



2과목-소프트웨어 개발

(Part 4. 애플리케이션 테스트 관리)

4. 애플리케이션 테스트 관리-SEC_01(애플리케이션 테스트) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(애플리케이션 테스트)

1. 소프트웨어 테스트에서 검증(Verification)과 확인(Validation)에 대한 설명으로 틀린 것은?

- ① 소프트웨어 테스트에서 검증과 확인을 구별하면 찾고자 하는 결함 유형을 명확하게 하는 데 도움이 된다.
- ② 검증은 소프트웨어 개발 과정을 테스트하는 것이고, 확인은 소프트웨어 결과를 테스트 것이다.
- ③ 검증은 작업 제품이 요구 명세의 기능, 비기능 요구사항을 얼마나 잘 준수하는지 측정하는 작업이다.
- ④ 검증은 작업 제품이 사용자의 요구에 적합한지 측정하며, 확인은 작업 제품이 개발자의 기대를 충족시키는지 측정한다.

2. 다음 설명의 소프트웨어 테스트의 기본 원칙은?

파레토 법칙이 좌우한다.

애플리케이션 결함의 대부분은 소수의 특정한 모듈에 집중되어 존재한다.

결함은 발생한 모듈에서 계속 추가로 발생할 가능성이 높다.

- ① 살충제 패러독스 ② 결함 집중
- ③ 오류 부재의 궤변 ④ 완벽한 테스트는 불가능

3. 소프트웨어 테스트에서 오류의 80%는 전체 모듈의 20% 내에서 발견된다는 법칙은?

- ① Brooks의 법칙 ② Boehm의 법칙
- ③ Pareto의 법칙 ④ Jackson의 법칙

4. 다음 중 애플리케이션 테스트에 대한 설명으로 틀린 것은?

- ① 애플리케이션 테스트는 프로그램 실행 전에 코드 리뷰, 인스펙션 등을 통해 사전에 오류를 발견하여 예방할 수 있다.
- ② 애플리케이션 테스트를 반복적으로 실행하여 제품의 신뢰도를 향상시킬 수 있다.
- ③ 테스트는 프로그램 개발이 완료된 후 체계적으로 계획하여 실행해야 한다.
- ④ 성공적인 테스트는 아직 발견되지 않은 오류를 찾아내는 것이다.

4. 애플리케이션 테스트 관리-SEC_01(애플리케이션 테스트) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(애플리케이션 테스트)

5. 다음 중 애플리케이션 테스트에 대한 설명으로 틀린 것은?

- ① 애플리케이션 테스트는 소프트웨어에 잠재되어 있는 결함을 찾아내는 일련의 행위이다.
- ② 테스트는 고객의 요구사항을 만족했는지 Verification 해야 한다.
- ③ 테스트는 오류 검출뿐만 아니라 새로운 오류의 유입도 방지 할 수 있다.
- ④ 테스트를 효과적으로 실행하면 최소한의 시간과 노력으로 많은 결함을 찾을 수 있다.

6. 개발된 소프트웨어가 사용자의 요구사항 및 기대 수준 등을 만족하는지 점검하기 위해 애플리케이션 테스트를 진행하려고 할 때, 이에 대한 설명으로 틀린 것은?

- ① 완벽한 테스트는 불가능하다.
- ② 테스트는 정황(Context)에 의존한다.
- ③ 파레토 법칙을 적용할 수 있다.
- ④ 테스트와 위험은 정비례한다.

4. 애플리케이션 테스트 관리-SEC_02(애플리케이션 테스트의 분류) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(애플리케이션 테스트의 분류)

1. 테스트를 목적에 따라 분류했을 때, 강도(Stress) 테스트에 대한 설명으로 옳은 것은?

- ① 시스템에 고의로 실패를 유도하고 시스템이 정상적으로 복귀 하는지 테스트한다.
- ② 시스템에 과다 정보량을 부과하여 과부하 시에도 시스템이 정상적으로 작동되는지를 테스트한다.
- ③ 사용자의 이벤트에 시스템이 응답하는 시간, 특정 시간 내에 처리하는 업무량, 사용자 요구에 시스템이 반응하는 속도 등을 테스트한다.
- ④ 부당하고 불법적인 침입을 시도하여 보안 시스템이 불법적인 침투를 잘 막아내는지 테스트한다.

2. 다음 중, 정적 테스트와 동적 테스트에 대한 설명으로 틀린 것은?

