때 99.9999% 가용성을유지해야한다</td><td></td><td></td><td></td></tr><tr><td>아키텍처 판단</td><td>민감점</td><td>결충점</td><td>위험</td><td>무위험</td></tr><tr><td>CPU백업</td><td>S2</td><td></td><td>R8</td><td></td></tr><tr><td>데이터채널을백업하지않음</td><td>S3</td><td>T3</td><td>R9</td><td></td></tr><tr><td>Watchdog</td><td>S4</td><td></td><td></td><td>N12</td></tr><tr><td>Heartbeat</td><td>S5</td><td></td><td></td><td>N13</td></tr><tr><td>Fallover Routing</td><td>S6</td><td></td><td></td><td>N14</td></tr></table><div><div>근거</div><div><div><div>❖ 다른 하드웨어와운영시스템을도입해서같은원인으로동시에 장애에빠지지 않도록한다 (위험R8 완화)</div><div>❖ 최악의경우라도4초안에작업을넘길 수 있다</div><div>❖ Heartbeat 과watchdog 방식을둘 때2초 간격으로장애를감지할수 있다</div><div>❖ Watchdog은 간단하고신뢰성이높을것이다</div><div>❖ 데이터채널을백업하지않으면가용성요구사항을지키지못할수 있다.</div></div></div><div><div>아키텍처 다이어그램</div><div><div><div>프라이머리CPU (OS1)</div><div>↓ 1초 간격 heartbeat</div><div>Watchdog을 가진백업 CPU (OS2)</div><div>↓</div><div>스위치CPU (OS3)</div></div></div></div></div></div><tr><td></td><td></td></tr></div></div>	시나리오 번호	A05	시나리오	프라이머리 CPU에 발생하는 하드웨어 장애를 감지해서 복구한다.		품질속성	가용성(Availability)				환경	일반 운영 환경				자극	CPU 가운데 하나가 장애를 일으킴				대응	장애CPU가 처리하던작업을 다른CPU로 넘긴다 이때 99.9999% 가용성을유지해야한다				아키텍처 판단	민감점	결충점	위험	무위험	CPU백업	S2		R8		데이터채널을백업하지않음	S3	T3	R9		Watchdog	S4			N12	Heartbeat	S5			N13	Fallover Routing	S6			N14		
- Analyze the Architectural Approaches	<div><div>- 유틸리티 트리를 통해 평가 범위 결정한후, 아키텍처 접근 방법과 결정사항을 평가하는 단계</div><div><div><div>아키텍처 접근법 분석서</div><table><tr><td>시나리오 번호</td><td>A05</td><td>시나리오</td><td colspan="2">프라이머리 CPU에 발생하는 하드웨어 장애를 감지해서 복구한다.</td></tr><tr><td>품질속성</td><td>가용성(Availability)</td><td></td><td></td><td></td></tr><tr><td>환경</td><td>일반 운영 환경</td><td></td><td></td><td></td></tr><tr><td>자극</td><td>CPU 가운데 하나가 장애를 일으킴</td><td></td><td></td><td></td></tr><tr><td>대응</td><td>장애CPU가 처리하던작업을 다른CPU로 넘긴다 이때 99.9999% 가용성을유지해야한다</td><td></td><td></td><td></td></tr><tr><td>아키텍처 판단</td><td>민감점</td><td>결충점</td><td>위험</td><td>무위험</td></tr><tr><td>CPU백업</td><td>S2</td><td></td><td>R8</td><td></td></tr><tr><td>데이터채널을백업하지않음</td><td>S3</td><td>T3</td><td>R9</td><td></td></tr><tr><td>Watchdog</td><td>S4</td><td></td><td></td><td>N12</td></tr><tr><td>Heartbeat</td><td>S5</td><td></td><td></td><td>N13</td></tr><tr><td>Fallover Routing</td><td>S6</td><td></td><td></td><td>N14</td></tr></table><div><div>근거</div><div><div><div>❖ 다른 하드웨어와운영시스템을도입해서같은원인으로동시에 장애에빠지지 않도록한다 (위험R8 완화)</div><div>❖ 최악의경우라도4초안에작업을넘길 수 있다</div><div>❖ Heartbeat 과watchdog 방식을둘 때2초 간격으로장애를감지할수 있다</div><div>❖ Watchdog은 간단하고신뢰성이높을것이다</div><div>❖ 데이터채널을백업하지않으면가용성요구사항을지키지못할수 있다.</div></div></div><div><div>아키텍처 다이어그램</div><div><div><div>프라이머리CPU (OS1)</div><div>↓ 1초 간격 heartbeat</div><div>Watchdog을 가진백업 CPU (OS2)</div><div>↓</div><div>스위치CPU (OS3)</div></div></div></div></div></div><tr><td></td><td></td></tr></div></div>	시나리오 번호		A05	시나리오	프라이머리 CPU에 발생하는 하드웨어 장애를 감지해서 복구한다.		품질속성	가용성(Availability)				환경	일반 운영 환경				자극	CPU 가운데 하나가 장애를 일으킴				대응	장애CPU가 처리하던작업을 다른CPU로 넘긴다 이때 99.9999% 가용성을유지해야한다				아키텍처 판단	민감점	결충점	위험	무위험	CPU백업	S2		R8		데이터채널을백업하지않음	S3	T3	R9		Watchdog	S4			N12	Heartbeat	S5			N13	Fallover Routing	S6			N14				
	시나리오 번호	A05	시나리오	프라이머리 CPU에 발생하는 하드웨어 장애를 감지해서 복구한다.																																																									
품질속성	가용성(Availability)																																																												
환경	일반 운영 환경																																																												
자극	CPU 가운데 하나가 장애를 일으킴																																																												
대응	장애CPU가 처리하던작업을 다른CPU로 넘긴다 이때 99.9999% 가용성을유지해야한다																																																												
아키텍처 판단	민감점	결충점	위험	무위험																																																									
CPU백업	S2		R8																																																										
데이터채널을백업하지않음	S3	T3	R9																																																										
Watchdog	S4			N12																																																									
Heartbeat	S5			N13																																																									
Fallover Routing	S6			N14																																																									

