

ICT의 가치를 이끄는 사람들!!

128회

## 컴퓨터시스템응용기술사 기출풀이 3교시

## 국가기술자격 기술사 시험문제

정보처리기술사 제 128 회

제 3 교시

분야	정보통신	종목	컴퓨터시스템응용	수험 번호		성명	
----	------	----	----------	-------	--	----	--

※ 다음 문제 중 4 문제를 선택하여 설명하시오. (각 25 점)

1. 계층구조의 통신 프로토콜 설계 시 고려해야 할 다음 각 요소에 대하여 설명하시오.  
 가. 오류 제어  
 나. 흐름 제어  
 다. 데이터 전달 방식
2. TCP(Transmission Control Protocol)에 대하여 다음을 설명하시오.  
 가. TCP 헤더(header)  
 나. TCP 헤더에 포함된 제어 플래그(Control Flag)의 종류
3. 디지털 포렌식(forensic) 과정에서 실제로 증거를 수집하려면 네트워크, 시스템, 데이터베이스 및 응용 프로그램 등 다양한 분야에 관한 지식이 필요하다. 다음 각 분야에서 디지털 포렌식의 증거를 수집하기 위한 방법을 설명하시오.  
 가. 네트워크 증거 수집  
 나. 시스템 증거 수집  
 다. 응용 프로그램 증거 수집
4. 클라우드 서비스 제공자는 AIaaS(AI as a Service)를 통해 인공지능을 쉽게 이용할 수 있도록 다양한 API(Application Program Interface)를 제공한다. 다음 항목을 설명하시오.  
 가. AIaaS 개념 및 특징  
 나. AIaaS 구조  
 다. 아마존 웹서비스(AWS)와 렉스(Lex)
5. KS X ISO/IEC/IEEE 29119 을 활용하여 다음을 설명하시오.  
 가. 테스트 설계 및 구현 프로세스  
 나. 테스트 설계 기법  
 다. ISO/IEC 25010 에서 정의한 테스트 품질특성과 명세기반 테스트 설계 기법 간의 매핑
6. 관계 데이터 모델이 가지고 있는 개체 무결성(Entity Integrity)과 참조 무결성(Referential Integrity)에 대하여 설명하시오.

**1. 계층구조의 통신 프로토콜 설계 시 고려해야 할 다음 각 요소에 대하여 설명하시오.**

- 문제  
제  
가. 오류 제어  
나. 흐름 제어  
다. 데이터 전달 방식

출제영역	네트워크	난이도	★★☆☆☆
출제배경	- 가상화 NW 증가로 인한 OSI 7 Layer 기본 원리 및 주요 기능의 중요성 증가		
출제빈도	- 응용 125회 4교시, 응용 120회 3교시 출제		
참고자료	<ul style="list-style-type: none"> <li>- 네트워크 흐름, 혼잡, 오류 제어 기법 (VictoryWoo)</li> <li>- 데이터 전송 방식(안경잡이 이야기)</li> </ul>		
Keyword	<ul style="list-style-type: none"> <li>- 네트워크, TCP/UDP, 오류(검출, 재전송, 수정, FEC, BEC), 흐름(전송속도차 관리, Stop&amp;Wait, Sliding Window), 전달방식(Point-to-Point, Broadcast, Unicast, Multicast)</li> </ul>		
풀이	이정현(125회 정보관리기술사)		

**1. OSI 7 Layer 2, 4 계층 핵심기술, 오류제어의 개념 및 유형**

가. 오류제어의 개념

개념	<ul style="list-style-type: none"> <li>- 데이터링크, 네트워크 계층에서 신뢰성 있는 데이터 전갈을 위해 오류를 검출, 재전송, 수정하는 제어기법</li> </ul>		
특징	<b>1) FEC(전진오류제어)</b> <ul style="list-style-type: none"> <li>- 송신측 잉여비트 삽입, 수신측 오류 정정</li> <li>- 데이터링크에서만 사용</li> </ul>	<b>2) BEC(후진오류제어)</b> <ul style="list-style-type: none"> <li>- 수신 검출, 송신측 재전송 요청</li> <li>- 데이터링크, 네트워크 계층에서 사용</li> </ul>	
	<b>3) 발생원인</b> <ul style="list-style-type: none"> <li>- 감쇄, 자연왜곡, 잡음</li> </ul>	<b>4) 오류 정정 주체</b> <ul style="list-style-type: none"> <li>- FEC 수신측</li> <li>- BEC 송신측</li> </ul>	
유형	<pre> graph TD     EC[오류제어(Error)] --&gt; FEC[전진 오류 수정(FEC)]     EC --&gt; BEC[후진 오류 수정(BEC)]     FEC --&gt; B1[비블록 코드]     FEC --&gt; T1[Turbo 코드]     FEC --&gt; C1[콘볼루션 코드]     BEC --&gt; D1[검출방식]     BEC --&gt; P1[패리티 검사]     BEC --&gt; CS1[체크섬 검사]     BEC --&gt; SR1[순환 중복 검사]     BEC --&gt; WF1[수정방식]     BEC --&gt; ARQ1[정지-대기 ARQ]     BEC --&gt; GBN1[Go-Back-N]     BEC --&gt; SRP1[Selective-Repeat]     BEC --&gt; ARAQ1[적응형 ARQ]     BEC --&gt; HARQ1[H-ARQ]   </pre>		

- 오류제어는 데이터 신뢰성을 위해 매우 중요하며, 오류제어기법은 FEC, BEC로 유형을 분류 가능함.

## 나. 오류제어의 유형

유형	세부유형	설명
FEC	- 블록(비) 코드	- 블록코드는 해밍코드, 상승코드, 비블롯코드는 콘볼루션코드, Turbo 코드를 지칭하는 기법
	- Turbo 코드	- 길쌈부호 중 쉽게 부호화 할 수 있는 것을 조합한 기법
	- 콘볼루션 코드	- 현재-과거 신호간의 상관관계 이용 특히 메모리, 디코딩, 해밍거리에서 활용할 기법
	- BCH 코드	- 선형 선회 블록 부호이며 생성 다항식에 의해 정의된 기법
	- 해밍 코드	- 데이터와 패리티가 비트 관계 즉 1 비트 수정, 2 비트 검출 가능한 기법
BEC	- 패리티 검사	- 패리티 비트 활용, 코드워드 전송 중 간섭의 의한 신호, 타이밍 변경 등의 오류 검출 기법
	- 채크섬 검사	- 송신측 전송 데이터 16bit로 구분후 1의 보수 합을 전송시 수신측에서 오류를 검출하는 기법
	- 순환 중복 검사(CRC)	- 가환대수학 Modulo-2 연산 기반으로 오류를 검출하는 기법
	- Go-Back-N	- 에러발생시 NAK 수신 이후 프레임부터 전송하는 기법
	- Selective-Repeat	- 각각 프레임에 대한 수신확인 후 에러발생 프레임만 재전송 하는 기법
	- 적응형 ARQ	- 전송효율을 고려하려, 데이터 오류 발생확률에 따라 프레임 길이를 동적변환하는 기법
	- H-ARQ	- FEC와 ARQ를 결합하여 전송효율과 재전송횟수를 감소시킨 복합 재전송 기법

- 오류제어도 중요하지만 데이터의 원활한 통신을 위해 흐름제어가 필요.

## 2. 흐름제어의 개념 및 유형

## 가. 흐름제어의 개념

개념	- 송신측 데이터 생산율, 수신측 소비율 차이, 패킷 손실 및 과도 수신 방지를 위한 전송속도 제어 기법	
유형	<pre> graph TD     Flow[흐름제어(Flow)] --&gt; Stop[Stop-and-wait (ACK)]     Flow --&gt; Window[Sliding Window (Buffer + ACK)]     </pre>	<ul style="list-style-type: none"> <li>- 송신 데이터만 전송</li> <li>- 확인 후 다음 데이터 전송</li> <li>- 버퍼만큼의 데이터를 한 번에 전송</li> </ul>

- 통상 Stop and wait, Sliding Window 방식을 이용하여 흐름제어를 수행.

## 나. 오류제어의 유형

유형	흐름도	설명
Stop and wait		<p><b>(개념)</b> 송신 데이터 전송시, 수신측에서 ACK 반환시 다음 데이터를 전송하는 흐름제어 기법</p> <p><b>(특징)</b></p> <ul style="list-style-type: none"> <li>- ACK, NACK</li> <li>- 가장 단순한 기법이나 전송효율 저하</li> <p><b>(절차)</b></p> <ul style="list-style-type: none"> <li>- 송신측 한번에 하나의 프레임 전송 → 수신측 에러 판단 → 에러시 NAK, 정상시 ACK → 다음데이터 전송</li> </ul> </ul>
Sliding Window		<p><b>(개념)</b> 사전 약속된 크기의 프레임 N 만큼 프레임을 전송하는 흐름제어 기법</p> <p><b>(특징)</b></p> <ul style="list-style-type: none"> <li>- Buffer, ACK, NACK</li> <li>- 수신측 오버헤드 감소, 사전규약 필요</li> <p><b>(절차)</b></p> <ul style="list-style-type: none"> <li>- 송/수신측 Buffer 지정 → 송신측 프레임 N 만큼 전송 수신측 에러 판단 → 에러시 NAK, 정상시 ACK → 다음데이터 전송</li> </ul> </ul>

- 흐름제어후 데이터링크 계층일 경우 회선제어를 수행, 네트워크 계층은 분할/재조합, 연결/혼잡제어를 수행
- 오류 제어와 흐름 제어에서 송/수신자간에 신뢰성이 있는 통신을 위해 데이터 전달 방식에 대한 고려 필요

