

[실전 모의고사 3회]

1	2	3	4	5	6	7	8	9	10
3	4	2	1	1	4	1	1	4	1
11	12	13	14	15	16	17	18	19	20
2	1	2	2	1	3	4	4	3	1
21	22	23	24	25	26	27	28	29	30
1	1	2	2	4	3	3	4	1	3
31	32	33	34	35	36	37	38	39	40
2	3	1	1	4	3	3	4	4	1
41	42	43	44	45	46	47	48	49	50
2	2	3	3	3	2	4	3	3	2
51	52	53	54	55	56	57	58	59	60
2	1	2	3	4	4	2	3	1	4
61	62	63	64	65	66	67	68	69	70
1	4	4	3	4	2	2	2	4	2
71	72	73	74	75	76	77	78	79	80
1	4	1	2	1	3	3	3	1	2
81	82	83	84	85	86	87	88	89	90
1	3	2	3	4	4	2	4	1	3
91	92	93	94	95	96	97	98	99	100
2	4	4	3	1	4	1	2	3	2

1. ③  
운영체제는 하드웨어와 소프트웨어 자원을 관리하고 컴퓨터 프로그램을 위한 공통 서비스를 제공하는 소프트웨어를 의미한다.

2. ④

OSI 7 Layer	PDU
응용 계층	Data
표현 계층	Data
세션 계층	Data
전송 계층	Segment
네트워크계층	Packet
데이터링크 계층	Frame
물리 계층	Bits

3. ②  
설문에 대한 설명이다.

4. ①  
① 0..\* 또는 \*: 0을 포함한 무한개의 인스턴스

5. ①  
요구사항 관리 도구에 대한 설명이다.

6. ④  
④ HCI(Human Computer Interaction)은 인간과 컴퓨터간의 상호작용에 관한 연구로, 사람이 시스템을 보다 편리하고 안전하게 사용할 수 있도록 연구하고 개발하는 학문이다.

7. ①  
목업(Mockup)은 디자인, 사용 방법 설명, 평가 등을 위해 와이어프레임보다 좀 더 실제 화면과 유사하게 만든 정적인 형태의 모형이다. 시각적으로만 구성 요소를 배치하는 것으로 실제로 구현되지 않는다.

8. ①  
- 호환성: 소프트웨어 제품이 비슷한 환경에서 연관된 표준, 관례 및 규정을 준수하는 능력이다. 변경성이 결함 제거 또는 환경 변화에 따른 수정에 필요한 노력 정도에 따른 특성

9. ④  
UI 설계서 표지는 다른 문서와 혼동되지 않도록 프로젝트명 또는 시스템명을 포함시켜 작성한다.

10. ①  
UI 설계 도구 유형은 UI 구현 절차에 따라 기획단계에서 화면 UI 설계 및 인터랙션 적용에 적합한 도구와 설계 작업 이후 이미 설계된 이미지의 인터랙션 적용에 적합한 도구로 나눌 수 있다.

11. ②  
공통 모듈 명세 작성 원칙에는 정확성, 명확성, 완전성, 일관성, 추적성이 있다.

12. ①  
모듈 결합도에는 자료 결합, 스탬프 결합, 제어 결합, 외부 결합, 공통 결합, 내용 결합이 있다.

13. ②  
② 행위모델-컴포넌트 행위-계산, 실행제어, 상호작용

14. ②  
② 컴포넌트는 처리, 저장, 전달을 수행하는 소프트웨어 시스템의 구성요소이며 커넥터는 컴포넌트를 연결하는 연결자를 말한다.

15. ①  
소프트웨어 아키텍처 4+1 View의 구현 관점(Implementation View)은 독립적으로 실행되는 컴포넌트와 이들 간 관계를 정의한다.

16. ③  
다형성(polymorphism)에 대한 설명이다.