- ① 정적 테스트는 개발한 프로그램을 실행하지 않고 테스트 한다. ② 동적 테스트는 개발한 프로그램을 직접 실행하면서 오류를 찾는 테스트이다.
- ③ 동적 테스트에는 워크스루, 인스펙션, 코드 검사 등이 있다.
- ④ 정적 테스트는 개발 초기에 결함을 발견함으로써 개발 비용을 낮추는데 도움이 된다.

3. 다음 중 확인(Validation) 테스트에 대한 설명으로 옳은 것은?

- ① 개발자의 시각에서 테스트를 진행한다.
- ② 제품이 올바르게 생산되고 있는가를 확인한다.
- ③ 소프트웨어가 명세서대로 만들어졌는지를 중점을 두고 테스트 한다.
- ④ 소프트웨어가 사용자의 요구사항을 충족시키는가에 중점을 두고 테스트한다.

4. 다음 중 애플리케이션 테스트에 대한 설명으로 틀린 것은?

- ① 사용자의 요구사항에 대한 명세를 빠짐없이 테스트 케이스로 만들어 구현하고 있는지 확인하는 테스트는 명세 기반 테스트이다.
- ② 테스터의 이전 경험과 기술을 기반으로 수행하는 테스트는 경험 기반 테스트이다.
- ③ 소프트웨어 내부의 논리 흐름에 따라 테스트 케이스 작성하고 확인하는 테스트는 구조 기반 테스트이다.
- ④ 동등 분할, 경계 값 분석, 구문 기반, 결정 기반 등은 명세 기반 테스트이다.

4. 애플리케이션 테스트 관리-SEC_02(애플리케이션 테스트의 분류) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(애플리케이션 테스트의 분류)

5. 다음 중 워크스루(Walkthrough)와 인스펙션(Inspection)에 대한 설명으로 가장 옳지 않은 것은?

- ① 워크스루는 전문가들에 의해 개발자의 작업 내역이 검토 된다.
- ② 워크스루는 제품 개발자가 주체가 된다.
- ③ 워크스루는 오류 발견과 발견된 오류의 문제 해결에 중점을 둔다.
- ④ 인스펙션은 워크스루를 발전시킨 형태이다.

4. 애플리케이션 테스트 관리-SEC_03(테스트 기법에 따른 애플리케이션 테스트) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(테스트 기법에 따른 애플리케이션 테스트)

1. 소프트웨어 테스트와 관련한 설명으로 틀린 것은?

- ① 화이트박스 테스트는 모듈의 논리적인 구조를 체계적으로 점검 할 수 있다.
- ② 블랙박스 테스트는 프로그램의 구조를 고려하지 않는다.
- ③ 테스트 케이스에는 일반적으로 시험 조건, 테스트 데이터, 예상 결과가 포함되어야 한다.
- ④ 화이트박스 테스트에서 기본 경로(Basis Path)란 흐름 그래프 의 시작 노드에서 종료 노드까지의 서로 독립된 경로로 사이클을 허용하지 않는 경로를 말한다.

2. White Box Testing에 대한 설명으로 옳지 않은 것은?

- ① Base Path Testing, Boundary Value Analysis가 대표적인 기법이다.
- ② Source Code의 모든 문장을 한 번 이상 수행함으로써 진행된다.
- ③ 모듈 안의 작동을 직접 관찰할 수 있다.
- ④ 산출물의 각 기능별로 적절한 프로그램의 제어 구조에 따라 선택, 반복 등의 부분들을 수행함으로써 논리적 경로를 점검한다.

3. 블랙박스 테스트를 이용하여 발견할 수 있는 오류가 아닌 것은?

- ① 비 정상적인 자료를 입력해도 오류 처리를 수행하지 않는 경우
- ② 정상적인 자료를 입력해도 요구된 기능이 제대로 수행되지 않는 경우
- ③ 반복 조건을 만족하는데도 루프 내의 문장이 수행되지 않는 경우
- ④ 경계 값을 입력할 경우 요구된 출력 결과가 나오지 않는 경우

4. 블랙박스 테스트 기법으로 거리가 먼 것은?

- | | |
|------------|-------------|
| ① 기초 경로 검사 | ② 동치 클래스 분해 |
| ③ 경계 값 분석 | ④ 원인 결과 그래프 |

4. 애플리케이션 테스트 관리-SEC_03(테스트 기법에 따른 애플리케이션 테스트) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(테스트 기법에 따른 애플리케이션 테스트)

5. 평가점수에 따른 성적부여는 다음 표와 같다. 이를 구현한 소프트웨어를 경계 값 분석 기법으로 테스트 하고자 할 때 다음 중 테스트 케이스의 입력 값으로 옳지 않은 것은?