## 3. 데이터 전달 방식의 개념 및 유형

## 가. 데이터 전달 방식의 개념

개념	<p><b>(교환)</b> 전달 경로가 둘 이상일 때 데이터를 어디로 전달할지 결정하는 기능</p> <p><b>(전송)</b> 상호연결된 시스템간의 신뢰성이 있는 데이터 전송을 보장하기 위한 기능</p>		
유형	<pre> graph TD     A[데이터 전달 방식 (Send)] --&gt; B[호스트 기반]     A --&gt; C[네트워크 기반]     A --&gt; D[지리적 기반]     </pre> <ul style="list-style-type: none"> <li>- 스타형</li> <li>- 트리형</li> <li>- 링형</li> <li>- 완전형</li> <li>- 불규칙형</li> <li>- 브로드캐스트</li> <li>- 유니캐스트</li> <li>- 멀티캐스트</li> <li>- LAN</li> <li>- MAN</li> <li>- WAN</li> </ul>		

- 크게 지리적, 호스트(토플로지), 네트워크 기반으로 유형으로 분류 가능

## 나. 데이터 전달 방식의 유형

구분	세부유형	설명
호스트 기반 (Point-to-Point)	- 스타형	- 중앙에 있는 하나의 중개 호스트 주위로 여러 호스트를 일대일로 연결하는 방식
	- 트리형	- 스타형의 다단계 확장형, 주변호스트가 중개호스트로 확장되는 과정을 반복해 네트워크를 연결하는 방식
	- 링형	- 호스트의 연결이 순환 고리를 갖고, 전송데이터를 브로드캐스팅하는 방식
	- 완전형	- 네트워크내 모든 호스트를 1:1로 연결하는 방식
	- 불규칙형	- 네트워크 연결구조를 분류할 수 없는 구조
네트워크 기반 (Cast)	- 브로드캐스팅	- 특정 호스트가 데이터 전송시 네트워크에 연결된 모든 호스트에게 데이터가 전송되는 방식
	- 유니캐스팅	- 송신 호스트가 한 번의 전송으로 수신 호스트 하나에만 데이터를 전송하는 방식
	- 멀티캐스팅	- 송신 호스트가 한 번의 전송으로 다수의 수신 호스트에 전송하는 방식
지리적 기반	- LAN	- 근거리 통신망, TCP/IP, 허브, 사무실, 병원, 학교
	- MAN	- 도시 규모 통신망, ISP 역할, 회선 제공 회사
	- WAN	- 원거리 통신망, 국가간 통신, 스위치/점대점방식

- TCP/I의 주요 기능이며, NW 가상화 기술인 SDN, NFV의 기업도입이 증가함에 따라 기본 제어기능인 오류제어, 흐름제어, 전송방식의 매커니즘과 원리에 대한 심화학습이 필요.

"끝"

## 기출풀이 의견

- 쉬워보이지만, 막상 접근하려니 쉽지 않은 문제. 저의 경우 개념/유형으로 접근했지만, 세부적인 기법 2~3개를 심도있게 작성하는 것이 높은 점수를 받을 것으로 예상됩니다. 또한 NW도메인이므로 도식화와 키워드 도출에 신경쓰시면 좋은 점수 기대합니다.

## 2. TCP(Transmission Control Protocol)에 대하여 다음을 설명하시오.

문제 제 가. TCP 헤더(header)

나. TCP 헤더에 포함된 제어 플래그(Control Flag)의 종류

출제영역	네트워크	난이도	★★☆☆☆
출제배경	- 가상화 NW 증가로 인한 OSI 7 Layer 기본 원리 및 주요 기능의 중요성 증가		
출제빈도	- 응용 125회 4교시, 응용 120회 3교시 출제		
참고자료	- TCP Header, Control Flag (정보통신기술용어해설)		
Keyword	<ul style="list-style-type: none"> <li>- TCP : 연결지향, 신뢰성, 순서보장, Stream, Segment</li> <li>- 헤더 : Source Port, Destination Port, Sequence-number, Ack-number, HLEN, Reserved, Control Flag, Windows Size, Checksum, Urgent Pointer</li> <li>- 제어 플래그 : URG, ACK, PSH, RST, SYN, FIN</li> </ul>		
풀이	이정현(125회 정보관리기술사)		

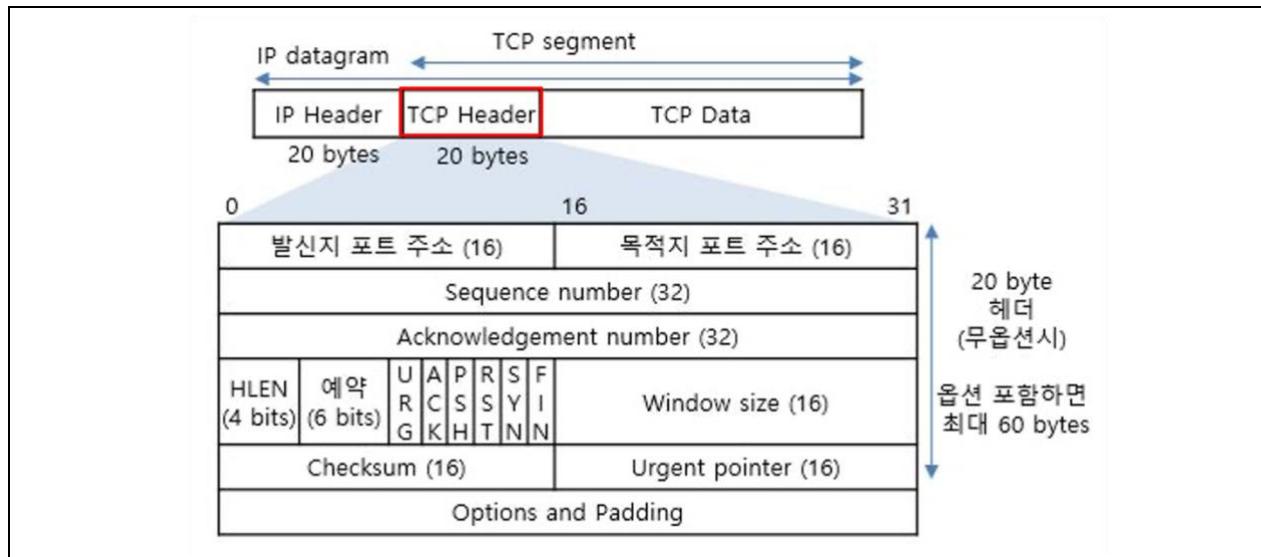
## 1. OSI 7 Layer 4, 전송 계층 핵심 프로토콜 TCP 의 개요

TCP 프로토콜 위치	OSI	TCP/IP	TCP/IP Protocol Suite	
	Application Presentation Session	Application	HTTP FTP SMTP DNS RIP SNMP	
	Transport	Transport	TCP UDP	
	Network	Internet	ARP IP IGMP ICMP	
	Data-Link	Link	Ethernet Token Ring Frame Relay ATM	
	Physical	Physical		
개념	- OSI 7 Layer 의 전송 계층에 해당하며 양 종단 호스트 내 프로세스 상호 간에 신뢰적인 연결 지향성 서비스를 제공하는 프로토콜			
특징	<b>1. 연결지향 프로토콜</b> <ul style="list-style-type: none"> <li>- 가상회선방식</li> <li>- 스트림 기반 전송방식</li> <li>- Unicast 방식, UDP 대비 저속</li> </ul>		<b>2. 신뢰가능 프로토콜</b> <ul style="list-style-type: none"> <li>- 분할/재조합</li> <li>- 연결제어, 흐름제어</li> <li>- 오류제어, 혼잡제어</li> </ul>	

- 특히 TCP 프로토콜은 UDP 대비 Stack 면에서 큰차이를 보이며, 제어 플래그(Control frag)가 존재

## 2. TCP 헤더

## 가. TCP 헤더 구조도



- TCP Header는 모든 패킷이 수신자에게 정상적으로 전송됨을 확인하기 위해 복잡한 구조를 지님.

