

# ISTQB® Certified Tester Syllabus Foundation Level

Compatible with Syllabus version 4.0

---

International Software Testing Qualifications Board

---



## Sample Exam D - Answers (샘플문제 D - 해설)

Released Version 1.3

Translated Version 1.0

**(사)케이에스티큐비 KSTQB**

Korean Software Testing Qualifications Board

Copyright Notice © International Software Testing Qualifications Board (hereinafter called ISTQB®) and Korean Software Testing Qualifications Board(hereinafter called KSTQB).

All rights reserved.

This document may be copied in its entirety, or extracts made, if the source is acknowledged.

## 서문 Introduction

### 이 문서의 목적

이 샘플 시험문제지의 예시 문항과 답안 및 관련 정답은 다음과 같은 목적으로 주제별 전문가와 숙련된 문제 작성자로 구성된 팀에 의해 작성되었습니다:

- ISTQB® 회원 국가보드(National Boards) 및 시험 기관의 문제 출제 활동 지원
- 교육 제공자 및 시험 응시자에게 시험 문제의 예시 제공

**이 문제들은 공식 시험에서 그대로 사용할 수 없습니다.**

실제 시험에는 다양한 문제가 포함될 수 있으며, 이 샘플 시험문제지는 출제 가능한 모든 문제 유형, 스타일 또는 길이의 예시를 갖고 있지 않으며, 공식 시험보다 더 어렵거나 쉬울 수 있습니다.

### 지침 Instructions

이 문서에서는 아래와 같은 사항을 제공합니다:

- 각 정답에 대해 다음을 포함하는 정답 키워드 표:
  - K-레벨, 학습 목표(LO: Learning Objective), 및 문제 당 점수(Point value)
- 각 정답에 대해 다음을 포함하는 추가 문제에 대한 정답 키워드 표:
  - K-레벨, 학습 목표(LO), 및 문제 당 점수
- 모든 문제를 포함하는 답안 세트:
  - 정답
  - 각 옵션 답안에 대한 근거 해설
  - K-레벨, 학습 목표, 및 문제 당 점수
- 모든 문제를 포함하는 추가 답안 세트 [모든 샘플 시험에 적용되는 것은 아님\*]:
  - 정답
  - 각 옵션 답안에 대한 근거
  - K-레벨, 학습 목표, 및 문제 당 점수
- 문제는 별도의 문서에 들어 있습니다.

## 정답표

문제 번호(#)	정답	학습 목표(LO)	K-레벨	배점
1	d	FL-1.1.1	K1	1
2	c	FL-1.2.3	K2	1
3	a	FL-1.3.1	K2	1
4	b	FL-1.4.1	K2	1
5	a	FL-1.4.3	K2	1
6	d	FL-1.4.5	K2	1
7	a	FL-1.5.2	K1	1
8	b	FL-1.5.3	K2	1
9	a	FL-2.1.2	K1	1
10	a	FL-2.1.3	K1	1
11	d	FL-2.1.4	K2	1
12	b	FL-2.1.6	K2	1
13	a	FL-2.2.2	K2	1
14	b	FL-2.3.1	K2	1
15	c	FL-3.1.1	K1	1
16	c	FL-3.1.2	K2	1
17	b	FL-3.2.2	K2	1
18	b	FL-3.2.3	K1	1
19	b	FL-4.1.1	K2	1
20	b, e	FL-4.2.1	K3	1

문제 번호(#)	정답	학습 목표(LO)	K-레벨	배점
21	c	FL-4.2.2	K3	1
22	a	FL-4.2.3	K3	1
23	b	FL-4.2.4	K3	1
24	c	FL-4.3.1	K2	1
25	a	FL-4.3.3	K2	1
26	c	FL-4.4.1	K2	1
27	d	FL-4.4.2	K2	1
28	d	FL-4.5.1	K2	1
29	a	FL-4.5.3	K3	1
30	b, d	FL-5.1.3	K2	1
31	a	FL-5.1.4	K3	1
32	b	FL-5.1.5	K3	1
33	c	FL-5.1.7	K2	1
34	b	FL-5.2.1	K1	1
35	b, e	FL-5.2.2	K2	1
36	c	FL-5.3.2	K2	1
37	d	FL-5.4.1	K2	1
38	a	FL-5.5.1	K3	1
39	b	FL-6.1.1	K2	1
40	c	FL-6.2.1	K1	1

## 정답

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
1	d	<p>a) 정답이 아닙니다. 테스트 대상에서 결함을 찾아 수정하는 것은 일반적인 테스트 목적이 아닙니다. 결함을 식별하는 것은 테스트의 목적이지만, 결함을 수정하는 것은 테스트 활동이 아닙니다.</p> <p>b) 정답이 아닙니다. 개발자와 효과적인 의사소통을 유지하는 것은 테스트의 일반적인 목적중 하나가 아닙니다. 이것이 이해관계자에게 정보를 제공해 현명한 결정을 내릴 수 있게 하는 등의 다른 테스트 목적을 달성하는 데 유용하지만, 테스트를 수행하는 주된 이유는 아니기 때문입니다.</p> <p>c) 정답이 아닙니다. 법적 요구사항이 충족되었는지 확인하는 것은 테스트의 일반적인 목적이 아닙니다. 벨리데이션은 시스템이 운영 환경에서 사용자 및 기타 이해관계자의 요구사항을 충족하는지 확인하는 것과 관련이 있습니다. 법적 요구사항 준수 여부를 확인하는 것은 베리피케이션의 한 형태입니다.</p> <p>d) 정답입니다. 테스트 대상의 품질에 대한 신뢰를 구축하는 것은 합격하는 테스트를 실행함으로써 달성됩니다.</p>	FL-1.1.1	K1	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
2	c	<p>a) 정답이 아닙니다. 보너스 계산 오류는 시스템의 장애이지 결함이 아닙니다.</p> <p>b) 정답이 아닙니다. 시스템이 장애인 사용자를 적절히 지원하지 않는 것은 결과적으로 벌금으로 이어지는 장애이지만, 벌금 그 자체는 장애가 아닙니다(규제 시스템이 올바르게 작동하는 것으로 보입니다).</p> <p>c) 정답입니다. 오류는 프로그래머에 의해 발생했으며, 이 실수는 그들이 심한 시간 압박 하에 일하면서 생긴 것인데, 이는 후속 결함의 근본 원인입니다.</p> <p>d) 정답이 아닙니다. 장애인 사용자를 적절히 고려하지 않은 사용자 인터페이스의 불량 설계는 설계자 오류로 인한 설계 결함입니다. 따라서 사용자 인터페이스의 설계에는 설계자 오류가 아닌 설계 결함이 포함되어 있습니다.</p>	FL-1.2.3	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
3	a	<p>a) 정답입니다. '테스트 효과는 줄어든다' 원칙은 변경되지 않은 코드에 대해 동일한 테스트를 반복해도 새로운 결함을 발견하기 어렵다는 개념과 관련이 있으며, 따라서, 테스트를 수정하는 것이 필수적일 수 있습니다. 상위 수준의 테스트 조건을 사용해 매번 새로운 테스트를 생성함으로써 테스트가 동일하지 않게 하고 '소모'되지 않도록 해야 합니다.</p> <p>b) 정답이 아닙니다. '결함 부재의 오류' 원칙은 많은 테스트를 수행하고 결함을 발견하지 못하더라도 사용자의 요구사항이 충족되도록 하는 것과 관련이 있습니다(즉, 벨리데이션도 필요함). 테스트 케이스를 생성하고 테스트를 실행하기 위해 상위 수준의 테스트 조건을 사용하는 것으로는 이 문제를 직접 해결하지 못합니다.</p> <p>c) 정답이 아닙니다. '조기 테스트는 시간과 비용을 절약한다' 원칙은 파생된 작업 산출물에서 후속 결함이 발생하는 것을 방지하기 위해 초기에 결함을 수정함으로써 비용을 줄이고 장애 가능성을 낮추는 것과 관련이 있습니다. 이는 일반적으로 테스트(정적 및 동적)를 가능한 빨리 시작함으로써 해결되지만, 테스트 케이스를 생성하고 테스트를 실행하기 위해 상위 수준의 테스트 조건을 사용하는 것은 이를 해결하지 않습니다.</p> <p>d) 정답이 아닙니다. '결함은 집중된다' 원리는 일반적으로 파레토 분포를 따르는 시스템의 결함 분포와 관련이 있습니다. 테스트 케이스를 생성하고 실행하기 위해 상위 수준의 테스트 조건을 사용하는 것은 이를 해결하지 않으며, 일반적으로 리스크 기반 테스트에서 다룹니다.</p>	FL-1.3.1	K2	3