평가 점수	성적
80~100	A
60~79	B
0~59	C

- ① 59 ② 80
- ③ 90 ④ 101

6. 화이트박스 검사 기법에 해당하는 것으로만 짝지어진 것은?

㉠ 데이터 흐름 검사	㉡ 루프 검사
㉢ 동등 분할 검사	㉣ 경계값 분석
㉤ 원인 결과 그래프 기법	㉥ 오류 예측 기법

- ① ㉠, ㉡ ② ㉠, ㉣
- ③ ㉡, ㉤ ④ ㉢, ㉥

7. 화이트박스 테스트와 관련한 설명으로 틀린 것은?

- ① 화이트박스 테스트의 이해를 위해 논리 흐름도(Logic Flow Diagram)를 이용할 수 있다.
- ② 테스트 데이터를 이용해 실제 프로그램을 실행함으로써 오류를 찾는 동적 테스트(Dynamic Test)에 해당한다.
- ③ 프로그램의 구조를 고려하지 않기 때문에 테스트 케이스는 프로그램 또는 모듈의 요구나 명세를 기초로 결정한다.
- ④ 테스트 데이터를 선택하기 위하여 검증 기준(Test Coverage)을 정한다.

8. 모듈의 논리적 구조를 체계적으로 점검하는 구조 테스트로, 이 방식의 종류에는 기초 경로 검사, 조건 검사, 데이터 흐름 검사, 루프 검사 등이 있는 것은?

- ① 화이트 박스 테스트
- ② 블랙 박스 테스트
- ③ 레드 박스 테스트
- ④ 블루 박스 테스트

4. 애플리케이션 테스트 관리-SEC_04(개발 단계에 따른 애플리케이션 테스트) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(개발 단계에 따른 애플리케이션 테스트)

1. 개별 모듈을 시험하는 것으로, 모듈이 정확하게 구현되었는지, 예정한 기능이 제대로 수행되는지를 점검하는 것이 주 목적인 테스트는?

- ① 통합 테스트(Integration Test)
- ② 단위 테스트(Unit Test)
- ③ 시스템 테스트(System Test)
- ④ 인수 테스트(Acceptance Test)

2. 다음 중 단위 테스트를 통해 발견할 수 있는 오류가 아닌 것은?

- ① 알고리즘 오류에 따른 원치 않는 결과
- ② 탈출구가 없는 반복문의 사용
- ③ 모듈 간의 비정상적 상호 작용으로 인한 원치 않는 결과
- ④ 틀린 계산 수식에 의한 잘못된 결과

3. 검증(Validation) 검사 기법 중 개발자의 장소에서 사용자가 개발자 앞에서 행해지며, 오류와 사용상의 문제점을 사용자와 개발자가 함께 확인하면서 검사하는 기법은?

- ① 디버깅 검사 ② 형상 검사
- ③ 자료구조 검사 ④ 알파 검사

4. 알파, 베타 테스트와 가장 밀접한 연관이 있는 테스트 단계는?

- ① 단위 테스트 ② 인수 테스트
- ③ 통합 테스트 ④ 시스템 테스트

4. 애플리케이션 테스트 관리-SEC_04(개발 단계에 따른 애플리케이션 테스트) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(개발 단계에 따른 애플리케이션 테스트)

5. 필드 테스트(Field Testing)이라고도 불리며, 개발자 없이 고객의 사용환경에 소프트웨어를 설치하여 검사를 수행하는 인수 검사 기법은?

- ① 베타 검사 ② 알파 검사
- ③ 형상 검사 ④ 복구 검사

6. 소프트웨어 생명주기 모델 중 V 모델과 관련한 설명으로 틀린 것은?

- ① 요구 분석 및 설계 단계를 거치지 않으며 항상 통합 테스트를 중심으로 V 형태를 이룬다.
- ② Perry에 의해 제안되었으며 세부적인 테스트 과정으로 구성 되어 신뢰도 높은 시스템을 개발하는데 효과적이다.
- ③ 개발 작업과 검증 작업 사이의 관계를 명확히 들어내 놓은 폭포수 모델의 변형이라고 볼 수 있다.
- ④ 폭포수 모델이 산출물 중심이라면 V 모델은 작업과 결과의 검증에 초점을 둔다.

7. 다음 중 개발 단계에 따른 소프트웨어 테스트 종류가 아닌 것은?