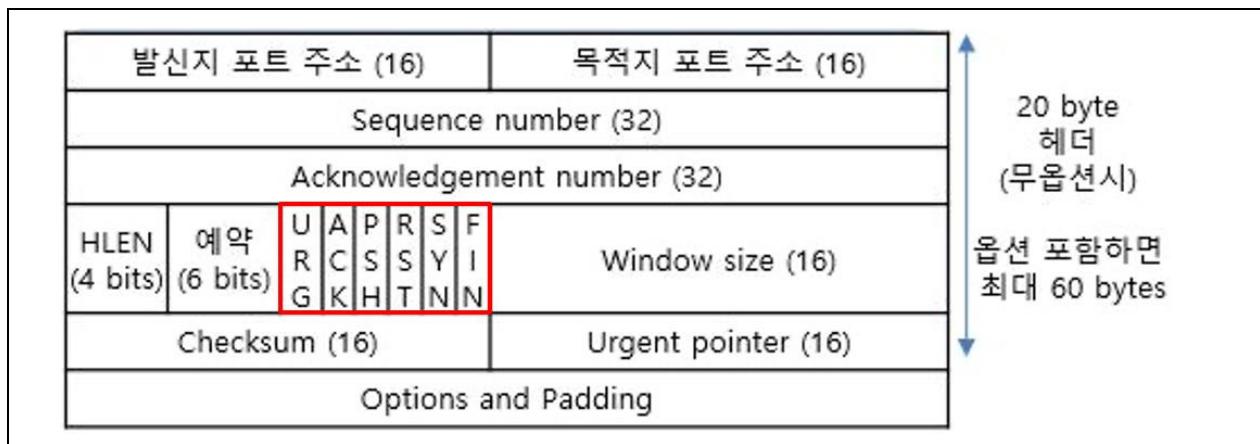
## 나. TCP 헤더 상세 설명

구분	비트	설명
<b>발신지 포트 주소 (Source)</b>	16bit	<ul style="list-style-type: none"> <li>- 출발지(송신) 포트 번호(IP 주소 + 포트번호 = 소켓 주소)</li> <li>- 양쪽 호스트 내 종단 프로세스 식별 필드</li> </ul>
<b>목적지 포트 주소 (Destination)</b>	16bit	<ul style="list-style-type: none"> <li>- 목적지(수신) 포트 번호(IP 주소 + 포트번호 = 소켓 주소)</li> <li>- 양쪽 호스트 내 종단 프로세스 식별 필드</li> </ul>
<b>Sequence number</b>	32bit	<ul style="list-style-type: none"> <li>- 송신 데이터 순서 번호, 데이터 시작 순번 입</li> <li>- 0이 아닌 임의의 수 할당</li> </ul>
<b>Ack number</b>	32bit	<ul style="list-style-type: none"> <li>- 상대방이 다음에 전송할 순서 번호를 입력</li> <li>- ACK 와 함께 해당 필드에 상대방 전송 번호를 입력</li> </ul>
<b>HLEN</b>	4bit	<ul style="list-style-type: none"> <li>- 헤더길이, 4bit 워드 단위로 표시</li> <li>- 기본헤더 20byte 와 옵션 헤더 포함 시 최대 40byte 로 구성</li> </ul>
<b>Reserved</b>	6bit	<ul style="list-style-type: none"> <li>- 예약 필드</li> <li>- 현재 사용하지 않는 필드</li> </ul>
<b>Control Flags</b>	6bit	<ul style="list-style-type: none"> <li>- URG(긴급), ACK(응답), PSH(버퍼 데이터), RST(강제종료), SYN(연결), FIN(연결종료)로 구성</li> </ul>
<b>Window size</b>	16bit	<ul style="list-style-type: none"> <li>- 수신 버퍼의 여유 공간</li> <li>- Sliding Window 제어 방식</li> </ul>
<b>Checksum</b>	16bit	<ul style="list-style-type: none"> <li>- 헤더를 포함한 전체 세그먼트에 대한 오류를 검사하는 필드</li> <li>- 검사합, CRC, BlockSum 등</li> </ul>
<b>Urgent Pointer</b>	16bit	<ul style="list-style-type: none"> <li>- 세그먼트가 USG 플래그일 경우 사용하는 필드</li> <li>- 긴급데이터의 위치값이 할당된 필드</li> </ul>
<b>Options</b>	20 ~ 40byte	<ul style="list-style-type: none"> <li>- TCP MSS 옵션 협상, 윈도우 제어 제공</li> <li>- 선택확인응답 기능, 타임스탬프 옵션 정의 제공</li> </ul>

- TCP header에서 연결회선 제어와 데이터를 관리하는 제어 플래그(Control Flag)는 매우 중요.

### 3. TCP 제어플래그

#### 가. TCP 제어플래그 구조도



- 총 6 개의 플래그로 구성되며, TCP 연결회선 제어 및 데이터 관리를 위해 사용

#### 나. TCP 헤더 제어플래그 상세설명

구분	주요역할	설명
<b>URG (Urgent)</b>	- 긴급 데이터 할당	- 송신측 상위 계층이 긴급 데이터 확인시 URG 를 1 로 설정하고 순서에 상관없이 우선 전송
<b>ACK (Acknowledgment)</b>	- 확인응답 제공	- Acknowledgement number 가 할당되었음을 알림 - 1 확인번호 유효, 0 확인번호 미포함, SYN 전송시 1로 셋팅
<b>PSH (Push)</b>	- 버퍼 데이터 공간	- 버퍼링된 데이터를 상위 계층으로 즉시 전달 - 수신측 버퍼가 찰 때까지 기다리지 않고 즉시 전달
<b>RST (Reset)</b>	- 강제연결 초기화	- 연결확립(Established)된 회선에 강제 리셋 요청(RST=1로 설정) - LISTEN, SYN_RCV 일 경우 LISTEN / 그외 상태 일 경우 CLOSED
<b>SYN (Synchronize)</b>	- 연결 및 회선개설	- TCP 연결설정 초기화를 위한 순서번호의 동기화 - 연결요청(SYN1, ACK0), 연결허락(SYN1, ACK1), 연결설정(ACK1)
<b>FIN (Finish)</b>	- 연결해제 및 종료	- 송신기가 데이터 보내기를 끝마침 - 종결요청(FIN1), 종결응답(FIN1, ACK1)

- OSI 7 Layer 와 TCP/IP 전송계층에서 목적은 동일하나 반대성향을 지닌 UDP 프로토콜이 존재

### 4. TCP 와 UDP 비교

비교항목	TCP	UDP
연결방식	연결형 프로토콜	비연결형 프로토콜
패킷 교환 방식	가상 회선 방식	데이터그램 방식
전송 순서	전송 순서 보장	전송 순서 변경 불가
수신 여부 확인	수신 여부 확인	수신 여부 확인 불가능
통신 방식	1 : 1 통신	1 : 1, 1 : N, N : N 통신
신뢰성	높음	낮음
속도	저속	고속

- 최근 TCP 의 HOL(Head Of Line) 문제로 UDP + TLS 를 결합한 QUIC 을 적용하고 있는 상황

"끝"



### 기출풀이 의견

2. NW 도메인이며, 기본토pic으로, 철저히 기본기를 연습하셨다면 고득점 예상합니다. 특히 OSI 7 Layer와 TCP/IP 모델은 빈출 토픽이며, 최근 계속 출제되고 있습니다. 이를 고려해, 평소에 준비해 주시면 좋겠습니다.

문제 제 3. 디지털 포렌식(forensic) 과정에서 실제로 증거를 수집하려면

네트워크, 시스템, 데이터베이스 및 응용 프로그램 등 다양한 분야에  
관한 지식이 필요하다. 다음 각 분야에서 디지털 포렌식의 증거를

수집하기 위한 방법을 설명하시오.

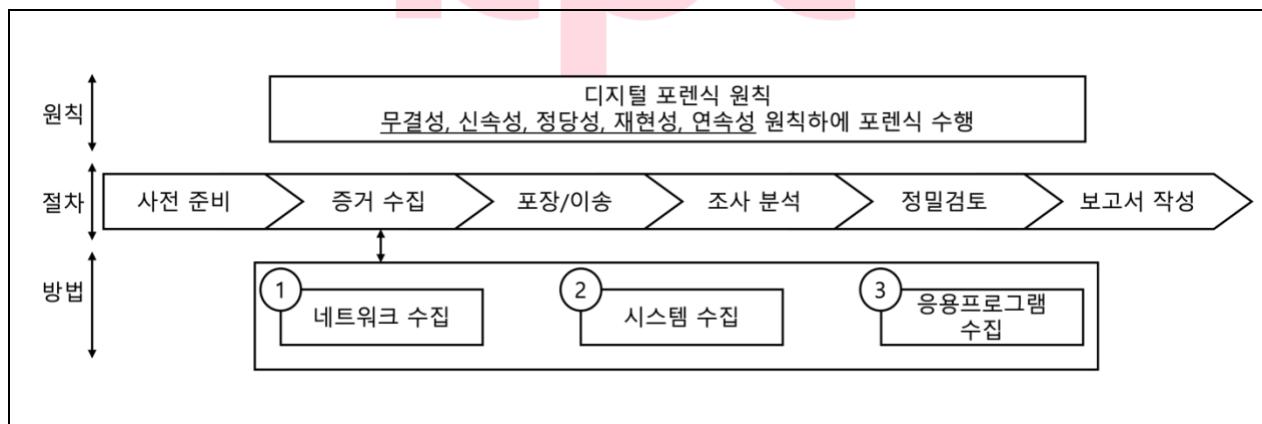
가. 네트워크 증거 수집

나. 시스템 증거 수집

다. 응용 프로그램 증거 수집

출 제 영 역	보안	난 이 도	★★☆☆☆
출 제 배 경	- 디지털 데이터의 법적 증거물 활용 증가		
출 제 빈 도	<ul style="list-style-type: none"> <li>- 118 회 관리 4 교시, 117 회 관리 4 교시, 응용 첫 출제</li> <li>- 디지털 포렌식(도리의 디지털라이프)</li> </ul>		
참 고 자 료	<ul style="list-style-type: none"> <li>- 디지털 포렌식(HM Company)</li> <li>- 디지털 포렌식(도리의 디지털라이프)</li> </ul>		
Key word	<ul style="list-style-type: none"> <li>- 네트워크 : 라우터, IPS, IDS, WAS/WEB Server</li> <li>- 시스템 : 운영체계, 응용프로그램, 프로세스, 디스크 포맷 타입</li> <li>- 응용프로그램 : 웹브라우저, 캐시, 쿠키, 다운로드내역, 인스턴트 메신저, 이메일</li> </ul>		
풀 이	이정현(125 회 정보관리기술사)		

## 1. 디지털 포렌식의 개요



- 사이버 범죄 행위에 대한 증거 데이터를 수집하여 법적 증거물로 활용하는 방법 및 범죄 수사 기술
- 특히 네트워크, 시스템, 응용프로그램 내 정보를 수집