17. ④  
\* 오버라이딩(overriding) 조건  
- 자식 클래스에서 오버라이딩하는 메서드는 상위 클래스의 메서드와  
ㄱ. 이름이 같아야 한다.  
ㄴ. 매개변수가 같아야 한다.

ㄷ. 리턴 타입이 같아야 한다.

18. ④

④ 연계 방식은 직접 연계 방식과 간접 연계 방식으로 분류할 수 있다.

19. ③

③ 인스펙션에 대한 설명이다.

20. ①

동기(Sync) 거래에 대한 설명이다.

21. ①

preorder: 부모 노드 → 왼쪽 자식 노드 → 오른쪽 자식 노드

22. ①

- \* SQL(Structured Query Language) 특징
  - 관계대수와 관계해석을 기초로 한 고급 데이터 언어
  - 이해하기 쉬운 형태
  - 대화식 질의어로 사용 가능
  - 데이터 정의, 조작, 제어 기능 제공
  - COBOL, C, PASCAL 등의 언어에 삽입 → 내장 SQL
  - DBMS에서 사용되는 비절차적 대화형 Language

23. ②

② 매개 변수를 받을 수 있고, 반복적으로 사용할 수 있는 Object이다

24. ②

- ① START: SQL script를 실행
- ③ GET: 파일의 내용을 버퍼로 읽음
- ④ SPOOL: 조회결과를 파일로 저장

25. ④

- \* SQL 코드 인스펙션 대상
  - 사용되지 않는 변수
  - 사용되지 않는 서브쿼리
  - Null 값과 비교
  - 과거의 데이터 타입을 사용

26. ③

공통 모듈에 대한 설명이다.

27. ③

- ③ <header>에 대한 설명이다.
  - <footer>: 섹션의 꼬리말을 나타내고 저자나 저작권 등을 포함한다.

28. ④

- ④ 구글 캘린더에 대한 설명이다.
  - 구글 드라이브: 팀원 간 또는 고객과 문서를 공유하거나 공동

작업을 할 수 있는 온라인 도구

29. ①

① 추가적으로 다양한 이기종 연동을 고려한다.

30. ③

③ 애플리케이션 모니터링 도구의 기능이다.

31. ②

② 예측 단계에 대한 설명이다.

\* SPICE 모델의 프로세스 수행 능력 6단계

0. 불완전(incomplete) 단계 : 미구현 또는 미달성

- 프로세스가 구현되지 않음. - 프로세스가 목적을 달성하지 못함.

1. 수행(performed) 단계 : 프로세스 수행 및 목적 달성

- 프로세스를 수행하고 목적을 달성함. - 프로세스가 정의된 산출물을 생산함.

2. 관리(managed) 단계 : 프로세스 수행 계획 및 관리

- 정의된 자원의 한도 내에서 그 프로세스가 작업 산출물을 인도함.

3. 확립(established) 단계 : 정의된 표준 프로세스 사용

- 소프트웨어 공학 원칙을 기반으로 정의된 프로세스를 수행함.

4. 예측(predictable) 단계 : 프로세스의 정량적 이해 및 통제

- 프로세스가 목적 달성을 위해 통제됨.

- 프로세스가 양적 측정을 통해 일관되게 수행됨.

5. 최적화(optimizing) 단계 : 프로세스를 지속적으로 개선

- 프로세스 수행을 최적화함.

- 지속적 개선을 통해 업무 목적을 만족시킴.

32. ③

③ 동일한 프로젝트 파일에 대해서 여러 개발자가 동시 개발 할 수 있어야 한다.

33. ①

① 안드로이드 환경에 적합한 도구, 그라들(Gradle)에 대한 설명이다.

34. ①

\* 소프트웨어 테스트 프로세스

- 계획 수립 단계: 테스트 계획 수립 프로세스, 테스트 시나리오 및 케이스 관리 프로세스

- 테스트 수행 단계: 테스트 환경 관리 프로세스, 테스트 수행 관리 프로세스, 코드 커버리지 관리 프로세스, 형상 및 배포 관리 프로세스

- 테스트 결과 단계: 테스트 결과 검토 프로세스

35. ④

일관성 검사(Consistent) 오라클에 대한 설명이다.