- ① 단위 테스트
- ② 시스템 테스트
- ③ 화이트박스 테스트
- ④ 통합 테스트

4. 애플리케이션 테스트 관리-SEC_05(통합 테스트) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(통합 테스트)

1. 테스트 드라이버(Test Driver)에 대한 설명으로 틀린 것은?

- ① 시험 대상 모듈을 호출하는 간이 소프트웨어이다.
- ② 필요에 따라 매개 변수를 전달하고 모듈을 수행한 후의 결과를 보여줄 수 있다.
- ③ 상향식 통합 테스트에서 사용된다.
- ④ 테스트 대상 모듈이 호출하는 하위 모듈의 역할을 한다.

2. 통합 테스트(Integration Test)와 관련한 설명으로 틀린 것은?

- ① 시스템을 구성하는 모듈의 인터페이스와 결합을 테스트 하는 것이다.
- ② 하향식 통합 테스트의 경우 넓이 우선(Breadth First)방식으로 테스트를 할 모듈을 선택할 수 있다.
- ③ 상향식 통합 테스트의 경우 시스템 구조도의 최상위에 있는 모듈을 먼저 구현하고 테스트한다.
- ④ 모듈 간의 인터페이스와 시스템의 동작이 정상적으로 잘되고 있는지를 빨리 파악하고자 할 때 상향식 보다는 하향식 통합 테스트를 사용하는 것이 좋다.

3. 하향식 통합에 있어서 모듈 간의 통합 시험을 위해 일시적으로 필요한 조건만을 가지고 임시로 제공되는 시험용 모듈을 무엇이라고 하는가?

- ① Stub ② Driver
- ③ Procedure ④ Function

4. 다음이 설명하는 애플리케이션 통합 테스트 유형은?

깊이 우선 방식 또는 너비 우선 방식이 있다.

상위 컴포넌트를 테스트 하고 점증적으로 하위 컴포넌트를 테스트 한다.

하위 컴포넌트 개발이 완료되지 않은 경우 스텝(Stub)을 사용하기도 한다.

- ① 하향식 통합 테스트 ② 상향식 통합 테스트
- ③ 회귀 테스트 ④ 빅뱅 테스트

4. 애플리케이션 테스트 관리-SEC_05(통합 테스트) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(통합 테스트)

5. 상향식 통합 테스트(Bottom-up Integration Test)의 과정이 옳게 나열된 것은?

- ㉠ 드라이버라는 제어 프로그램의 작성
- ㉡ 낮은 수준의 모듈들을 클러스터로 결합
- ㉢ 클러스터의 검사
- ㉣ 클러스터를 상위로 결합

- ① ㉠ -> ㉡ -> ㉢ -> ㉣ ② ㉡ -> ㉠ -> ㉢ -> ㉣
- ③ ㉡ -> ㉢ -> ㉠ -> ㉣ ④ ㉠ -> ㉡ -> ㉣ -> ㉢

6. 하향식 통합에 있어서 모듈 간의 통합 시험을 하기 위해 일시적으로 필요한 조건만을 가지고 임시로 제공되는 시험용 모듈을 무엇이라고 하는가?

- ① Driver
- ② Stub
- ③ Sub-Program
- ④ Dummy-Program

7. 하향식 통합에 대한 설명으로 가장 적합하지 않은 것은?

- ① 프로그램의 상위 모듈에서 하위 모듈 방향으로 통합하면서 테스트 하는 기법이다.
- ② 마지막까지 독립된 프로그램 형태를 갖지 못한다.
- ③ 주요 제어 모듈의 종속 모듈들을 스텝으로 대체한다.
- ④ 깊이 우선 방식이나 넓이 우선 방식에 의해 통합한다.

4. 애플리케이션 테스트 관리- SEC_06(애플리케이션 테스트 프로세스) 출제 예상 문제

출제 예상 문제(애플리케이션 테스트 프로세스)

1. 다음 중 애플리케이션 테스트의 과정으로 옳게 나열된 것은?

- ㉠ 테스트 분석
- ㉡ 테스트 계획
- ㉢ 테스트 실행
- ㉣ 테스트 케이스 설계
- ㉤ 테스트 결과 분석

- ① ㉠ -> ㉡ -> ㉢ -> ㉣ -> ㉤
- ② ㉠ -> ㉡ -> ㉣ -> ㉢ -> ㉤
- ③ ㉡ -> ㉠ -> ㉣ -> ㉢ -> ㉤
- ④ ㉡ -> ㉠ -> ㉢ -> ㉣ -> ㉤

2. 다음 중 테스트를 마치면 산출되는 문서가 아닌 것은?