## 2. 네트워크 증거 수집

증거 수집 방법	1 라우터	2 보안장치	3 서버
	<ul style="list-style-type: none"> <li>- 휘발성 정보 수집</li> <li>- 메모리 덤프</li> </ul>	<ul style="list-style-type: none"> <li>- IDS</li> <li>- IPS</li> </ul>	<ul style="list-style-type: none"> <li>- Web Server</li> <li>- Was Server</li> </ul>

구분	방법	설명
라우터	- 휘발성 정보 수집	- 라우터의 연결정보, 트래픽, 세션등의 정보 수집 - router# show clock/version/users/access-lists/ip/arp
	- 라우터 메모리 덤프	- TFTP, FTP, RCP, Flash Disk 을 이용해 메모리 복제 - router# exception dump TFTP, FTP, RCP, Flash Disk
보안장치	- IDS	- 탐지 중심으로 정해진 룰 기반으로 트래픽 모니터링 - snort 기준 /var/adm/snort/logs
	- IPS	- 탐지/차단 중심으로 정해진 룰 기반 트래픽 모니터링 - snort 기준 /var/adm/snort/logs
서버	- Web Server	- CentOS 아파치 서버 기준, /var/log/httpd 안에 위치 - tail -f /var/log/httpd/access_log
	- Web Application Server	- CentOS 아파치 서버 기준, /var/lib/tomcat/logs - tail -f /var/lib/tomcat/logs/Catalina.out (초기설정시)

- 네트워크 증거 수집은 패킷, 트래픽의 송신지, 목적지를 판별하거나 를 위반사례를 수집

### 3. 시스템 증거 수집

증거 수집 방법	1 운영체계	2 프로세스	3 디스크
유형	세부유형	설명	
운영체계	- Windows	<ul style="list-style-type: none"> <li>- 보안 및 설정 정보가 담긴 윈도우 레지스트리 수집</li> <li>- 레지스트리 SAM, SECURITY 에 저장된 계정정보 수집</li> </ul>	
	- Unix / Linux	<ul style="list-style-type: none"> <li>- 시스템로그, 계정정보, 권한체계 정보 수집</li> <li>- 로그위치(/var/log), 계정정보(/etc/passwd) 수집</li> </ul>	
프로세스	- 프로세스 정보 수집	<ul style="list-style-type: none"> <li>- 실행 프로세스 목록, 프로세스 사용률, 프로세스 모듈</li> <li>- pslist, Listdlls, handle 도구를 사용하여 수집</li> </ul>	
	- 서비스 정보 수집	<ul style="list-style-type: none"> <li>- 서비스 목록 및 SVC, AVCTP 서비스 정보 수집</li> <li>- net stat, psservice 통해 실행/등록된 서비스 정보 수집</li> </ul>	
디스크	- 파일	<ul style="list-style-type: none"> <li>- 특정 시간대 사용된 파일 복구 및 수집</li> <li>- Read, Modify, Access 내역을 통해 무결성 파악</li> </ul>	
	- 저장매체	<ul style="list-style-type: none"> <li>- 하드디스크, SSD, USB 에 저장된 정보 수집</li> <li>- GMD, MD-Smart, Data Compass, FTK Imager 이용</li> </ul>	

- 시스템 증거 수집은 모바일에 저장된 데이터도 포함 하므로 넓은 영역을 지닌 수집방법

## 4. 응용프로그램 증거 수집

증거 수집 방법	1 웹브라우저	2 인스턴스 메신저	3 이메일
	- 캐시 / 쿠키 - 패킷	- 대화내용 - 다운로드 내역	- PST 파일 - OST 파일
구분	세부유형	설명	
웹브라우저	- Internet Explorer	<ul style="list-style-type: none"> <li>- 임시파일, 쿠키, 히스토리 수집(버전별 일부상이)</li> <li>- 히스토리내 URL, 다운로드 기록, 마지막 방문시간</li> </ul>	
	- Chrome	<ul style="list-style-type: none"> <li>- SQLite 포맷으로 저장하고 수집정보는 IE와 동일</li> <li>- 인터넷 접속 기록, Cache 파일, 자주 접속한 웹 정보</li> </ul>	
인스턴스 메신저	- 카카오톡	<ul style="list-style-type: none"> <li>- 로그, 해시, 파일복구, 임시파일(액셀) 복구 증거 수집</li> <li>- Jailbreak, Dump, FTK imger 를 이용하여 증거 수집</li> </ul>	
	- 네이트온	<ul style="list-style-type: none"> <li>- 윈도우 기준 C:\Windows 사용자계정/appdata/local 수집</li> <li>- 숨김폴더해제 후 대화내용 수집</li> </ul>	
이메일	- Outlook	<ul style="list-style-type: none"> <li>- 개인 폴더 파일(pst), 오프라인 폴더 파일(ost) 수집</li> <li>- 삭제된 경우 지운 편지함 내 OST 통합 확인을 수집</li> </ul>	
	- Mozilla Thunderbird	<ul style="list-style-type: none"> <li>- 메일 요약 인덱스 파일(msf) 수집후 Outloo 으로 연동</li> <li>- 윈도우, 리눅스, OS X 호환으로 메일기록 수집 용이</li> </ul>	

- 증거를 훼손을 위해 삭제, 은닉, 조작, 풋프린트 최소화 기술을 이용해 훼손하므로 안티포렌식 대응기술이 필요

## 5. 안티포렌식 대응기술

구분	대응기술	설명
탐지	- 데이터 탐지 - 스테가노그래피 탐지	- 인덱스 파일탐지, Bitwise-Slack Sapce 탐지 - 다른 파일에 숨긴 기밀정보를 탐지
복구	- 디스크 복구 - 패스워드 복원	- 데이터카빙, 시그니처/엔트로피분석 통해 정보 복구 - Brute Force, Dictionary 공격에 손상된 정보 복구

- 디지털 포렌식은 수사기관, 기업체, E-discovery 등 다양한 영역으로 확장

"끝"

## 기출풀이 의견

3. 디지털포렌식 원칙, 절차, 방법에 대해 큰그림을 보여주고, 각 수집방법에 대해 서술하시면 좋은 점수 기대합니다. 다만 보안전문가라는 느낌을 줄 수 있도록 표현하는 것이 중요합니다.

## 문제

4. 클라우드 서비스 제공자는 AlaaS(AI as a Service)를 통해 인공지능을 쉽게 이용 할 수 있도록 다양한 API(Application Program Interface)를 제공한다. 다음 항목을 설명하시오.

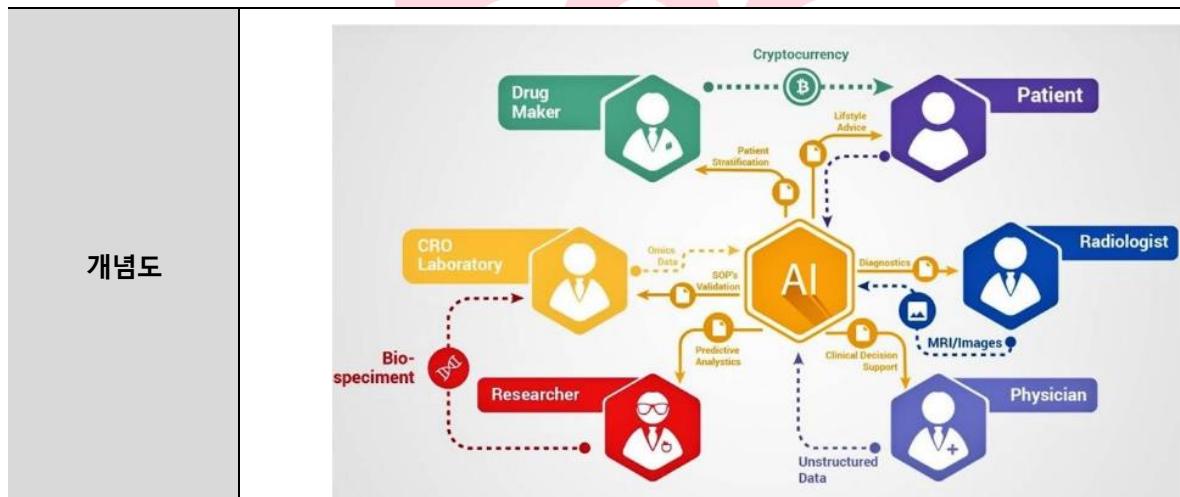
가. AlaaS 개념 및 특징

나. AlaaS 구조

다. 아마존 웹서비스(AWS)와 렉스(Lex)

출제 영역	디지털서비스	난이도	★★☆☆☆
출제 배경	- AI의 대중화 및 지능형 인프라 구성 및 챗봇 활용		
출제 빈도	- 미출제		
참고자료	<ul style="list-style-type: none"> <li>- Amazon AWS &amp; Lex (아마존 공식 홈페이지)</li> <li>- AlaaS : 클라우드, 엣지 AI, 학습 및 모델링을 한 AI 엔진, Appl, 프레임워크 제공</li> <li>- 구조 : 인프라, AI 플랫폼, 오토 ML, APIs, AI 서비스</li> <li>- AWS : Personalize(개인 맞춤형 추천 서비스), Forecast(예측서비스), DeepLens(사물탐지 학습)</li> <li>- Lex : 대화형 음성 및 텍스트 I/F 개발, 챗봇</li> </ul>		
Keyword			
풀이	이정현(125 회 정보관리기술사)		