36. ③

\* 결함 관리 프로세스

## (1) 에러(결함) 발견

: 요구사항 분석, 설계, 테스트 실행 중 에러가 발견될 경우, 테스트 전문가와 프로젝트팀과 논의한다.

## (2) 에러(결함) 등록

: 결함 관리 대장에 발견된 에러를 등록한다.

## (3) 에러(결함) 분석

: 등록된 에러가 단순 에러인지 아니면 실제 결함인지 분석한다.

## (4) 결함 확정

: 등록된 에러가 실제 결함으로 확정될 경우, 결함 확정 상태로 설정한다.

## (5) 결함 할당

: 결함을 해결할 담당자를 지정하여 결함을 할당하고 결함 할당 상태로 설정한다.

## (6) 결함 조치

: 결함에 대해 수정 활동을 수행하고, 수정이 완료된 경우 결함 조치 상태로 설정한다.

## (7) 결함 조치 검토 및 승인

: 수정이 완료된 결함에 대해 확인 테스트를 수행하고, 정상적으로 결함 조치가 완료된 경우 결함 조치 완료 상태로 설정한다.

## 37. ③

③ 상위 컴포넌트 개발이 안 되었을 경우 더미 컴포넌트인 드라이버(Driver)를 작성한다.

## 38. ④

④ 통합 테스트의 목적과 범위는 테스트 수행에 대한 정확한 인식과 범위를 정확하게 식별하기 위함이다.

## 39. ④

④ View에 대한 설명이다.

## \* 구성요소

- View: 모델이 처리한 결과를 화면에 보여주는 요소 (정보 표시, 사용자에게 보여줌)
- Controller: 사용자 요청을 받아 그 요청을 처리할 모델 호출, 모델이 처리 후 결과를 뷰에게 전달 (Model 과 View 연결, 사용자와의 상호 작용을 관리)
- Model: 사용자 요청을 처리해 사용자에게 출력할 데이터를 만드는 요소 (내부 구성요소, 핵심 기능, 도메인의 지식을 저장보관)

## 40. ①

웹 서비스(Web Service)에 대한 설명이다.

## 41. ②

② 부분 완료(partially committed)-트랜잭션의 마지막 명령문을 성공적으로 실행한 직후의 상태

## 42. ②

오손 데이터 읽기(Dirty Read)에 대한 설명이다.

## 43. ③

미디어 회복(Media Recovery) 기법에 대한 설명이다.

## 44. ③

## \* 해시 함수 종류

- 계수 분석(Digit Analysis) - 제산법(Division)
- 제곱법(Mid-square) - 폴딩(Folding)
- 기수변환(Radix Transformation)

## 45. ③

Index Full Scan 에 대한 설명이다.

## 46. ②

② ROLLUP() 함수에 비해 계산이 많다.

## 47. ④

①, ②, ③ 비율관련 함수

④ 행 순서관련 함수

## 48. ③

③ 비즈니스 로직을 캡슐화하여 데이터 관리를 단순화할 수 있다.

## 49. ③

③ 계층적 데이터 모델 - 데이터베이스를 트리 구조로 표현한다.

## 50. ②

② 타원 - 속성(attribute)을 나타낸다.

- 사각형: 개체(Entity)를 나타낸다.

## 51. ②

- 차수 (Degree): 속성들의 수

- 카디널리티 (cardinality) : 튜플들의 수

## 52. ①

## \* 관계 대수 종류

- 순수 관계 연산자 : SELECT, PROJECT, JOIN, DIVISION
- 일반 집합 연산자 : 합집합, 교집합, 차집합, 카티션프로덕트(곱하기 연산)

## 53. ②

- 외래키: 어떤 R에서 다른 R을 참조할 때 참조 기준이 되는 속성으로 참조하고자 하는 R의 기본 키와 동일하다.