- ① 테스트 계획서
- ② 테스트 케이스
- ③ 테스트 다이어그램
- ④ 테스트 결과서

3. 테스트를 진행한다고 할 때 언제 오류를 발견하는 것이 가장 좋은가?

- ① 요구사항 분석 단계
- ② UI 설계 단계
- ③ UI 구현 단계
- ④ 결함 추적 및 관리 단계

4. 다음 중, 결함관리 순서로 나열할 경우 올바른 것은?

- ㉠ 에러 발견
- ㉡ 에러 등록
- ㉢ 에러 분석
- ㉣ 결함 확정
- ㉤ 결함 할당
- ㉥ 결함 조치

- ① ㉠ -> ㉡ -> ㉢ -> ㉣ -> ㉤ -> ㉥
- ② ㉠ -> ㉢ -> ㉡ -> ㉣ -> ㉤ -> ㉥
- ③ ㉡ -> ㉣ -> ㉤ -> ㉥ -> ㉠ -> ㉢
- ④ ㉡ -> ㉣ -> ㉤ -> ㉢ -> ㉠ -> ㉥

4. 애플리케이션 테스트 관리- SEC_06(애플리케이션 테스트 프로세스) 출제 예상 문제

출제 예상 문제(애플리케이션 테스트 프로세스)

5. 결함(Defect)의 원인이 되는 것으로, 일반적으로 소프트웨어 개발자, 분석가 등 사람에 의해 발생한 실수를 무엇이라고 하는가?

- ① 결함
- ② 결점
- ③ 에러
- ④ 버그

4. 애플리케이션 테스트 관리- SEC_07(테스트 케이스/테스트 시나리오/테스트 오라클) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(테스트 케이스/테스트 시나리오/테스트 오라클)

1. 다음이 설명하는 테스트 용어는?

- 테스트의 결과가 참인지 거짓인지를 판단하기 위해서 사전에 정의된 참 값을 입력하여 비교하는 기법 및 활동을 말한다.
- 종류에는 참, 샘플링, 휴리스틱, 일관성 검사가 존재한다.

- ① 테스트 케이스 ② 테스트 시나리오
③ 테스트 오라클 ④ 테스트 데이터

2. 다음 중 테스트 오라클에 대한 설명으로 옳지 않은 것은?

- ① 샘플링 오라클 : 특정한 몇몇 테스트 케이스의 입력 값들에 대해서만 기대하는 결과를 제공하는 오라클이다.
- ② 토탈 오라클 : 모든 테스트 케이스의 입력 값에 대해 기대하는 결과를 제공하는 오라클이다.
- ③ 휴리스틱 오라클 : 특정 테스트 케이스의 입력 값에 대해 기대하는 결과를 제공하고, 나머지 입력 값들에 대해서는 추정으로 처리하는 오라클이다.
- ④ 일관성 검사 오라클 : 애플리케이션의 변경이 있을 경우 테스트 케이스의 수행 전과 후의 결과 값이 동일한지를 확인하는 오라클 이다.

3. 테스트 케이스와 관련한 설명으로 틀린 것은?

- ① 테스트의 목표 및 테스트 방법을 결정하기 전에 테스트 케이스를 작성해야 한다.
- ② 프로그램에 결함이 있더라도 입력에 대해 정상적인 결과를 낼 수 있기 때문에 결함을 검사할 수 있는 테스트 케이스를 찾는 것이 중요하다.
- ③ 개발된 서비스가 정의된 요구사항을 준수하는지 확인하기 위한 입력 값과 실행 조건, 예상 결과의 집합으로 볼 수 있다.
- ④ 테스트 케이스 실행이 통과되었는지 실패하였는지 판단하기 위한 기준을 테스트 오라클(Test Oracle)이라고 한다.

4. 다음 중 테스트 시나리오(Test Scenario)에 대한 설명으로 가장 옳지 않은 것은?

- ① 테스트 시나리오는 테스트 수행을 위한 여러 개의 테스트 케이스 들의 집합이다.
- ② 테스트 시나리오는 어떤 기능을 어떤 순서대로 테스트 할 것인지 절차를 기술한다.
- ③ 테스트 시나리오는 전체를 하나의 시나리오로 작성해야 한다.
- ④ 테스트 시나리오를 통해 테스트 순서를 미리 정함으로써 테스트 항목을 빠짐없이 수행할 수 있다.

4. 애플리케이션 테스트 관리- SEC_07(테스트 케이스/테스트 시나리오/테스트 오라클) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(테스트 케이스/테스트 시나리오/테스트 오라클)

5. 다음 중 테스트 오라클의 종류가 아닌 것은?