### 1. AlaaS 개념 및 특징

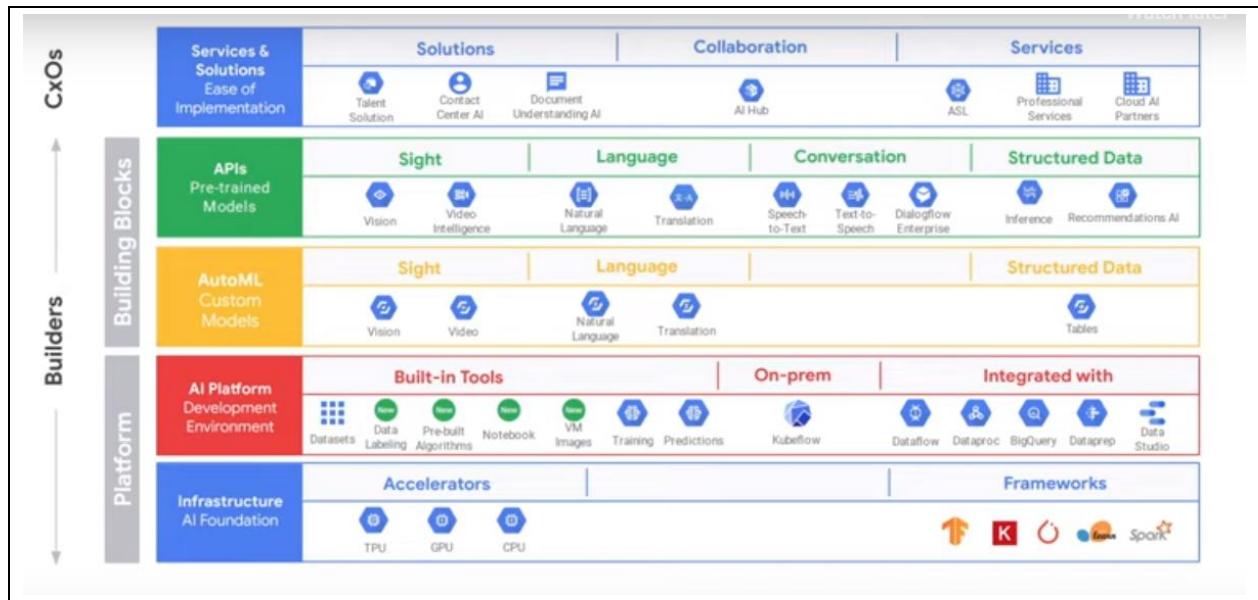


개념도			
개념	AI 학습, 모델링, 데이터 가공/분석 등 AI 사용을 위한 AI 엔진, 어플리케이션, 프레임워크를 제공하는 클라우드 서비스		
특징	1. 봇 / NLP 지원 - 챗봇, 가상 비서, 오토메일링 - 고객 서비스 및 마케팅 특화	2. 인지 컴퓨팅 API - API 제공 통한 빠른 AI 구현 - NLP, 컴퓨터비전, 지능검색, 번역 등	3. 완전 관리형 머신러닝 - AI 모델 구축 프레임워크 제공 - 적은 데이터로 ML 구축 가능
	4. 높은 비용효율 - AI 투자 비용 절감 - AI 전문 지식이 부족해도 AI 구현 활용		

- 클라우드로 제공 되므로 별도의 HW/SW의 제약없이 AI 서비스 구축 가능. 기업환경에 따라 AlaaS 구조 변경.

## 2. AlaaS 구조

### 가. AlaaS 구조도



- AlaaS 는 총 5 개의 영역구분, Pre-trained model, Custom model, Development model 을 선택하여 AI 구축 가능

### 나. AlaaS 구조 설명

구분	구성요소	설명
Services & Solutions	- Solutions	- Talent, Contact Center AI, Document AI 제공 및 해결책 제시
	- Collaboration	- AI Hub 기반으로 AI 모델, 서비스, 데이터 수집등 협업툴 제공
	- Services	- ASL(Advanced Solutions Lab), 비즈니스 AI 모델 기술지원
APIs (학습 필요 없음)	- Sight/Language	- 컴퓨터비전, NLP, CNN 통한 AI 모델 학습 지원도구 제공
	- Conversation	- Speech-to-Text, Dialogflow 등 음성 및 텍스트 변환도구 제공
	- Structured Data	- Recommendations AI로 개인제품 추천서비스 구축
Custom Models (AutoML)	- Sight	- Vision, Video 등 AutoML에서 제공된 데이터 셋 활용
	- Language	- NLP, Translation 등 AI 모델 학습을 위한 도구 지원
	- Structured Data	- Google Tables 통해 실시간 협업 DB 생성 및 활용
AI Platform (사용자 지정 모델)	- Built-in Tools	- 아나콘다, Notebook, Python 등 AI 구현에 필요한 개발도구 제공
	- On-Premise	- 클라우드 기반 AI 구축 및 서비스 설정 기능 제공
	- Integrated	- Dataflow, BigQuery 등 데이터 통합 도구 제공
Infrastructure	- Accelerators	- TPU, GPU, CPU 데이터 분석 및 딥러닝용 NPU 하드웨어 제공
	- Frameworks	- Tensorflow, Kubeflow, Spark, Beam 등 ML 프레임워크 제공

- 특히 아마존은 웹서비스 기반으로 다양한 솔루션을 제공하고, 최근 AI 챗봇인 렉스(Lex)를 발표

## 3. 아마존 웹서비스와 렉스

## 가. 아마존 웹서비스

개념도	<p>AWS Cloud WAN diagram illustrating the architecture:</p> <ul style="list-style-type: none"> <li><b>Network Attachments:</b> Connections and resources included in your network.</li> <li><b>AWS Cloud WAN:</b> Build, manage, and monitor global wide area networks.</li> <li><b>Central Dashboard:</b> Create connections between branch offices, data centers, and Amazon VPCs.</li> <li><b>Network Policy:</b> Defines rules and apply policies to configure and manage your network.</li> <li><b>Unified, centrally managed network:</b> The final outcome.</li> </ul>
개념	<ul style="list-style-type: none"> <li>- 웹서비스에 필요한 컴퓨팅, DB, 저장소, 컨테이너, 웹/모바일, 서비스, AI 등 제품을 제공하는 클라우드 서비스</li> </ul>
주요 서비스	<ul style="list-style-type: none"> <li>- Amazon EC2, S3, Aurora, DynamoDB, RDS, Lambda, VPC, SageMaker 등 서비스 제공</li> </ul>

- 최근 AI 도입 및 챗봇 보편화로 기계학습 서비스에 음성 및 텍스트 기반 챗봇인 렉스(Lex)를 추가

## 나. 렉스

개념도	<p>Amazon Lex diagram illustrating the chatbot building process:</p> <ol style="list-style-type: none"> <li><b>Amazon Lex:</b> Build conversational interfaces into any app using voice &amp; text.</li> <li><b>Define your chat experience:</b> What issue/problem are you solving for and what are the desired outcomes.</li> <li><b>Intent:</b> Desired action <b>Utterance:</b> User input <b>Prompt:</b> Request data <b>Slot:</b> Required data <b>Fulfillment:</b> Completed action</li> <li><b>Build and deploy your chatbot:</b> Test your chatbot before deploying into production.</li> </ol>
개념	<ul style="list-style-type: none"> <li>- 대화형 인터페이스를 애플리케이션에 설계, 구축, 테스트, 배포하기 위해 고급 자연어 모델을 사용하는 완전관리형 AI 챗봇 서비스</li> </ul>
주요 기능	<ul style="list-style-type: none"> <li>- 고품질 음성인식 및 자연어 처리(컨텍스트 기반 사용자 의도 파악, 연속적 대화)</li> <li>- 빌더 생산성(다양한 플랫폼에 배포가능, 스트리밍 지원, 다중 개발자 환경 지원)</li> </ul>

- 금융서비스, 보험, 소매, 통신, 여행등 다양한 분야에서 활용 가능한 챗봇으로 높은 범용성을 지님.

"끝"

## 기출풀이 의견

4. 개인적으로 어려운 문제였습니다. AlaaS에 대한 전반적 지식과 AWS 경험자라면 고득점을 예상합니다.