## 54. ③

단일 값 속성(single-valued attribute)에 대한 설명이다.

## 55. ④

## \* 접근통제 유형

- 임의 접근 통제(DAC, Discretionary Access control)

- 강제 접근 통제(MAC, Mandatory Access Control)
- 역할기반 접근 통제(RBAC, Role Based Access Control)

56. ④

\* 물리 데이터 모델 품질 기준

- 정확성
- 완전성
- 준거성
- 최신성
- 일관성
- 활용성

57. ②

② 조건절 컬럼에는 함수는 사용하지 않는다.

58. ③

- 데이터 전환 데이터 흐름도

: Source DB→추출→정제→변환→적재→검증→Target DB

59. ①

① 순차 처리와 랜덤 처리가 모두 가능하므로, 목적에 따라 융통성 있게 처리한다.

60. ④

④ 전환 인프라 구축에 대한 설명이다.

\* 데이터 전환 테스트 및 검증 단계

데이터 전환 개발이 완료되면 전환 테스트를 진행하고 계획된 대로 전환이 되었는지를 검증한다.

(가) 전환 검증 설계

: 원천 데이터의 추출 및 변환 후 목적 데이터베이스에 전환하기 위한 검증 방법과 단계를 설계한다.

(나) 추출 검증

: 원천의 데이터베이스, 파일, 문서로부터 전환하기 위한 데이터를 정확히 추출했는 지를 검증한다.

(다) 변환 결과 검증

: 추출된 원천 데이터를 목적 데이터베이스에 전환하기 위해 변환 작업에 대한 검증을 수행한다.

(라) 최종 전환 설계

: 검증 과정에서 발견된 수정 사항을 반영하여 최종 전환 설계를 완성한다.

(마) 전환 인프라 구축

: 실제 데이터 전환을 위한 인프라(서버, DB 등)를 구성한다.

(바) n차 전환 검증

: 반복적인 데이터 전환 테스트를 통해 데이터 정합성을 확보한다.

61. ①

CSS(Cascading Style Sheets)에 대한 설명이다.

62. ④

\* 개발언어 선정 기준

- 적정성
- 효율성
- 이식성
- 친밀성
- 범용성

63. ④

④ 소프트웨어 라이브러리의 결점은 성능 향상이나 출력 형식을 바꾸고자 할 때 세부사항을 조절할 수 없고 라이브러리를 취득·학습·설정하는 데에 시간과 비용이 든다.

64. ③

- 매크로 처리 과정

: 매크로 정의 인식→매크로 정의 저장→매크로 호출 인식→매크로 호출 확장

65. ④

④ XQuery 삽입에 대한 설명이다.

- 크로스사이트 요청 위조: 검증되지 않은 외부 입력 값에 의해 브라우저에서 악의적인 코드가 실행되어 공격자가 원하는 요청(Request)이 다른 사용자(관리자 등)의 권한으로 서버로 전송되는 보안악점

66. ②

② Quartz 스케줄러의 주요 구성 요소이다.

\* 스프링 배치의 주요 특징

- IoC(Inversion Of Control, 제어의 역행): 메소드나 객체의 호출 작업을 개발자가 결정하는 것이 아니라, 외부에서 결정되는 것. 개발자는 프레임워크에 필요한 부품을 개발하고, 조립하는 방식의 개발을 하게 된다. 이렇게 조립된 코드의 최종 호출은 프레임워크 내부에서 결정된 대로 이뤄지게 되는데, 이런 현상을 '제어의 역행'이라고 한다.

- DI(Dependency Injection, 의존성 주입): 제어의 역행이 일어날 때 스프링이 내부에 있는 객체(스프링에서는 빈(Bein)이라는 용어를 사용)들 간의 관계를 관리할 때 사용하는 기법

- AOP(Aspects Oriented Programming, 관점 지향 프로그래밍): 횡단 관심사를 모듈로 분리하는 프로그래밍의 패러다임

67. ②

프로그래밍 언어에 대한 설명이다.