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
4	b	<p>A. 테스트 분석 - 테스트가 필요한 기능을 식별하기 위해 테스트 베이스를 분석하고 테스트 조건으로 정의한 다음, 관련 리스크를 가지고 우선순위를 정합니다. 이 테스트 분석 중에 테스트 베이스의 결함이 드러나는 경우가 많고, 테스트 대상의 테스트 용이성을 평가하기도 합니다. (업무 4)</p> <p>B. 테스트 설계 - 테스트 조건을 사용해 테스트 케이스와 테스트 데이터 요구사항, 탐색적 테스팅을 위한 테스트 차터 등 필요한 테스트웨어를 만드는 작업이 포함됩니다. (업무 1)</p> <p>C. 테스트 구현 - 수동 및 자동 테스트 스크립트와 같은 테스트 절차를 테스트 케이스에서 만들고, 이를 테스트 스위트로 조합할 수 있습니다. 테스트 절차의 우선순위를 정하고 테스트 실행 일정으로 배열합니다. (업무 3)</p> <p>D. 테스트 완료 - 릴리스, 반복 주기 완료, 테스트 레벨 종료와 같은 프로젝트 마일스톤에서 이루어진다. 테스트웨어를 식별하고 재사용을 위해 보관하거나 해당 팀에 인계하고, 테스트 환경을 종료하며, 교훈과 향후 개선점을 파악하기 위해 테스트 활동을 분석합니다. (업무 2)</p> <p>따라서:</p> <p>a) 정답이 아닙니다.</p> <p>b) 정답입니다. 정답은 1B, 2D, 3C, 4A입니다.</p> <p>c) 정답이 아닙니다.</p> <p>d) 정답이 아닙니다.</p>	FL-1.4.1	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
5	a	<p>각 테스트웨어를 나열하고 이를 생성하는 테스트 활동을 고려할 때:</p> <ul style="list-style-type: none"> <li>i. 테스트 완료 보고서는 테스트 완료 활동의 산출물입니다.</li> <li>ii. 테스트 입력값과 예상 결과로 사용되는 데이터베이스에 저장된 데이터는 테스트 데이터로, 테스트 구현 활동의 산출물입니다.</li> <li>iii. 테스트 환경을 구축하는 데 필요한 요소 목록은 테스트 환경 요구사항으로, 테스트 설계 활동의 산출물입니다.</li> <li>iv. 실행 순서에 따라 문서화된 테스트 케이스 순서는 테스트 절차로, 테스트 구현 활동의 산출물입니다.</li> <li>v. 테스트 케이스는 테스트 설계 활동의 산출물입니다.</li> </ul> <p>테스트 구현은 다음과 같은 산출물을 만든다: 테스트 절차(iv), 자동화된 테스트 스크립트, 테스트 스위트, 테스트 데이터(ii), 테스트 실행 일정, 그리고 스텝, 드라이버, 시뮬레이터, 서비스 가상화 등 테스트 환경 요소.</p> <p>따라서:</p> <ul style="list-style-type: none"> <li>a) 정답입니다 목록의 ii와 iv는 테스트 구현 결과로 생성됩니다</li> <li>b) 정답이 아닙니다</li> <li>c) 정답이 아닙니다</li> <li>d) 정답이 아닙니다</li> </ul>	FL-1.4.3	K2	1



문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
6	d	<p>a) 정답이 아닙니다. 테스트 역할은 주로 테스트 분석, 테스트 설계, 테스트 구현, 테스트 실행과 같은 테스트의 기술적 측면에 대한 책임을 진다. 결함 식별을 위한 테스트 베이스 평가와 테스트 용이성을 위한 테스트 대상 평가는 테스트 분석의 일부로 수행하는 작업이므로, 테스트 역할이 수행할 가능성이 높다.</p> <p>b) 정답이 아닙니다. 테스트 역할은 주로 테스트 분석, 테스트 설계, 테스트 구현, 테스트 실행과 같은 테스트의 기술적 측면에 대한 책임을 집니다. 테스트 환경 요구사항 정의는 테스트 설계의 일부로 수행하는 작업이므로, 테스트 역할이 수행할 가능성이 높습니다.</p> <p>c) 정답이 아닙니다. 테스트 역할은 주로 테스트 분석, 테스트 설계, 테스트 구현, 테스트 실행과 같은 테스트의 기술적 측면에 대한 책임을 집니다. 테스트 대상의 테스트 용이성 평가는 테스트 분석의 일부로 수행하는 작업이므로, 테스트 역할이 수행할 가능성이 높습니다.</p> <p>d) 정답입니다. 테스트 관리 역할은 주로 테스트 계획, 테스트 모니터링과 제어, 테스트 완료와 관련된 활동을 담당합니다. 따라서 테스트 완료 활동의 주요 산출물인 테스트 완료 보고서 작성은 테스트 관리 역할이 수행할 가능성이 높습니다.</p>	FL-1.4.5	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
7	a	<p>a) 정답입니다. 전체 팀 접근법은 팀원 간 활발한 의사소통과 협업을 장려합니다.</p> <p>b) 정답이 아닙니다. 전체 팀 접근법은 품질에 대한 집단 책임을 우선시하지만, 개별 팀원은 여전히 품질에 대해 동등한 책임을 집니다.</p> <p>c) 정답이 아닙니다. 전체 팀 접근법은 팀이 어떻게 협업하는지에 관한 것으로, 더 높은 품질의 결과물을 목표로 합니다. 그러나 이것이 반드시 최종 사용자에게 더 빠른 배포로 이어지는 것은 아닙니다.</p> <p>d) 정답이 아닙니다. 전체 팀 접근법을 사용할 때 테스터는 비즈니스 담당자와 함께 인수 테스트를 만듭니다. 이 접근법이 외부 비즈니스 사용자와의 협업을 줄일 것이라는 제안은 없습니다.</p>	FL-1.5.2	K1	1