## 5. KS X ISO/IEC/IEEE 29119 을 활용하여 다음을 설명하시오.

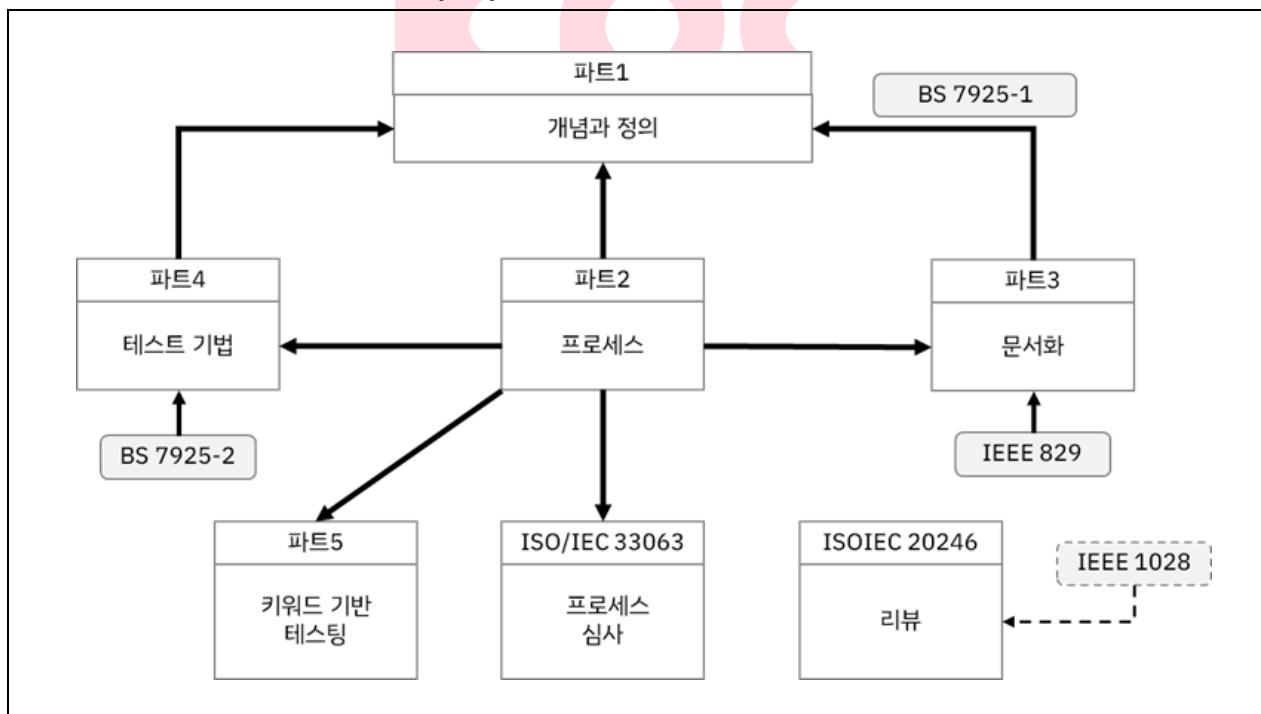
가. 테스트 설계 및 구현 프로세스

문제 제 나. 테스트 설계 기법

다. ISO/IEC 25010 에서 정의한 테스트 품질특성과 명세기반 테스트 설계 기법간의 매핑

출제영역	소프트웨어 공학	난이도	★★☆☆☆
출제배경	- SW 필요성/중요성 증가로 인한 테스트기법 및 프로세스 대두		
출제빈도	- 미출제		
참고자료	<ul style="list-style-type: none"> <li>- ISO/IEC/IEEE 29119 요약 정리본 (악마성)</li> <li>- 소프트웨어 품질의 정의 ISO/IEC 25010 품질특성 (개발 안하는 공대생)</li> </ul>		
Keyword	<ul style="list-style-type: none"> <li>- TTA(한국정보통신기술협회), 과기부(과학기술정보통신부) 주도하 제정된 국내 테스트 표준</li> <li>- 설계 및 구현 프로세스 : 기능세트 식별, 테스트 컨디션 도출, 테스트 커버리지 항목 도출, 테스트케이스 도출, 테스트 세트 구성, 테스트 절차 도출</li> <li>- 설계기법 : 명세기반, 구조기반, 경험기반</li> </ul>		
풀이	이정현(125 회 정보관리기술사)		

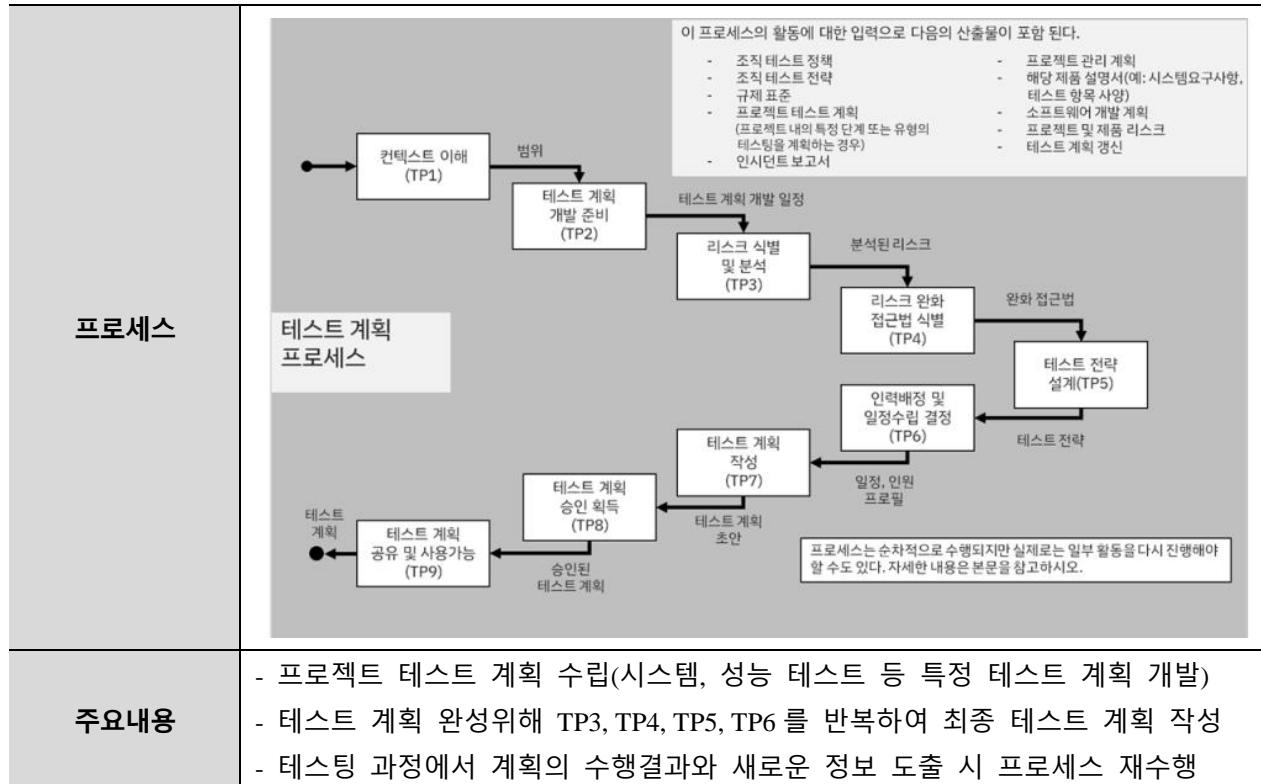
### 1. SW SDLC 테스트 표준 KS X ISO/IEC/IEEE 29119 표준의 개요



- TTA(한국정보통신기술협회)와 과기부(과학기술정보통신부) 주도하에 제정된 SW 테스트 표준
- SW 테스트에 관한 개념/정의, 기법, 프로세스, 문서화, 키워드 기반 테스트, 심사, 리뷰 등 상세한 내용 수록
- 파트 2 테스트 프로세스에서 정의된 내용을 기반으로 테스트 설계 및 구현프로세스에서 사용