Step2	- Brainstorm and Prioritize Scenarios	<div><div>- 전체 이해관계자로부터 시나리오를 생성하는 단계</div><div>- 전체 이해관계자가 참여하는 투표 프로세스를 통해 우선순위 정함</div><div>- 유틸리티 트리는 시나리오를 위한 하향식 메커니즘 제공</div><div>- 시나리오 Brainstorming 은 상향식 접근법 시나리오 생성</div></div>
	- Analyze the Architectural Approaches	<div><div>- 선정된 높은 우선순위의 시나리오를 분석</div><div>- 부가적인 아키텍처 접근 방법, 위험 요소, 민감 요소 찾아냄</div></div>
	- Present the Results	<div><div>- Step1~2 를 통해 수행한 평가결과에 대해 최종 보고서 작성</div></div>

- ATAM 은 소프트웨어 개발 주기의 초기 단계에서 수행할 수 있으며, 구현이 아닌 설계된 디자인을 평가하기 때문에 상대적으로 비용이 저렴하고 빠르다는 장점 있음.

“끝”

## 기출풀이 의견

5. 소프트웨어 아키텍처 평가가 오랜만에 나왔습니다. 향후에도 지속적으로 나올 수 있으니 ATAM이 외에 SAAM, CBAM, ADR, ARID 등도 학습해둘 필요가 있겠습니다.





문 제	6. 디지털 뉴딜 '데이터 댐'의 핵심인 인공지능 학습용 빅데이터 구축 사업의 성공적 추진을 위한 정책과 데이터 품질에 대한 관심도가 높아지고 있다. 다음의 내용을 설명하시오. 1) 인공지능 학습용 데이터의 특성 2) 데이터 획득, 정제 방법과 기준 3) 데이터 라벨링(Data Labeling) 및 어노테이션(Annotation) 방식		
출 제 영 역	디지털서비스	난 이 도	★★★☆☆
출 제 배 경	- 디지털 뉴딜 관련 데이터 댐 학습용 사업이 지속되면서 데이터 품질에 대한 문제가 출제됨		
출 제 빈 도	미출제		
참 고 자 료	- 인공지능 학습용 데이터 품질관리 가이드라인 v1.0(NIA, 2021.2) - 인공지능 학습용 데이터셋 구축 안내서(NIA, 2021.2)		
Key word	- 비정형데이터, 획득, 정제, 학습, 개인정보, 라벨링, 어노테이션, 바운딩		
풀 이	조승우(124 회 정보관리기술사)		