68. ②

\* 논리 연산자 종류

- &&: 두 개의 논릿값이 모두 참이면 참을 반환하고, 그렇지 않으면 거짓을 반환한다.

- ||: 두 개의 논릿값 중 하나가 참이면 참을 반환하고, 그렇지 않으면 거짓을 반환한다.

- !: 부정(not)

69. ④

④ 과학 계산이나 하드웨어 제어에 주로 사용한다.

70. ②

다형성(Polymorphism)에 대한 설명이다.

71. ①

① 객체지향 프로그래밍 언어이다.



저장된 로그를 분석해 시작을 나타내는 Start는 있지만 완료를 나타내는 Commit이 없는 트랜잭션들이 작업한 변경 내용들을 모두 취소한다.

#### 90. ③

\* 접근통제방법

- 접근제어방법: 사용자가 데이터베이스에 직접적인 경로를 통해서 접근할 때 이를 통제하는 방법
- 정보흐름제어: 사용자가 접근할 수 있는 데이터들 사이에서 정보의 분배, 흐름이 발생할 때 권한이 부여되지 않은 데이터 사이에서 부당한 데이터 전달을 통제하는 방법
- 추론제어: 데이터에 직접적인 접근으로 정보를 얻기보다는 간접적인 수단으로 정보를 부당하게 접근하지 못하도록 통제하는 방법

#### 91. ②

② Seven Touchpoints에 대한 설명이다.

#### 92. ④

위험한 형식 파일 업로드에 대한 설명이다.

#### 93. ④

균형 검사(balance check)에 대한 설명이다.

#### 94. ③

캡슐화에 대한 설명이다.

#### 95. ①

파밍 공격에 대한 설명이다.

#### 96. ④

LAND 공격 (Local Area Network Denial Attack)에 대한 설명이다.

#### 97. ①

① HTTP에 대한 설명이다.

\* 웹서버의 기능

- HTTP: HTTP의 Request와 Response 처리를 수행한다.
- 통신 기록: 처리 기능을 Log화 시켜 추적성을 보장한다.
- 인증: 허용된 사용자나 관리자만을 위한 인증을 제공한다.
- HTTPS 지원: HTTP의 보안이 강화된 버전으로 주로 전자상거래에서 사용된다.
- 정적 콘텐츠 관리: 동적 콘텐츠보다 빠른 동작과 캐시에 사용되어 신속한 서비스에 활용된다.
- 가상 호스팅: 기본적 호스트인 main host를 제외한 나머지 host를 말한다.
- 대역폭 스로틀링: 네트워크의 과부하를 막아 QoS를 보장하는데 사용된다.

#### 98. ②

② 보안 프레임워크에 대한 설명이다.

#### 99. ③

\* 시스템별 로그 위치

- /usr/adm: 초기 유닉스, BSD 계열
- /var/adm: 최근 유닉스, SVR 계열
- /var/log: 일부 BSD 계열: BSD, FreeBSD, SUN Solaris, Linux
- /var/run: 일부 Linux 계열

#### 100. ②

셸 모니터에 대한 설명이다.

\* 웹/애플리케이션/서버 보안 솔루션

- 웹방화벽: SQL Injection, XSS 등과 같은 웹 공격을 탐지·차단하며 정보 유출 방지, 부정 로그인 방지, 웹 사이트 위·변조방지를 위한 보안 솔루션
- 셸 모니터: 서버 파일 시스템의 변조를 실시간 삼입되는 웹 서버 악성 코드 및 악성 코드 유포지를 검사하고 관리자에게 통보하여 검역 조치하는 보안 솔루션
- 서버 백신: 운영서버의 정상적인 프로그램이나 데이터를 파괴하도록 개발된 악성 프로그램을 치료하는 프로그램