8	b	<p>테스팅 독립성의 장단점을 나열한 것을 각각 고려할 때:</p> <ul style="list-style-type: none"> <li>i. 이상적으로는 테스터와 개발자 간에 긴밀한 협업을 원하는데, 이는 격리에 의해 증가되지 않습니다. 따라서 이것은 단점입니다.</li> <li>ii. 테스터와 개발자는 다양한 배경, 기술적 관점, 잠재적 편향을 가지고 있어서, 테스터가 시스템 명세 및 구현 과정에서 이해관계자의 가정에 대해 유용하게 문제를 제기할 수 있습니다. 따라서 이것은 장점입니다.</li> <li>iii. 테스팅 독립성의 주요 단점은 테스터가 개발팀과 단절되어 의사소통 문제, 협업 부족, 잠재적으로 적대적 관계를 만들 수 있다는 점입니다. 이로 인해 테스터는 지연과 릴리스 프로세스의 병목 현상에 대한 책임을 질 수 있습니다. 따라서 이것은 단점입니다.</li> <li>iv. 테스팅 독립성의 단점 중 하나는 테스터가 개발팀과 단절되어 개발자가 품질에 대해 책임감을 덜 느끼게 될 수 있다는 점입니다. 따라서 이것은 단점입니다.</li> <li>v. 테스팅 독립성의 주된 이점은 테스터가 인지 편향을 포함해 다양한 배경, 기술적 관점, 잠재적 편향으로 인해 개발자와 비교하여 다른 유형의 장애와 결함을 식별할 가능성이 더 높다는 것입니다.</li> </ul> <p>따라서:</p> <ul style="list-style-type: none"> <li>a) 정답이 아닙니다</li> <li>b) 정답입니다 목록에서 이점을 보여주는 항목은 ii와 v입니다</li> <li>c) 정답이 아닙니다</li> <li>d) 정답이 아닙니다</li> </ul>	FL-1.5.3	K2	1
---	---	--	----------	----	---