- ① 참(True) 오라클
- ② 거짓(False) 오라클
- ③ 샘플링(Sampling) 오라클
- ④ 추정(Heuristic) 오라클

6. 다음 중 테스트 시나리오와 테스트 케이스에 대한 설명으로 가장 옳지 않은 것은?

- ① 테스트 시나리오는 테스트 케이스의 동작 순서를 기술한 문서이다.
- ② 테스트 케이스는 테스트 절차를 명세한 문서이다.
- ③ 테스트 시나리오는 개발된 모듈 또는 프로그램 간의 연계가 정상적으로 동작하는지 테스트할 수 있도록 작성해야 한다.
- ④ 테스트 케이스는 테스트할 시스템이 수행해야 할 액션들로 구성된 일련의 단계이다.

4. 애플리케이션 테스트 관리-SEC_08(테스트 자동화 도구) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(테스트 자동화 도구)

1. 테스트 케이스 자동 생성 도구를 이용하여 테스트 데이터를 찾아내는 방법이 아닌 것은?

- ① 스텝(Stub)과 드라이버(Driver)
- ② 입력 도메인 분석
- ③ 랜덤(Random) 테스트
- ④ 자료 흐름도

2. 애플리케이션의 처리량, 응답 시간, 경과 시간, 자원 사용률에 대해 가상의 사용자를 생성하고 테스트를 수행함으로써 성능 목표를 달성하였는지를 확인하는 테스트 자동화 도구는?

- ① 명세 기반 테스트 설계 도구
- ② 코드 기반 테스트 설계 도구
- ③ 기능 테스트 수행 도구
- ④ 성능 테스트 도구

3. 단위 테스트에서 테스트의 대상이 되는 하위 모듈을 호출하고, 파라미터를 전달하는 가상의 모듈로 상향식 테스트에 필요한 것은?

- ① 테스트 스텝(Test Stub)
- ② 테스트 드라이버(Test Driver)
- ③ 테스트 슈트(Test Suites)
- ④ 테스트 케이스(Test Case)

4. 다음 중 테스트 자동화 도구의 장점에 대한 설명으로 가장 옳지 않은 것은?

- ① 테스트 자동화 도구는 테스트를 쉽고 효율적으로 수행할 수 있도록 도와준다.
- ② 테스트 자동화 도구는 테스트의 정확성을 유지하면서 테스트의 품질을 향상시킬 수 있도록 도와준다.
- ③ 테스트 자동화 도구의 사용 방법에 대한 교육 및 학습이 필요하다.
- ④ 테스트 결과에 대한 객관적인 평가 기준을 제공한다.

4. 애플리케이션 테스트 관리- SEC_08(테스트 자동화 도구) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(테스트 자동화 도구)

5. 다음 중 테스트 자동화에 대한 설명으로 가장 옳지 않은 것은?

- ① 테스트 자동화는 테스트 준비, 구현, 수행, 분석 등을 스크립트 형태로 구현한다.
- ② 테스트 자동화 도구를 이용하면 통계 작업과 그래프 등 다양한 표시 형태로 테스트 결과를 표시할 수 있다.
- ③ 테스트 엔지니어는 프로젝트를 완전히 이해한 후 테스트를 수행해야 하므로 프로젝트가 완료된 후 투입해야 한다.
- ④ 테스트 자동화 도구는 다중 플랫폼 호환성, 소프트웨어 구성, 기본 테스트 등 향상된 테스트 품질을 보장한다.

4. 애플리케이션 테스트 관리-SEC_09(결함 관리) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(결함 관리)

1. 소프트웨어 개발 활동을 수행함에 있어서 시스템이 고장 (Failure)을 일으키게 하며, 오류(Error)가 있는 경우 발생하는 것은?

- ① Fault ② Testcase
- ③ Mistake ④ Inspection

2. 다음 중 결함에 관한 설명으로 틀린 것은?

- ① 결함은 사용자의 기대 결과와 실제 소프트웨어를 실행했을 때의 결과 간의 차이를 의미한다.
- ② 결함 심각도는 우선순위에 따라 High, Medium, Low로 분류하기도 한다.
- ③ 결함 에이징은 테스트 진행 시간에 따른 결함 수를 측정한다.
- ④ 결함의 우선순위는 결함의 중요도와 심각도에 따라 결정된다.