## 1. 인공지능 학습용 데이터의 특성

### 가. 인공지능 학습용 데이터의 특성

분류	특성	설 명
데이터 측면	- 비정형 데이터 다수	- 텍스트, 음성, 이미지, 동영상 등 주로 비정형데이터가 다수를 차지 함
	- 파일단위 관리	- 데이터의 저장구조가 전통적인 DB 에 저장하는 방식이 아닌 인공지능 학습에 필요한 데이터 파일 단위로 관리
상이한 품질 측면	- 데이터 획득 측면	- 임무정의, 획득, 정제, 라벨링 등의 구축 단계를 거치면서 목적에 부합하는 데이터를 획득됨
	- 데이터 정제 측면	- 정제과정을 통해 기계학습에 적합한 데이터로 가공
	- 데이터 학습 측면	- 정제된 데이터에 학습을 위한 참값(Ground Truth)을 부여
품질 결정 특성	- 구축과정에서 결정	- 인공지능 학습용 데이터의 품질은 80%~90% 이상이 구축과정에서 결정
평가 특성	- 데이터 품질수준 평가	- 학습된 인공지능 모델의 성능 측정 결과로 데이터의 품질 수준을 평가하려는 특성

- 인공지능 학습용 데이터의 품질 확보를 위해서는 인공지능 학습용 데이터의 특성을 고려해야 하며, DB 에 저장하는 구조화된 데이터(Structured Data)를 대상으로 하는 일반적인 데이터 품질관리와는 차별화된 품질관리 체계를 확보하는 것이 중요함

### 나. 인공지능 학습용 데이터와 DB 데이터의 비교

구분	인공지능 학습용 데이터	DB 데이터
구축목적	- 인공지능 모델의 학습 목적	- 정보 검색 및 활용 목적
데이터유형	- 소리, 영상, 이미지, 자연어 등 비정형	- 텍스트 데이터 중심의 정형 데이터 중심

	데이터 중심	
데이터구조	- 원천데이터와 라벨링데이터가 쌍으로 구성	- 테이블과 컬럼으로 구성
데이터 구축절차	- 임무정의 > 획득 > 정제 > 라벨링 > 학습 > 저장	- 수집 > 정제 > 변환 > 저장
데이터 관리방식	- 파일형태로 저장 (폴더+JSON 파일)	- 범용적인 DBMS 에 저장 (테이블+컬럼구조)

- 인공지능 학습용 데이터의 획득, 정제에 따라 상이한 데이터 품질을 가지고 있기 때문에 이를 관리할 필요가 있음.

## 2. 데이터 획득, 정제 방법과 기준

### 가. 데이터 획득 방법

분류	방법	설 명
공통 데이터 획득	- 원시데이터 직접 제작	- 녹취, 필사 등으로 데이터 수집
	- 수행 및 참여기관 내·외부에 있는 데이터 수집	- API, 크롤링, 직접 수령 등 방법으로 수집
텍스트 데이터 획득	- API	- 수집서버를 이용하여 JSON 으로 TXT 수집
	- 웹크롤링	- 크롤링 서버를 이용하여 여러 텍스트 포맷을 TXT 변환 수집
이미지 데이터 획득	- 촬영	- 직접 내/외부에서 촬영하여 데이터 획득 - 사전에 계획을 수립하여 일정한 형식을 맞춤 필요 있음
	- 스캔	- 책, 글씨체 등 각종 문서를 스캔하여 수집
	- 클라우드 소싱	- 일정한 서식을 만들거나 작업 안내서 등을 작업자에게 배포하여 수집
음성데이터 수집	- 녹음	- 일정 포맷 형태로 음성 녹음
영상 데이터 수집	- 영상 촬영	- 획득계획을 수립하여 영상데이터 수집함