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
9	a	<p>a) 정답입니다. 각 테스트 레벨은 구체적이면서 구별되는 테스트 목적을 가지고 있는데, 이는 각 테스트 레벨에서 서로 다른 형태의 테스트 대상(예: 단일 컴포넌트, 완성된 시스템)을 테스트하기 때문입니다. 테스트 목적이 겹치면 불필요한 중복으로 이어질 수 있습니다.</p> <p>b) 정답이 아닙니다. 특정 테스트 레벨을 위한 테스트 분석과 설계는 조기 테스트를 촉진하기 위해 상응하는 개발 단계에서 시작해야 합니다(예: 인수 테스트 분석 및 설계는 요구사항 분석 중에 시작해야 함). 테스트 구현은 일반적으로 더 늦게 시작되며, 테스트 실행은 테스트 레벨 도중 시작됩니다.</p> <p>c) 정답이 아닙니다. 특정 테스트 레벨을 위한 테스트 설계는 조기 테스트를 촉진하기 위해 상응하는 개발 단계에서 시작해야 합니다. 그러나 테스트 설계(예: 테스트 케이스 생성)는 초안이 아닌 합의된 테스트 베이스를 기반으로 해야 합니다. 그렇지 않으면 나중에 변경될 설계를 위한 테스트 케이스 생성에 상당한 테스트 노력이 낭비될 수 있습니다.</p> <p>d) 정답이 아닙니다. 품질 제어는 모든 개발 활동에 적용되므로, 모든 소프트웨어 개발 활동에는 이에 상응하는 테스트 활동이 있습니다. 그러나 동적 테스트와 정적 테스트에는 같은 대칭성이 적용되지 않습니다. 명백히 상응하는 동적 테스트 활동이 없는 일부 정적 테스트 활동(예: 정적 분석)이 있습니다.</p>	FL-2.1.2	K1	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
10	a	<p>a) 정답입니다. 행위 주도 개발(BDD)은 개발에서 잘 알려진 테스트 우선 접근법의 예입니다.</p> <p>b) 정답이 아닙니다. 테스트 레벨 주도 개발(Test Level Driven Development)은 개발에서 테스트 우선 접근법의 올바른 예가 아닙니다.</p> <p>c) 정답이 아닙니다. 기능 주도 개발(Function-Driven Development)은 개발에서 테스트 우선 접근법의 올바른 예가 아닙니다.</p> <p>d) 정답이 아닙니다. 성능 주도 개발(Performance-Driven Development)은 개발에서 테스트 우선 접근법의 올바른 예가 아닙니다.</p>	FL-2.1.3	K1	1
11	d	<p>a) 정답이 아닙니다. 데브옵스(devOps)는 일반적으로 성능, 신뢰성과 같은 비기능 품질 특성의 가시성을 높입니다.</p> <p>b) 정답이 아닙니다. 데브옵스에서 사용하는 지속적 통합/지속적 전달(CI/CD)과 같은 자동화된 프로세스는 안정적인 테스트 환경을 용이하게 합니다.</p> <p>c) 정답이 아닙니다. 데브옵스에서 사용하는 CI/CD와 같은 자동화된 프로세스는 일반적으로 수동 테스트의 필요성을 줄여줍니다.</p> <p>d) 정답입니다. 데브옵스 구현은 배포 파이프라인 정의 및 설정, CI/CD 도구 도입 및 유지보수, 테스트 자동화 수립 및 유지보수 필요성 등 여러 리스크와 난제를 야기할 수 있습니다.</p>	FL-2.1.4	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
12	b	<p>a) 정답이 아닙니다. 회고의 이점에는 문제 공유를 통한 팀 결속과 학습, 작업 관행 검토 및 개선을 통한 개발자와 테스터 간의 더 나은 협업 등이 있습니다. 전체 팀 접근법에 서 요구하는 대로 팀원이 품질 달성에 전적으로 기여하지 않았다고 느끼는 개인을 지적하는 것은 팀 결속과 협업에 기여하지 않습니다.</p> <p>b) 정답입니다. 회고에서는 프로젝트에서 성공적이었고 유지해야 하는 측면, 그리고 개선할 수 있는 영역과 개선 방법에 대해 토론합니다.</p> <p>c) 정답이 아닙니다. 회고의 이점은 프로세스 개선을 통한 효과성과 효율성 향상에 기반을 둡니다; 회고는 불만을 털어놓고 경영진과 고객을 비난할 기회가 아닙니다. 또한, 회고 결과는 보통 테스트 완료 보고서에 기록되므로, 회의에서 말한 내용은 다른 이해 관계자가 읽을 수 있습니다.</p> <p>d) 정답이 아닙니다. 회고는 보통 반복 주기가 끝날 때 열리는 회의로, 팀원들은 현재 반복 주기에서 발생한 품질 관련 문제에 대해 논의하는 데 초점을 둡니다. 회고는 다음 반복 주기에 대한 계획이나 기술적 결정을 내리는 데 사용되지 않습니다. 이는 다음 반복 주기 시작 시 반복 계획 회의에서 수행됩니다.</p>	FL-2.1.6	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
13	a	<p>a) 정답입니다. 정렬 함수가 목록이나 배열의 요소를 오름차순으로 정렬하는지 확인하는 것은 정렬 함수의 기능적 정확성을 평가하는 것으로, 기능 테스트의 일부입니다.</p> <p>b) 정답이 아닙니다. 정렬 함수가 1초 이내에 완료되어야 한다는 비기능 요구사항을 충족하는지 평가하는 것은 성능 효율성을 테스트하는 것으로, 비기능 테스트의 일부입니다.</p> <p>c) 정답이 아닙니다. 정렬 함수를 오름차순에서 내림차순으로 정렬하도록 수정하는 것이 얼마나 쉬운지 평가하는 것은 수정 용이성을 테스트하는 것으로, 비기능 유지보수성 테스트의 한 형태이며, 비기능 테스트의 일부입니다.</p> <p>d) 정답이 아닙니다. 정렬 함수를 32비트 아키텍처에서 64비트 아키텍처로 이동했을 때 여전히 올바르게 동작하는지 평가하는 것은 적응성을 테스트하는 것으로, 이동성 테스트의 한 형태이며, 비기능 테스트의 일부입니다.</p>	FL-2.2.2	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
14	b	<p>a) 정답이 아닙니다. 테스터가 환전 시스템의 변경 용이성을 확인할 수 있다고 가정하면, 이는 유지보수 테스트가 아니라 유지보수성 테스트를 통해 수행될 것이므로, 이는 유지보수 테스트의 계기가 아닙니다.</p> <p>b) 정답입니다. 시스템 수정(수정 또는 개선 등)은 유지보수 테스트의 계기 예시입니다. 환전 시스템의 환불 옵션 제거는 유지보수 테스트로 이어지는 수정 사항입니다.</p> <p>c) 정답이 아닙니다. 애자일 팀이 환전 시스템에 새로운 고객 충성도 기능을 추가하는 사용자 스토리 개발을 시작했다면, 이는 팀이 새로운 기능을 테스트한 다음 리그레션 테스트를 수행하게 될 것입니다. 이 상황에서는 유지보수 테스트가 필요하지 않습니다.</p> <p>d) 정답이 아닙니다. 환전 시스템이 현지 언어와 영어 환전 거래를 모두 지원하도록 재구성하는 것은 유지보수 테스트의 세(3)가지 계기인 시스템 수정, 운영 환경 변경, 시스템 폐기에 해당하지 않습니다.</p>	FL-2.3.1	K2	1