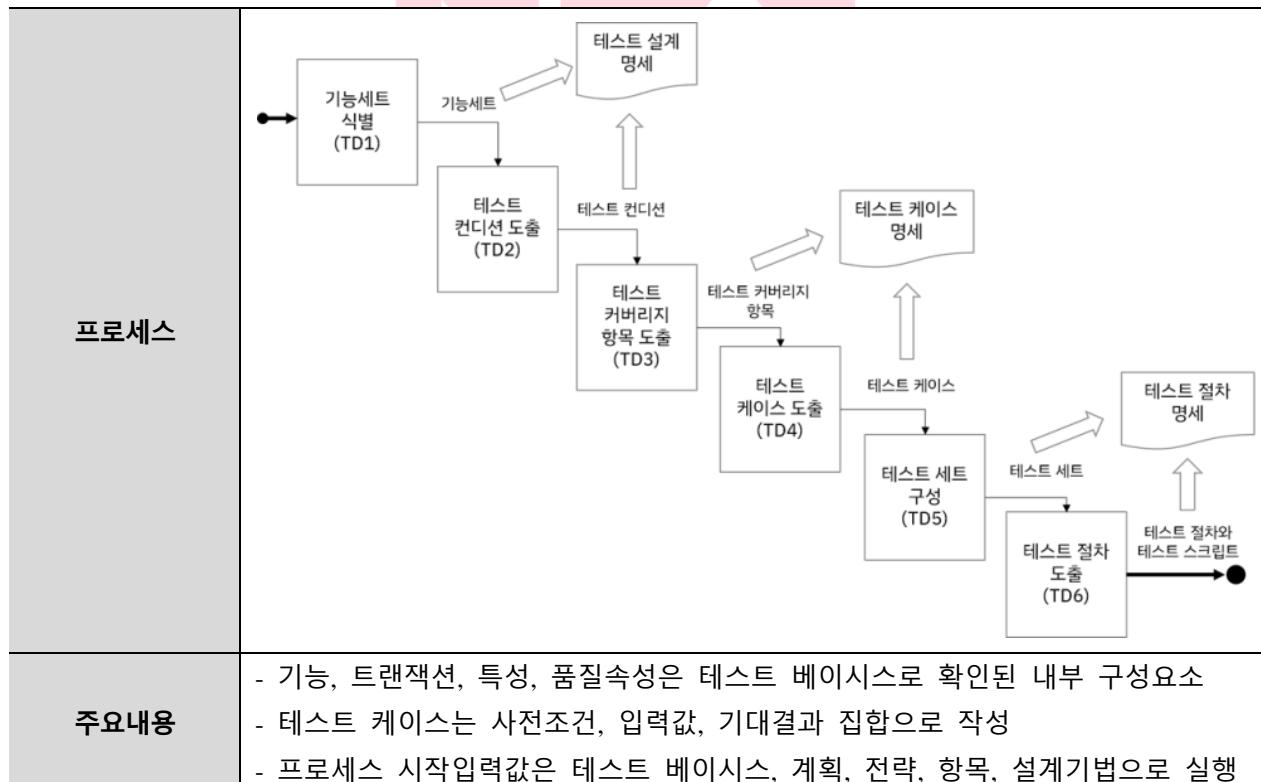
## 2. 테스트 설계 및 구현 프로세스

### 가. 테스트 설계 프로세스



- 테스트 설계 프로세스를 통해 테스트 케이스, 리스크 기반 테스트, 테스트 항목을 도출 후 구현 프로세스 실행

### 나. 테스트 구현 프로세스



- 설계 및 구현 프로세스 사용되는 테스트 설계기법은 명세/구조/경험 기반 테스팅을 활용

### 3. 테스트 설계 기법

#### 가. 테스트 설계 기법 유형



- 명세/경험 기반 테스트는 블랙박스, 구조 기반 테스트는 화이트박스 테스트에 해당

#### 나. 테스트 설계 기법 설명

구분	사전 준비	기법
명세 기반 테스트 설계 기법	<ul style="list-style-type: none"> <li>- 블랙박스 기법</li> <li>- 테스트 베이시스 문서</li> <li>- 요구사항 정의서</li> </ul>	<ul style="list-style-type: none"> <li>- 동등분할, 분류트리, 경계값 분석, 구문규칙, 조합테스트</li> <li>- 결정테이블, 원인결과 그래프, 상태전이, 시나리오, 랜덤</li> <li>- 테스트케이스 기반 검토 (SRS, 아키텍처, 유스케이스)</li> </ul>
구조 기반 테스트 설계 기법	<ul style="list-style-type: none"> <li>- 화이트박스 기법</li> <li>- 컴포넌트 / 코드 분석</li> <li>- 기 개발 테스트 케이스</li> </ul>	<ul style="list-style-type: none"> <li>- 구문, 분기, 결정, 분기조건, 분기조건조합 테스트</li> <li>- 변경조건 결정 커버리지, 데이터 흐름 테스트</li> <li>- 테스트 완료 조건으로 활용, 테스트 충분함 측정</li> </ul>
경험 기반 테스트 설계 기법	<ul style="list-style-type: none"> <li>- 블랙박스 기법</li> <li>- 경험 및 노하우 활용</li> <li>- 비기능 테스트 케이스</li> </ul>	<ul style="list-style-type: none"> <li>- 탐색적 테스트, 오류추정, 체크리스트 테스트</li> <li>- 경험 노하우 기반 분류정리 및 특성테스트 시행</li> <li>- 비기능 테스트 수행 시 주로 활용</li> </ul>

- 특히 명세기반 테스트 설계시 ISO/IEC 25010 정의한 테스트 품질특성을 고려하여 매핑시 초기비용 절감효과

## 4. ISO/IEC 25010 테스트 품질특성과 명세기반 테스트 설계 기법간의 매핑

## 가. ISO/IEC 25010 테스트 품질특성 및 명세기반 테스트 설계 매핑도

품질 특성	품질 부특성	시험 설계 기법															예상 추정
		경계값 분석	원인-결과 그래프	분류 드리 방법	조합 시험 설계	의사결정 표시 시험	동등 분할	랜덤 시험	시나리오 시험	상태 전이 시험	선택스 시험	유스 케이스 시험	분기 시험	갈정 시험	분기 조건 시험	갈정 조건 조합 시험	데이터 흐름 시험
기능 적합성	기능 완전성	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
	기능 정확성	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
	기능 적절성	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●	●
성능 효율성	시간 효율성	●		●													
	자원 사용성																
	수용성	●															
호환성	공존성	●		●	●												
	상호운영성																●
사용성	적합 인지성																
	학술성																
	운영성																
신뢰성	사용자 오류 보호	●	●	●	●	●	●										
	사용자 인터페이스 호감성																
	접근성																
보안성	성숙성							●									
	가용성							●									
	결합 허용성	●						●									●
이식성	복구성							●									
	기밀성	●						●									
	무결성	●						●									
	부인방지							●									
	주적성							●									
	인증성							●									
	적용성							●									●
	설치성							●									●
	대지성							●									●

- 품질 특성 및 부특성별로 수행이 필요한 시험 유형을 선정하고 평가수행 및 평가판정 진행

## 나. SW 품질 평가 판정 요약서

품질특성	품질 부특성	품질지표	결과	점수	가중치(%)	판정
기능 적합성	기능 완전성	기능 구현 커버리지	...	...	...	PASS
	기능 정확성	계산 정확성	....	....	....	PASS
	기능 적정성	기능 적정성	...	...	...	FAIL
...	...	...	....	....	....	

- 평가 요구사항에서 수립된 평가 기준에 따라 개별 품질 속성에 대해 품질 평가

"끝"

## 기출풀이 의견

2. SW테스트와 SW품질평가 표준을 복합하여 출제된 문제입니다. 2개의 표준을 물어봤으므로 정확한 키워드와 논리적 연결성이 중요합니다.

**문제 제 6. 관계 데이터 모델이 가지고 있는 개체 무결성과 참조 무결성(Referential Integrity)에 대하여 설명하시오.**

출제영역	데이터베이스	난이도	★★☆☆☆
출제배경	- 데이터의 중요성 증가, 중복제거 및 무결성을 유지 위해 필요		
출제빈도	- 110회 응용 2교시, 108회 관리 2교시		
참고자료	<ul style="list-style-type: none"> <li>- 데이터 무결성 (대웅제약 기술블로그)</li> <li>- 관계데이터 무결성 제약조건 (삶속에 배워가는 모든 것)</li> </ul>		
Keyword	<ul style="list-style-type: none"> <li>- 무결성 : 인가되지 않은 방법으로 변경할 수 없도록 보호하는 성질</li> <li>- 개체 : 기본키, 엔티티중복 허용안함, 동일 PK 가질수없음, Null 허용안함</li> <li>- 참조 : 외래키, 외래키가 참조하는 다른 개체의 기본키에 해당하는 값이 기본키 값이나 Null 이어야 함.</li> </ul>		
풀이	이정현(125회 정보관리기술사)		

### 1. 데이터 무결성의 개요

데이터 무결성	<ul style="list-style-type: none"> <li>- 사용자가 관계형 테이블에 입력, 수정, 삭제, 조회의 데이터 조작을 수행할 때 데이터의 일관성과 정확성을 유지 할 수 있도록 하는 일련의 업무 규칙</li> </ul>	
데이터 무결성 종류	<p>Diagram illustrating relational integrity:</p> <ul style="list-style-type: none"> <li><b>Customer Table:</b> 고객번호 (Primary Key), 고객명, 성별</li> <li><b>Order Table:</b> 주문번호 (Primary Key), 고객번호 (Foreign Key), 주문수량, 할인율</li> <li><b>Relationship:</b> 속성무결성 (Customer.CustomerNumber = Order.CustomerNumber)</li> <li><b>Integrity Constraints:</b> 개체무결성 (Customer), 참조무결성 (Order), 사용자정의무결성 (Customer, Order relationship).</li> </ul>	
데이터 무결성 종류	1) 개체 무결성	<ul style="list-style-type: none"> <li>- 엔티티가 중복되지 않고, 동일한 기본키를 설정 할 없고, Null을 허용하지 않는 성질</li> </ul>
	2) 참조 무결성	<ul style="list-style-type: none"> <li>- 외래키가 참조하는 다른 개체의 기본키에 해당하는 값이 기본키 값이나 Null이어야 하는 성질</li> </ul>
	3) 속성 무결성	<ul style="list-style-type: none"> <li>- 속성의 값은 기본값, Null 여부 및 도메인이 지정된 규칙을 준수하여 존재 해야 하는 성질</li> </ul>
	4) 사용자정의 무결성	<ul style="list-style-type: none"> <li>- 사용자의 의미적 요구사항을 준수해야 하는 성질</li> </ul>
	5) 키 무결성	<ul style="list-style-type: none"> <li>- 한 릴레이션에 같은 키 값을 가진 튜플들은 허용 할 수 없는 성질</li> </ul>
	<ul style="list-style-type: none"> <li>- 개체무결성과 참조무결성은 테이블 내의 속성 값의 정확성과 유일성을 유지함에 있어 매우 중요</li> </ul>	

## 2. 개체 무결성

### 가. 개체 무결성 개념 및 조건

개체 무결성의 개념	개체 무결성 규칙
<p>주 키(기본키, PK, 특정행을 유일하게 인식하는 하나 이상의 열)는 널(NULL, 모르는 값, 정해지지 않은 값)을 포함하지 않는 성질</p>	<p>1) 식별자를 구성하는 각 속성이 널(NULL)이 아닐 것 2) 엔터티 내 특정 건의 유일성을 보장 할 것 3) 최소한의 속성 집합(Minimal Set) 일 것</p>