- 수집된 데이터를 학습용 데이터로 변환하기 위하여 정제프로세스 수립이 필요함

### 나. 데이터 정제 방법과 기준

분류	방법	설 명
정제 방식	- 정제 프로세스 수립	- 어노테이션 단계에 들어가기 전에 학습용 데이터로 적합한 데이터를 선별하고 처리하는 정제 프로세스를 획득방법별로 수립
	- 도구 이용	- 도구(소프트웨어)를 활용하여 정해진 규칙에 따라 제외 또는 변환하는 방법
	- 직접 검사	- 작업자가 직접 눈으로 확인하여 검사하는 방법
정제 기준	- 정제 기준 수립	- 데이터 구축 목적, 데이터 유형, 도메인 특성에 따른 데이터 정제기준을 수립
	- 촬영 이미지 데이터	- 촬영 이미지 데이터의 정제 기준으로는 촬영 장비,

		이미지 크기, 비율, 화질 등의 요소
	- 스캔 이미지 데이터	- 스캔 품질, 노이즈 여부 등이 있으며, 어떤 기준으로 정제하는지 정의
	- 텍스트 데이터	- 텍스트 분량, 텍스트 문법의 정확성, 텍스트 내용의 적절성, 획득주제와의 연관성 등을 고려하여 부적절한 데이터를 필터링하거나 라벨링하기 적합한 형태 및 내용으로 수정

- 데이터 획득 및 정제 시, 개인정보사항 고려 및 관련 법규사항을 확인하여 이를 반영하여 획득 및 정제를 할 수 있도록 해야함

### 3. 데이터 라벨링(Data Labeling) 및 어노테이션(Annotation) 방식

#### 가. 데이터 라벨링(Data Labeling) 및 어노테이션(Annotation) 방식 개요

데이터 라벨링	어노테이션
- 인공지능이 기계학습에 활용할 수 있도록 기능이나 목적에 부합하는 정보를 원천데이터에 부착하는 활동	- 데이터 라벨링 시 원천데이터에 주석을 표시하는 작업을 의미하며, 추가 부착되는 설명정보 데이터는 기능 목적에 따라 다양한 형태로 표현될 수 있으며 이러한 설명정보 표현방식을 지칭하는 용어 ※ 용어사용 예 : 사물 바운딩박스 어노테이션, 클래스 라벨링 어노테이션 등

- 어노테이션 작업한 형태의 집합을 데이터 라벨링이라고 함

#### 나. 데이터 라벨링(Data Labeling) 및 어노테이션(Annotation) 방식

데이터 유형	라벨링 기능	어노테이션 방식
텍스트	- 텍스트 분류(Text Classification)	- 클래스 라벨(단일, 다중)
	- 개체명 인식(Named Entity Recognition)	- 단어(구문) 라벨
	- 관계-의존성 정의(Relation-Dependencies)	- 단어(구문) 라벨링 및 두 단어 사이의 관계
이미지	- 이미지 분류(Image Classification)	- 클래스 라벨(단일, 다중)
	- 객체 인식(Object Recognition)	- 바운딩 박스(사각형) - 폴리곤(다각형)
동영상	- 동영상 분류(Video Classification)	- 클래스 라벨(단일, 다중)
	- 객체 인식(Object Recognition)	- 바운딩 박스(사각형)
	- 객체 추적(Object Tracking)	- 키 포인트(정점) - 폴리곤(다각형) - 폴리라인(선)
오디오	- 오디오 분류(Audio Classification)	- 클래스 라벨
	- 오디오 세그멘테이션(Audio Segmentation)	
	- 음성인식(음성 > 텍스트 변환) (Speech to Text)	- 텍스트 전사

기타	- 시계열 세그멘테이션(Time-Series Segmentation)	- 클래스 라벨
	- HTML 문서 분류(HTML Classification)	

- 데이터 라벨링에 포함하지 않아야 할 개인정보 등을 필터링하는 정제 기준을 마련하여 데이터 정제 때문부터 고려해야함

"끝"



#### 기출풀이 의견

6. 인공지능 학습용 데이터 구축 사업은 디지털 뉴딜의 핵심사업으로 향후에도 지속적으로 출제가능성이 있기 때문에 음성, 영상, 텍스트 등 수집기준 및 정제기준에 대한 숙지가 필요합니다.