문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
15	c	<p>a) 정답이 아닙니다. 대부분의 작업 산출물은 어떤 형태로든 정적 테스트를 사용해 검사할 수 있으며, 계약서는 반드시 사람이 해석할 수 있어야 하므로 정적 테스트의 한 형태인 리뷰를 수행할 수 있습니다.</p> <p>b) 정답이 아닙니다. 대부분의 작업 산출물은 어떤 형태로든 정적 테스트를 사용해서 검사할 수 있으며, 테스트 계획서는 사람이 해석할 수 있어야 하므로 정적 테스트의 한 형태인 리뷰를 수행할 수 있습니다.</p> <p>c) 정답입니다. 대부분의 작업 산출물은 어떤 형태로든 정적 테스트를 사용해서 검사할 수 있습니다. 그러나 암호화된 코드처럼 사람이 해석하기에 너무 복잡하고 도구로 분석하면 안 되는 작업 산출물에는 적합하지 않습니다. 암호화된 코드는 인간에게 너무 복잡하고, 제대로 암호화되었다면 대부분의 도구로 분석할 수 없을 것입니다.</p> <p>d) 정답이 아닙니다. 대부분의 작업 산출물은 어떤 형태로든 정적 테스트를 사용해서 검사할 수 있으며, 테스트 차터는 사람이 해석할 수 있어야 하므로 정적 테스트의 한 형태인 리뷰를 수행할 수 있습니다.</p>	FL-3.1.1	K1	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
16	c	<p>접근할 수 없는 코드, 의도한 대로 구현되지 않은 설 패턴, 실행 불가능한 작업 산출물 등, 정적 테스트만으로 식별할 수 있는 결함 유형들이 있습니다. 또한 코드 리뷰에서 리뷰어가 발견할 수 있고 동적 테스트 중에 관찰 가능한 장애를 일으키는 프로그래밍 결함과 같이 정적 테스트와 동적 테스트 모두에서 식별할 수 있는 결함 유형들도 있습니다. 그리고 코드나 시스템을 실행할 때만 관찰할 수 있는 성능 문제나 메모리 문제 등, 동적 테스트만으로 탐지할 수 있는 결함 유형들도 있습니다.</p> <p>따라서,</p> <ul style="list-style-type: none"> <li>a) 정답이 아닙니다.</li> <li>b) 정답이 아닙니다.</li> <li>c) 정답입니다.</li> <li>d) 정답이 아닙니다.</li> </ul>	FL-3.1.2	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
17	b	<p>나열된 다섯(5)개의 설명과 해당하는 리뷰 프로세스 활동이 다음과 같습니다:</p> <ol style="list-style-type: none"> <li>1. 이것은 '의사소통 및 분석' 활동의 일부를 설명합니다</li> <li>2. 이것은 '수정 및 보고' 활동의 일부를 설명합니다</li> <li>3. 이것은 '개별 리뷰' 활동의 일부를 설명합니다</li> <li>4. 이것은 '계획' 활동의 일부를 설명합니다</li> <li>5. 이것은 '리뷰 착수' 활동의 일부를 설명합니다</li> </ol> <p>실러버스에 설명된 ISO/IEC 20246의 일반적인 리뷰 프로세스는 다음 활동을 논리적 순서로 포함합니다:</p> <ul style="list-style-type: none"> <li>• 계획 (4)</li> <li>• 리뷰 착수 (5)</li> <li>• 개별 리뷰 (3)</li> <li>• 의사소통 및 분석 (1)</li> <li>• 수정 및 보고 (2)</li> </ul> <p>따라서:</p> <ol style="list-style-type: none"> <li>a) 정답이 아닙니다</li> <li>b) 정답입니다. 활동의 정확한 순서는 4 - 5 - 3 - 1 - 2 입니다</li> <li>c) 정답이 아닙니다</li> <li>d) 정답이 아닙니다</li> </ol>	FL-3.2.2	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
18	b	<p>a) 정답이 아닙니다. 관리자는 리뷰 대상을 결정하고 직원과 시간 등의 자원을 할당할 책임이 있습니다.</p> <p>b) 정답입니다. 중재자(또는 퍼실리테이터)는 시간 관리, 토론 중재, 모든 사람이 자유롭게 의견을 말할 수 있는 안전한 환경 조성 등 리뷰 회의의 효과적인 진행에 책임이 있습니다</p> <p>c) 정답이 아닙니다. 의장은 리뷰에서 인정되는 역할이 아닙니다.</p> <p>d) 정답이 아닙니다. 리뷰 리더는 리뷰 팀원 선정, 리뷰 회의 일정 계획, 리뷰가 성공적으로 완료되도록 하는 등의 리뷰 프로세스를 감독할 책임이 있습니다.</p>	FL-3.2.3	K1	1
19	b	<p>a) 정답이 아닙니다. 이 문서는 테스트 대상의 내부 구조를 언급하지 않고 테스트 대상의 기대되는 동작을 명시합니다. 따라서 화이트박스 테스트 기법은 테스트 케이스 설계에 도움이 되지 않습니다.</p> <p>b) 정답입니다. 이 문서는 테스트 대상의 기대되는 동작을 명시하는 요구사항입니다. 따라서 이 경우 가장 적합한 테스트 기법은 블랙박스 테스트 기법(예: 경계값 분석 또는 결정 테이블 테스트)입니다.</p> <p>c) 정답이 아닙니다. 경험 기반 테스트 기법도 이 문서를 기반으로 테스트 케이스를 설계하는 데 사용할 수 있지만, 블랙박스 테스트 기법이 더 적합합니다. 이 문서는 정확한 비즈니스 규칙을 설명하고 있으며, 또한 "\$100을 초과하는" 같은 표현은 경계값 분석과 같은 블랙박스 테스트 기법으로 테스트해야 하는 중요한 동등 분할 경계가 존재함을 시사합니다.</p> <p>d) 정답이 아닙니다. 리스크 기반 테스트 기법은 공인된 테스트 기법 유형이 아닙니다.</p>	FL-4.1.1	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
20	b, e	<p>아직 커버되지 않은 동등 분할이 두 개 있는데, 이는 "학생 할인"과 "경로 할인"에 해당합니다.</p> <p>a) 정답이 아닙니다. <math>CY - BY = 64</math>이므로, 이 입력값은 이미 커버된 "할인 없음" 분할에 해당합니다.</p> <p>b) 정답입니다. <math>CY - BY = 65</math>이므로, 이 입력값은 아직 커버되지 않은 분할("경로 할인")에 해당합니다.</p> <p>c) 정답이 아닙니다. <math>CY - BY = -65</math>이므로, 이 입력값은 이미 커버된 "오류 메시지" 분할에 해당합니다.</p> <p>d) 정답이 아닙니다. <math>CY - BY = 18</math>이므로, 이 입력값은 이미 커버된 "할인 없음" 분할에 해당합니다.</p> <p>e) 정답입니다. <math>CY - BY = 0</math>이므로, 이 입력값은 아직 커버되지 않은 분할("학생 할인")에 해당합니다.</p>	FL-4.2.1	K3	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
21	c	<p>세 개의 동등 분할이 있습니다: {..., -2, -1}, {0, 1, 2}, {3, 4, ...}.</p> <p>2개 선택 경계값 분석(2-value BVA)을 위해서는 모든 동등 분할의 모든 경계값을 커버해야 합니다.</p> <p>경계값은 -1 ("온도가 너무 낮음" 분할), 0, 2 ("온도 정상" 분할), 3 ("온도가 너무 높음" 분할)입니다.</p> <p>따라서:</p> <p>a) 정답이 아닙니다</p> <p>b) 정답이 아닙니다</p> <p>c) 정답입니다. 정확한 옵션은 -1, 0, 2, 3입니다</p> <p>d) 정답이 아닙니다</p>	FL-4.2.2	K3	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
22	a	<p>테스트 케이스 TC1, TC2, TC3, TC4는 각각 결정 테이블의 규칙 R2, R3, R7, R6을 커버합니다.</p> <p>a) 정답입니다. "66 세", "미등록", "경험 없음"의 조건은 규칙 R4와 일치하는데, 이는 기존 테스트 케이스에서 커버되지 않으므로 이 테스트 케이스를 추가하면 결정 테이블 커버리지가 증가할 것입니다.</p> <p>b) 정답이 아닙니다. "55세", "미등록", "2년 경력"의 조건은 TC1에서 이미 커버된 규칙 R2와 일치합니다. 따라서 이 테스트 케이스를 추가해도 커버리지는 증가하지 않습니다.</p> <p>c) 정답이 아닙니다. "19세", "등록됨", "5년 경력"의 조건은 TC4에서 이미 커버된 규칙 R6과 일치합니다. 따라서 이 테스트 케이스를 추가해도 커버리지는 증가하지 않습니다.</p> <p>d) 정답이 아닙니다. 기존 테스트 케이스는 결정 테이블의 7개 열 중 4개만 커버합니다. 아직 커버되지 않은 열, 즉 R1, R4, R5를 커버하는 테스트 케이스를 추가하면 커버리지를 높일 수 있습니다.</p>	FL-4.2.3	K3	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
23	b	<p>a) 정답이 아닙니다. 이 5개 이벤트 순서는 서로 다른 4개의 유효 전이를 커버합니다(두 "NotAvailable" 이벤트는 모두 S1과 S3 사이의 동일한 전이에 해당). 이 테스트 케이스는 7개의 유효 전이 중 4개를 커버합니다.</p> <p>b) 정답입니다. 이 5개 이벤트 순서는 서로 다른 5개의 전이를 커버합니다(첫 번째 "Available" 이벤트는 S1과 S2 사이의 전이에 해당하고, 두 번째 "Available" 이벤트는 S3와 S2 사이의 전이에 해당하므로 두 개의 서로 다른 전이가 커버됨). 이 테스트 케이스는 7개의 유효 전이 중 5개를 커버하며 가장 높은 유효 전이 커버리지를 달성합니다.</p> <p>c) 정답이 아닙니다. 이 5개 이벤트 순서는 서로 다른 3개의 전이를 커버합니다(두 "Available" 이벤트는 모두 S1에서 S2로의 동일한 전이에 해당하고, 두 "ChangeRoom" 이벤트는 모두 S2에서 S1로의 동일한 전이에 해당). 이 테스트 케이스는 7개의 유효 전이 중 3개를 커버합니다.</p> <p>d) 정답이 아닙니다. 이 5개 이벤트 순서는 실행 가능한 테스트 케이스가 아닙니다. "Cancel" 후에 시스템이 End 상태에 도달하면 더 이상 유효한 전이를 실행할 수 없기 때문입니다.</p>	FL-4.2.4	K3	1