3. 결함 관리 프로세스 중 “전체 프로세스에 대한 결함 관리 일정, 인력, 업무 프로세스 등을 확보하여 계획을 수립하는 단계”는 어떤 단계에 속하는가?

- ① 결함 기록
- ② 결함 관리 계획
- ③ 결함 수정
- ④ 결함 검토

4. 테스트에서 발견되는 결함을 유형별로 분류할 때 속하지 않는 것은?

- ① 추세 결함 ② 기능 결함
- ③ GUI 결함 ④ 시스템 결함

4. 애플리케이션 테스트 관리-SEC_09(결함 관리) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(결함 관리)

5. 결함 심각도를 우선순위에 따라 분류할 때, "핵심 요구사항 미구현, 장시간 시스템 응답 지연, 시스템 다운 등과 같이 더 이상 프로세스를 진행할 수 없도록 만드는 결함"은 무엇인가?

- ① High ② Medium
- ③ Low ④ Minor

6. 결함 관리 도구의 종류로 틀린 것은?

- ① Case ② Trac
- ③ Mantis ④ Redmine

4. 애플리케이션 테스트 관리- SEC_10(애플리케이션 성능 분석) 출제 예상 문제

출제 예상 문제(애플리케이션 성능 분석)

1. 다음 중 애플리케이션의 성능을 측정하기 위한 지표가 아닌 것은?

- ① 신뢰도(Reliability)
- ② 처리량(Throughput)
- ③ 경과 시간(Turn Around Time)
- ④ 응답 시간(Response Time)

2. 다음 중 애플리케이션의 성능을 저하하는 원인이 아닌 것은?

- ① DB에 필요 이상의 많은 데이터를 요청하면 애플리케이션의 성능 저하 현상이 발생할 수 있다.
- ② 애플리케이션의 성능 저하 현상은 커넥션 풀의 크기를 크게 하면 발생하지 않지만 작게 하면 발생할 수 있다.
- ③ 대량의 파일을 업로드 하거나 다운로드 하여 처리 시간이 길어진 경우 애플리케이션의 성능 저하 현상이 발생할 수 있다.
- ④ 애플리케이션에 연결된 데이터베이스의 락(DB Lock)으로 인해 애플리케이션의 성능 저하 현상이 발생할 수 있다.

3. 다음 중 애플리케이션의 성능을 측정하는 지표들에 대한 설명으로 틀린 것은?

- ① 처리량(Throughput) : 일정 시간 내에 애플리케이션이 처리하는 일의 양
- ② 응답 시간(Response Time) : 애플리케이션에 요청을 전달한 시간 부터 응답이 도착할 때까지 걸린 시간
- ③ 경과 시간(Turn Around Time): 애플리케이션이 작업을 처리하기 시작한 시간부터 처리가 완료될 때까지 걸린 시간
- ④ 자원 사용률(Resource Usage) : 애플리케이션이 의뢰한 작업을 처리하는 동안의 CPU, 메모리, 네트워크 등 의 사용량

4. 소프트웨어 설계 시 구축된 플랫폼의 성능 특성 분석에 사용 되는 측정 항목이 아닌 것은?

- ① 응답시간(Response Time)
- ② 가용성(Availability)
- ③ 사용률(Utilization)
- ④ 서버 튜닝(Server Tuning)

4. 애플리케이션 테스트 관리- SEC_10(애플리케이션 성능 분석) 출제 예상 문제

출제 예상 문제(애플리케이션 성능 분석)

5. 다음 중 성능 테스트 도구가 아닌 것은?

- ① Scouter
- ② JMeter
- ③ LoadUI
- ④ OpenSTA

6. 다음 중 HTTP와 HTTPS의 가장 큰 차이점은 무엇인가?

- ① 프로토콜 ② 암호화
- ③ 클라이언트/서버 통신 ④ 웹 환경

4. 애플리케이션 테스트 관리-SEC_11(복잡도) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(복잡도)

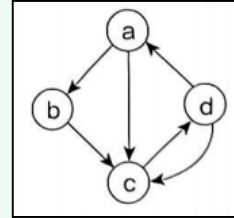
1. 알고리즘 시간 복잡도 $O(1)$ 이 의미하는 것은?

- ① 컴퓨터 처리가 불가
- ② 알고리즘 입력 데이터 수가 한 개
- ③ 알고리즘 수행시간이 입력 데이터 수와 관계 없이 일정
- ④ 알고리즘 길이가 입력 데이터보다 작음

2. 정렬된 N 개의 데이터를 처리하는데 $O(N \log N)$ 의 시간이 소요되는 정렬 알고리즘은?