- 개체 무결성을 보장하지 못할 경우 원래의 값이 표시되지 않거나, 중복 값이 발생

### 나. 개체 무결성 위반사례 및 해결방안

규칙	위반사례	해결방안																																	
식별자를 구성하는 각 속성이 Null이 아닐 것	<p>1) T1을 이용해 T2, T3 생성</p> <p>T1</p> <table border="1"> <tr><th>PK</th><th>A</th><th>B</th><th>C</th><th>D</th></tr> <tr><td>NULL</td><td>A1</td><td>B1</td><td>C1</td><td>D1</td></tr> <tr><td>KEY1</td><td>A2</td><td>B2</td><td>C2</td><td>D2</td></tr> </table> <p>T2</p> <table border="1"> <tr><th>PK</th><th>A</th><th>B</th></tr> <tr><td>NULL</td><td>A1</td><td>B1</td></tr> <tr><td>KEY1</td><td>A2</td><td>B2</td></tr> </table> <p>T3</p> <table border="1"> <tr><th>PK</th><th>C</th><th>D</th></tr> <tr><td>NULL</td><td>C1</td><td>D1</td></tr> <tr><td>KEY1</td><td>C2</td><td>D2</td></tr> </table> <p>2) T2, T3 조인 SELECT PK, A, B, C, D FROM T2, T3 WHERE T2.PK = T3.PK</p> <p>3) NULL 식별 불가, T1과 다른 결과를 나타냄.</p>	PK	A	B	C	D	NULL	A1	B1	C1	D1	KEY1	A2	B2	C2	D2	PK	A	B	NULL	A1	B1	KEY1	A2	B2	PK	C	D	NULL	C1	D1	KEY1	C2	D2	<ul style="list-style-type: none"> <li>- Null 값 제거</li> <li>- Null 허용 금지</li> </ul>
PK	A	B	C	D																															
NULL	A1	B1	C1	D1																															
KEY1	A2	B2	C2	D2																															
PK	A	B																																	
NULL	A1	B1																																	
KEY1	A2	B2																																	
PK	C	D																																	
NULL	C1	D1																																	
KEY1	C2	D2																																	
엔터티 내 특정 건의 유일성을 보장	<p>1) T1을 이용해 T2, T3 생성</p> <p>T1</p> <table border="1"> <tr><th>PK</th><th>A</th><th>B</th><th>C</th><th>D</th></tr> <tr><td>KEY1</td><td>A1</td><td>B1</td><td>C1</td><td>D1</td></tr> <tr><td>KEY1</td><td>A2</td><td>B2</td><td>C2</td><td>D2</td></tr> </table> <p>T2</p> <table border="1"> <tr><th>PK</th><th>A</th><th>B</th></tr> <tr><td>KEY1</td><td>A1</td><td>B1</td></tr> <tr><td>KEY1</td><td>A2</td><td>B2</td></tr> </table> <p>T3</p> <table border="1"> <tr><th>PK</th><th>C</th><th>D</th></tr> <tr><td>KEY1</td><td>C1</td><td>D1</td></tr> <tr><td>KEY1</td><td>C2</td><td>D2</td></tr> </table> <p>2) T2, T3 조인 SELECT PK, A, B, C, D FROM T2, T3 WHERE T2.PK = T3.PK</p> <p>3) Product – 다대다 곱집합 발생</p>	PK	A	B	C	D	KEY1	A1	B1	C1	D1	KEY1	A2	B2	C2	D2	PK	A	B	KEY1	A1	B1	KEY1	A2	B2	PK	C	D	KEY1	C1	D1	KEY1	C2	D2	<ul style="list-style-type: none"> <li>- 주 키(PK)를 구성하는 각각의 컬럼 수를 최소한의 집합으로 설정</li> <li>- 자식 테이블에 영향 미칠 경우 주 키(PK) 제외</li> </ul>
PK	A	B	C	D																															
KEY1	A1	B1	C1	D1																															
KEY1	A2	B2	C2	D2																															
PK	A	B																																	
KEY1	A1	B1																																	
KEY1	A2	B2																																	
PK	C	D																																	
KEY1	C1	D1																																	
KEY1	C2	D2																																	
최소한의 속성 집합	<p>사원</p> <table border="1"> <tr><td># 사원번호</td><td>* 사원명</td></tr> <tr><td>100</td><td>장동건</td></tr> <tr><td>200</td><td>황정민</td></tr> <tr><td>300</td><td>송중기</td></tr> </table> <p>부서</p> <table border="1"> <tr><td># 부서번호</td><td>* 부서명</td></tr> <tr><td>100</td><td>장동건</td></tr> <tr><td>200</td><td>황정민</td></tr> <tr><td>200</td><td>송강호</td></tr> <tr><td>300</td><td>송중기</td></tr> </table> <p>&lt;유일성 보장&gt;</p> <p>&lt;유일성 깨짐&gt;</p>	# 사원번호	* 사원명	100	장동건	200	황정민	300	송중기	# 부서번호	* 부서명	100	장동건	200	황정민	200	송강호	300	송중기	<ul style="list-style-type: none"> <li>- 유일성이 보장되도록 최소한의 속성 집합으로 구성</li> <li>- Minimal Set 구성</li> </ul>															
# 사원번호	* 사원명																																		
100	장동건																																		
200	황정민																																		
300	송중기																																		
# 부서번호	* 부서명																																		
100	장동건																																		
200	황정민																																		
200	송강호																																		
300	송중기																																		

- 개체 무결성을 통해 표시되지 않는 값과 중복 값을 제거, 참조 무결성을 이용해 업무규칙 보장

### 3. 참조 무결성

#### 가. 참조 무결성 개념 및 조건

개체 무결성의 개념	개체 무결성 규칙
관계 엔터티(테이블)의 모든 외래 키 값은 관련 있는 관계 엔터티의 모든 주 키 값이 존재해야 하는 성질	<p>1) 입력 규칙</p> <ul style="list-style-type: none"> <li>- 자식 엔터티의 행이 입력될 경우</li> </ul> <p>2) 삭제(수정) 규칙</p> <ul style="list-style-type: none"> <li>- 부모 엔터티의 행을 삭제 할 경우</li> <li>- 행의 주 키(PK)를 수정 할 경우</li> </ul>

#### 나. 참조 무결성의 사례 및 설명

규칙	사례	설명																																							
입력 규칙	<table border="1"> <thead> <tr> <th colspan="3">주문</th> </tr> <tr> <th>주문번호</th> <th>주문일자</th> <th>고객번호</th> </tr> </thead> <tbody> <tr> <td>20150909-100</td> <td>20150911</td> <td>A199701</td> </tr> <tr> <td>20150909-101</td> <td>20150915</td> <td>A200002</td> </tr> <tr> <td>20150909-102</td> <td>20150917</td> <td>A200111</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th colspan="3">주문상품</th> </tr> <tr> <th>주문번호</th> <th>상품코드</th> <th>수량</th> </tr> </thead> <tbody> <tr> <td>20150909-100</td> <td>M1001</td> <td>100</td> </tr> <tr> <td>20150909-100</td> <td>T2000</td> <td>20</td> </tr> <tr> <td>20150909-101</td> <td>T0001</td> <td>3000</td> </tr> <tr> <td>20150909-101</td> <td>D1000</td> <td>5000</td> </tr> <tr> <td>20150909-101</td> <td>M1001</td> <td>50</td> </tr> <tr> <td>20150909-102</td> <td>T2000</td> <td>10</td> </tr> </tbody> </table>	주문			주문번호	주문일자	고객번호	20150909-100	20150911	A199701	20150909-101	20150915	A200002	20150909-102	20150917	A200111	주문상품			주문번호	상품코드	수량	20150909-100	M1001	100	20150909-100	T2000	20	20150909-101	T0001	3000	20150909-101	D1000	5000	20150909-101	M1001	50	20150909-102	T2000	10	<ul style="list-style-type: none"> <li>- "주문" 엔터티</li> <li>입력 전 "고객"</li> <li>엔터티에 먼저 등록이 되야함</li> <li>- "주문상품"</li> <li>엔터티 입력 전 "주문" 엔터티에 먼저 등록이 되야함</li> </ul>
주문																																									
주문번호	주문일자	고객번호																																							
20150909-100	20150911	A199701																																							
20150909-101	20150915	A200002																																							
20150909-102	20150917	A200111																																							
주문상품																																									
주문번호	상품코드	수량																																							
20150909-100	M1001	100																																							
20150909-100	T2000	20																																							
20150909-101	T0001	3000																																							
20150909-101	D1000	5000																																							
20150909-101	M1001	50																																							
20150909-102	T2000	10																																							
삭제(수정) 규칙	<table border="1"> <thead> <tr> <th colspan="2">고객</th> </tr> <tr> <th>고객번호</th> <th>성명</th> </tr> </thead> <tbody> <tr> <td>A199701</td> <td>장동건</td> </tr> <tr> <td>A200002</td> <td>황정민</td> </tr> <tr> <td>A200111</td> <td>송중기</td> </tr> </tbody> </table>	고객		고객번호	성명	A199701	장동건	A200002	황정민	A200111	송중기	<ul style="list-style-type: none"> <li>- 부모 엔터티인 "고객" 행의 삭제/수정시 "고객" 엔터티의 주 키(PK)를 수정금지</li> <li>- "주문" 엔터티의 행을 함께 삭제하거나 외래키(FK) 가 수정되야함.</li> </ul>																													
고객																																									
고객번호	성명																																								
A199701	장동건																																								
A200002	황정민																																								
A200111	송중기																																								

- 데이터 무결성의 보장될 경우, 프로젝트 개발 기간을 현저히 줄이는 것이 가능.

"끝"

#### 기출풀이 의견

6. 데이터베이스의 기본토픽입니다. 또한 목차가 세부적으로 주어지지 않았으므로, 자신의 인사이트 기반으로 답안 작성이 필요한 문제입니다. DBA 또는 DAP를 경험하거나, 공부하신 분들은 실무적 측면으로 공격적인 답안을 작성하시면 좋을 듯 합니다.