문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
24	c	<p>a) 정답이 아닙니다. 결함이 있는 코드 라인이 실행될 때 반드시 장애를 일으키는 것은 아닙니다. 예를 들어, <math>x := y / z</math> 라인은 <math>z</math>가 0일 때만 장애를 일으킵니다.</p> <p>b) 정답이 아닙니다. 100% 구문 커버리지가 100% 분기 커버리지를 보장하지는 않습니다. 예를 들어, 다음 코드에서 <math>x=0</math>인 테스트 케이스는</p> <pre> 1. IF (x=0) THEN 2. A; 3. ENDIF </pre> <p>100% 구문 커버리지를 달성하지만 1 ~ 3까지의 분기는 커버하지 않습니다.</p> <p>c) 정답입니다. 100% 구문 커버리지는 각 실행 가능한 구문이 최소한 한 번은 실행되었음을 의미합니다.</p> <p>d) 정답이 아닙니다. 제거된 테스트 케이스가 다른 두 테스트 케이스에서 커버하지 않는 일부 구문을 커버할 수도 있는데, 이 경우 남은 두 테스트 케이스가 100% 구문 커버리지를 달성하지 못할 것입니다.</p>	FL-4.3.1	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
25	a	<p>a) 정답입니다. 모든 화이트박스 테스트 기법이 공유하는 근본적인 강점은 테스트하는 동안 전체 소프트웨어 구현을 고려하므로, 소프트웨어 명세가 모호하거나, 구식이거나, 불완전한 경우에도 결함 탐지를 용이하게 한다는 것입니다. 즉, 화이트박스 테스트는 블랙박스 테스트로는 식별할 수 없는, 코드에 (의도적으로든 우연히든) 추가되었지만 있어서는 안 되는 추가 기능과 같은 결함을 찾을 수 있습니다.</p> <p>b) 정답이 아닙니다. 커버리지를 정확히 정의할 수 있다는 것은 올바른 이유가 아닙니다. 달성한 커버리지 수준이 커버리지 측정 가능성보다 훨씬 더 큰 영향을 미칠 것입니다.</p> <p>c) 정답이 아닙니다. 소프트웨어가 하나 이상의 요구사항을 구현하지 않으면, 화이트박스 테스트로 누락으로 인한 결함을 식별하기 어려울 것입니다.</p> <p>d) 정답이 아닙니다. 이것이 사실이긴 하지만, 정적 테스트와 동적 테스트 양쪽에서 사용될 수 있는 능력과 화이트박스 테스트가 부족한 명세로도 결함 식별을 용이하게 한다는 주장 사이에는 연관성이 없으므로 올바른 답이 아닙니다.</p>	FL-4.3.3	K2	1
26	c	<p>오류 추정은 테스트의 지식을 기반으로 오류, 결함 및 장애를 예측하는 것입니다.</p> <p>a) 정답이 아닙니다. 이것은 개발자의 오류를 예측하는 예입니다.</p> <p>b) 정답이 아닙니다. 이것은 결함을 예측하는 예입니다.</p> <p>c) 정답입니다. 이것은 결함의 잠재적 근본 원인의 예인데, 오류, 결함, 장애 중 어느 것도 아니며 테스트가 예측하기 어렵습니다.</p> <p>d) 정답이 아닙니다. 이것은 아마도 이 어플리케이션 분야의 이전 시스템 경험을 바탕으로 장애를 예측하는 예로 보입니다.</p>	FL-4.4.1	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
27	d	<p>a) 정답이 아닙니다. 탐색적 테스팅에서 테스트 케이스는 보통 테스트 분석, 테스트 구현, 테스트 실행과 함께 탐색적 테스팅 세션 중에 생성됩니다.</p> <p>b) 정답이 아닙니다. 탐색적 테스팅에서는 테스터가 테스트 대상에 대해 배우는 동안 테스트를 동시에 설계, 실행 및 평가합니다.</p> <p>c) 정답이 아닙니다. 탐색적 테스팅 결과는 테스터의 경험에 크게 의존하기 때문에, 이 결과를 리스크 예측 지표로 사용하거나 이전 탐색적 테스팅 세션과 비교해 결함이 더 적을지 또는 많을지 평가하는 데 사용할 수 있음에도 불구하고, 잔존 결함 수를 예측할 수 있는 신뢰할 만한 결함 예측 모델의 좋은 예시는 되지 않습니다.</p> <p>d) 정답입니다. 탐색적 테스팅 중에 테스터는 유용하다고 생각하는 모든 기법을 사용할 수 있습니다.</p>	FL-4.4.2	K2	1
28	d	<p>a) 정답이 아닙니다. 플래닝 포커(Planning poker)는 이미 작성된 사용자 스토리의 노력을 추정할 수 있습니다. 전달해야 하는 것이 무엇인지 이해하는 데는 도움이 되지 않습니다.</p> <p>b) 정답이 아닙니다. 리뷰는 협업적 사용자 스토리 작성 프랙티스가 아닙니다.</p> <p>c) 정답이 아닙니다. 반복 계획은 프로젝트 관련 프랙티스로, 무엇을 전달해야 하는지 이해하기 위한 것이 아니라 작업을 계획하는 데 사용됩니다.</p> <p>d) 정답입니다. 대화(Conversation)는 소프트웨어가 어떻게 사용될 것인지 설명하고, 종종 팀이 의미있는 인수 조건을 정의할 수 있게 해주므로, 무엇을 전달해야 하는지에 대한 공유된 비전을 얻을 수 있게 합니다.</p>	FL-4.5.1	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
29	a	<p>a) 정답입니다. 이 테스트 케이스는 인수 조건 2, 3과 관련이 있습니다. 가격 범위를 설정할 수 있는지(인수 조건 2) 그리고 가격 범위 필터를 조정한 후 결과가 동적으로 업데이트되는지(인수 조건 3)를 확인하기 때문입니다.</p> <p>b) 정답이 아닙니다. 이 테스트 케이스는 어떤 인수 조건과도 관련이 없습니다. 이는 고객이 그것을 설정할 수 있다는 것이 아니라, 필터가 기본 최소 및 최대 가격 범위를 동적으로 설정하는지를 확인합니다.</p> <p>c) 정답이 아닙니다. 이 테스트 케이스는 어떤 인수 조건과도 관련이 없습니다. 사용자 스토리에서 다루지 않은 통화 변환 기능을 확인합니다.</p> <p>d) 정답이 아닙니다. 이 테스트 케이스는 어떤 인수 조건과도 관련이 없습니다. 이 사용자 스토리에서 다루지 않은 애플리케이션의 다양한 브라우저와의 호환성을 확인합니다.</p>	FL-4.5.3	K3	1