- ① 선택 정렬 ② 삽입 정렬
- ③ 버블 정렬 ④ 합병 정렬

3. 제어 흐름 그래프가 다음과 같을 때 McCabe의 Cyclomatic 수는 얼마인가?



- ① 3 ② 4
- ③ 5 ④ 6

4. n 개의 원소를 정렬하는 방법 중 평균 수행시간 복잡도와 최악 수행시간 복잡도가 모두 $O(n \log n)$ 인 정렬은?

- ① 삽입 정렬 ② 힙 정렬
- ③ 버블 정렬 ④ 선택 정렬

4. 애플리케이션 테스트 관리-SEC_11(복잡도) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(복잡도)

5. 빅오 표기법에서 효율성이 최악인 것은?

- ① $O(1)$ ② $O(n)$
- ③ $O(\log n)$ ④ $O(n^2)$

6. 시간 복잡도에 대한 설명으로 틀린 것은?

- ① 알고리즘을 수행하기 위해 프로세스가 수행하는 연산 횟수를 수치화한 것을 의미한다.
- ② 시간 복잡도가 높을수록 알고리즘의 실행시간이 짧고, 낮을수록 실행시간이 길어진다.
- ③ 알고리즘의 실행시간이 하드웨어적 성능이나 프로그래밍 언어의 종류에 따라 달라진다.
- ④ 시간이 아닌 명령어의 실행 횟수를 표기한다.

4. 애플리케이션 테스트 관리-SEC_12(애플리케이션 성능 개선) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(애플리케이션 성능 개선)

1. 소스 코드품질 분석 도구 중 정적 분석 도구가 아닌 것은?

- ① pmd ② checkstyle
- ③ valMeter ④ cppcheck

2. 외계인 코드(Alien Code)에 대한 설명으로 옳은 것은?

- ① 프로그램의 로직이 복잡하여 이해하기 어려운 프로그램을 의미한다.
- ② 아주 오래되거나 참고문서 또는 개발자가 없어 유지보수 작업이 어려운 프로그램을 의미한다.
- ③ 오류가 없어 디버깅 과정이 필요 없는 프로그램을 의미한다.
- ④ 사용자가 직접 작성한 프로그램을 의미한다.

3. 다음 중 클린 코드 작성 원칙으로 거리가 먼 것은?

- ① 누구든지 쉽게 이해하는 코드 작성
- ② 중복이 최대화된 코드 작성
- ③ 다른 모듈에 미치는 영향 최소화
- ④ 단순, 명료한 코드 작성

4. 다음에서 설명하는 클린 코드 작성 원칙은?

- 한 번에 한 가지 처리만 수행한다.
- 클래스/메소드/함수를 최소 단위로 분리한다.

- ① 다형성 ② 단순성
- ③ 추상화 ④ 의존성

4. 애플리케이션 테스트 관리- SEC_12(애플리케이션 성능 개선) 기출 및 출제 예상 문제

기출 및 출제 예상 문제(애플리케이션 성능 개선)

5. 소스 코드 정적 분석(Static Analysis)에 대한 설명으로 틀린 것은?

- ① 소스 코드를 실행시키지 않고 분석한다.
- ② 코드에 있는 오류나 잠재적인 오류를 찾아내기 위한 활동이다.
- ③ 하드웨어적인 방법으로만 코드 분석이 가능하다.
- ④ 자료 흐름이나 논리 흐름을 분석하여 비정상적인 패턴을 찾을 수 있다.

6. 클린 코드 작성원칙에 대한 설명으로 틀린 것은?

- ① 코드의 중복을 최소화 한다.
- ② 코드가 다른 모듈에 미치는 영향을 최대화하도록 작성한다.
- ③ 누구든지 코드를 쉽게 읽을 수 있도록 작성한다.
- ④ 간단하게 코드를 작성한다.

7. 코드의 간결성을 유지하기 위해 사용되는 지침으로 틀린 것은?

- ① 공백을 이용하여 실행문 그룹과 주석을 명확히 구분한다.
- ② 복잡한 논리식과 산술식은 괄호와 들여쓰기(Indentation)를 통해 명확히 표현한다.
- ③ 빈 줄을 사용하여 선언부와 구현부를 구별한다.
- ④ 한 줄에 최대한 많은 문장을 코딩한다.

8. 아주 오래되거나 참고문서 또는 개발자가 없어 유지보수 작업이 아주 어려운 프로그램을 의미하는 것은?

- ① Title Code ② Source Code
- ③ Object Code ④ Alien Code