30	b, d	<p>a) 정답이 아닙니다. 예산 승인은 시작 기준의 예입니다. 이미 완료된 활동에 대한 예산을 승인하는 것은 말이 되지 않습니다.</p> <p>b) 정답입니다. 예산 고갈은 유효한 완료 기준으로 볼 수 있습니다.</p> <p>c) 정답이 아닙니다. 자원의 가용성은 테스트를 위한 착수 기준의 예입니다.</p> <p>d) 정답입니다. 커버리지는 철저함의 척도이므로, 일반적인 완료 기준입니다.</p> <p>e) 정답이 아닙니다. 이는 프로젝트 시작 전에 확인하는 착수 기준의 예입니다.</p>	FL-5.1.3	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
31	a	<p>3점 추정 기법을 사용할 때, 최종 추정치(E)는 다음과 같이 계산됩니다:</p> $E = (a + 4 \cdot m + b) / 6$ <p>여기서 a는 가장 낙관적인 추정치, m은 가장 유력한 추정치, b는 가장 비관적인 추정치입니다.</p> <p>따라서:</p> <p>a) 정답입니다. 이 경우, 단일 테스트 케이스를 실행하는 데 대한 추정치는:</p> $E = (1\text{시간} + 4 \cdot 3\text{시간} + 8\text{시간}) / 6 = 3.5\text{시간}$ <p>따라서, 테스터가 4개의 테스트 케이스를 실행하는 데 필요한 총 시간은:</p> $3.5\text{시간} \cdot 4 = 14\text{시간}$ <p>b) 정답이 아닙니다</p> <p>c) 정답이 아닙니다</p> <p>d) 정답이 아닙니다</p>	FL-5.1.4	K3	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
32	b	<p>TC1이 가장 높은 커버리지(4/7 - Req1, Req3, Req4, Req7)를 달성하므로 가장 먼저 실행되어야 합니다.</p> <p>Req2, Req5, Req6은 아직 커버되지 않았습니다.</p> <p>남은 요구사항의 추가 커버리지를 가장 많이 달성하는 다음 테스트 케이스는 이 3개 중 2개(Req5, Req6)를 커버하는 TC3입니다. 따라서 TC3가 두 번째로 실행되어야 합니다.</p> <p>이제 아직 커버되지 않은 유일한 요구사항은 TC4에서 커버하는 Req2입니다. 따라서 TC4를 세 번째 테스트 케이스로 실행해야 합니다.</p> <p>그러므로, 마지막으로 실행되는 테스트 케이스는 TC2가 됩니다.</p> <p>따라서:</p> <p>a) 정답이 아닙니다</p> <p>b) 정답입니다</p> <p>c) 정답이 아닙니다</p> <p>d) 정답이 아닙니다</p>	FL-5.1.5	K3	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
33	c	<p>a) 정답이 아닙니다. 테스트 사분면은 테스트 레벨 간의 관계를 설명하는 것과 관련이 없습니다.</p> <p>b) 정답이 아닙니다. 테스트 사분면은 어떤 유형의 커버리지 평가에도 도움이 되지 않습니다.</p> <p>c) 정답입니다. 테스트 사분면은 관리자와 다른 이해관계자가 테스트 유형, 테스트 유형이 지원하는 활동(팀 지원 또는 제품 평가), 그리고 테스트 유형이 중점을 두는 관점(비즈니스 또는 기술 관점) 간의 관계를 이해할 수 있게 해줍니다.</p> <p>d) 정답이 아닙니다. 테스트 사분면은 심리학적 모델이 아닙니다.</p>	FL-5.1.7	K2	1
34	b	<p>리스크 평가는 정량적 또는 정성적 접근법을 사용하거나, 이 두 가지를 혼합해 사용할 수 있습니다. 정량적 접근법에서 리스크 수준은 리스크 발생 가능성과 리스크 영향을 곱한 값으로 계산합니다. 즉, 리스크 수준 = 리스크 발생 가능성 x 리스크 영향</p> <p>따라서, 리스크 영향 = 리스크 수준 / 리스크 발생 가능성이 된다.</p> <p>여기서는 리스크 영향 = <math>\\$1,000 / 50\% = \\$1,000 / 0.5 = \\$2,000</math>이다.</p> <p>따라서:</p> <p>a) 정답이 아닙니다</p> <p>b) 정답입니다</p> <p>c) 정답이 아닙니다</p> <p>d) 정답이 아닙니다</p>	FL-5.2.1	K1	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
35	b, e	<p>a) 정답이 아닙니다. 범위 증가(scope creep)는 기술적 문제와 관련된 프로젝트 리스크의 예입니다.</p> <p>b) 정답입니다. 열악한 아키텍처는 제품 특성을 언급하고 있으므로 제품 리스크의 예입니다.</p> <p>c) 정답이 아닙니다. 비용 절감은 조직 문제와 관련된 프로젝트 리스크의 예입니다.</p> <p>d) 정답이 아닙니다. 도구 지원 부족은 기술적 문제와 관련된 프로젝트 리스크의 예입니다.</p> <p>e) 정답입니다. 응답 시간 지연은 제품 특성을 언급하고 있으므로 제품 리스크의 예입니다.</p>	FL-5.2.2	K2	1
36	c	<p>a) 정답이 아닙니다. 테스트 진행 상황을 추적하고 추가 주의가 필요한 영역을 식별하는 것은 테스트의 지속적인 제어를 지원하는 예이며, 이는 테스트 보고서의 목적 중 하나입니다.</p> <p>b) 정답이 아닙니다. 실행된 테스트와 그 결과, 발견된 문제나 결함에 대한 정보를 제공하는 것은 주어진 테스트 레벨에서 수행된 테스트 활동을 요약하는 예이며, 이는 테스트 보고서의 목적 중 하나입니다.</p> <p>c) 정답입니다. 결함에 대한 정보를 제공하는 것은 결함 보고서의 목적이지만, 테스트 보고서의 목적이 아닙니다.</p> <p>d) 정답이 아닙니다. 다음 기간에 계획된 테스트에 대한 정보를 제공하는 것은 테스트 보고서의 목적 중 하나입니다.</p>	FL-5.3.2	K2	1



문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
37	d	<p>a) 정답이 아닙니다. 리스크 관리는 리스크 분석과 리스크 제어로 구성됩니다. 두 활동 모두 구성 항목을 다루는 것이 아니라 리스크를 다루기 때문에 릴리스를 구성한 파일의 재조립을 지원하지 않습니다.</p> <p>b) 정답이 아닙니다. 테스트 모니터링은 테스트에 대한 정보 수집과 관련이 있습니다. 이 정보는 테스트 진행 상황을 평가하고, 제품 리스크, 요구사항 또는 기타 인수 기준의 커버리지 목표 달성 등 테스트 완료 조건 또는 완료 조건과 연관된 테스트 작업의 충족 여부를 측정하는 데 사용됩니다. 테스트 제어는 테스트 모니터링의 정보를 사용해 가장 효과적이고 효율적인 테스트를 달성하기 위한 제어 지침, 지도, 필요한 시정 조치를 제공합니다. 이런 활동 모두 형상 항목의 관리와는 관련이 없습니다.</p> <p>c) 정답이 아닙니다. 전체 팀 접근법은 테스터가 팀 환경에서 효과적으로 일하고 팀 목표에 긍정적으로 기여하는 테스터의 기술을 바탕으로 합니다. 따라서 구성 항목이 아닌 팀 관련 문제에 초점을 맞춥니다.</p> <p>d) 정답입니다. 형상 관리는 작업 산출물을 식별, 제어, 추적하는 규율을 제공합니다. 형상 관리는 새로운 베이스라인이 만들어질 때 변경된 구성 항목의 기록을 유지합니다. 형상 관리를 사용하면 이전 테스트 결과를 재현하기 위해 이전 베이스라인으로 되돌릴 수 있습니다.</p>	FL-5.4.1	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
38	a	<p>a) 정답입니다. 이 정보를 추가하면 개발자가 동일한 입력 데이터를 사용할 수 있게 되어, 장애를 더 빨리 재현하고 결함을 더 빨리 식별할 수 있습니다.</p> <p>b) 정답이 아닙니다. 우선순위 값을 추가해도 결함 자체를 재현하는 데는 도움이 되지 않습니다.</p> <p>c) 정답이 아닙니다. 이 정보 중 일부는 가치가 있을 수 있지만, 각 단계 후에 메모리 덤프와 데이터베이스 스냅샷을 추가하는 것은 지나칩니다. 이런 산출물의 대부분은 개발자에게 쓸모없는 정보를 포함하고 보고서의 가독성을 떨어뜨리기 때문입니다. 또한 개발자가 이 정보를 분석하는 데 많은 시간을 소비하게 만들어 수정 프로세스를 길어지게 할 것입니다.</p> <p>d) 정답이 아닙니다. 질문은 특정 환경 구성에서 관찰된 결함을 개발자가 재현하는 데 도움을 주는 것에 관한 것이었습니다.</p>	FL-5.5.1	K3	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
39	b	<p>나열된 각 도구 범주를 고려할 때:</p> <ul style="list-style-type: none"> <li>i. 협업 도구 - 의사소통을 용이하게 합니다. 의사소통에는 테스트 실행의 용이성이 포함되지 않습니다.</li> <li>ii. 데브옵스 도구 - 데브옵스 제공 파이프라인, 작업 흐름 추적, 자동화된 빌드 프로세스, CI/CD를 지원합니다. 제공 파이프라인과 CI/CD는 CI를 위한 컴포넌트 테스트와 같은 테스트 실행의 용이성을 모두 포함합니다.</li> <li>iii. 관리 도구 - SDLC, 요구사항, 테스트, 결함, 형상 관리를 용이하게 해 테스트 프로세스 효율성을 높입니다. 이런 항목의 관리에는 테스트 실행의 용이성이 포함되지 않습니다.</li> <li>iv. 비기능 테스트 도구 - 테스터가 수동으로 수행하기 어렵거나 불가능한 비기능 테스트를 수행할 수 있게 합니다. 비기능 테스트에는 정적 테스트와 테스트 실행을 포함한 동적 테스트가 모두 포함될 수 있습니다.</li> <li>v. 테스트 설계 및 구현 도구 - 테스트 케이스, 테스트 데이터, 테스트 절차 생성을 용이하게 합니다. 이런 테스트웨어의 생성에는 테스트 실행의 용이성이 포함되지 않습니다.</li> </ul> <p>따라서:</p> <ul style="list-style-type: none"> <li>a) 정답이 아닙니다</li> <li>b) 정답입니다. 데브옵스 도구(ii)와 비기능 테스트 도구(iv) 모두 테스트 실행을 용이하게 합니다.</li> <li>c) 정답이 아닙니다</li> <li>d) 정답이 아닙니다</li> </ul>	FL-6.1.1	K2	1

문제 번호 (#)	정답	해설/근거	학습목표 (LO)	K-레벨	배점
40	c	<p>a) 정답이 아닙니다. 추가적인 심각도가 높은 결함의 식별은 테스트 자동화의 리스크라기보다는 이점일 것입니다.</p> <p>b) 정답이 아닙니다. 사람이 직접 도출하기에는 너무 복잡한 측정치를 제공하는 것은 일반적으로 테스트 자동화의 이점으로 간주됩니다.</p> <p>c) 정답입니다. 테스트 자동화가 개발 플랫폼과 호환되지 않는 경우, 테스트 자동화는 이들을 통합할 수 없게 되며, 예를 들어 테스트 입력을 테스트 대상에 전달하고 테스트 대상으로부터 테스트 결과를 받는 것이 불가능하게 됩니다.</p> <p>d) 정답이 아닙니다. 테스트 실행 시간이 상당히 단축되는 것은 일반적으로 테스트 자동화가 제공하는 이점으로 간주됩니다.</p>	FL-6.2.1	K1	1