

실전 모의고사

[실전 모의고사 1회]

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

1. ①

* 플랫폼의 기능

- 소프트웨어 개발 및 운영비용이 감소한다.
- 동일한 플랫폼 간 커뮤니티를 형성하여 네트워크 효과를 유발한다.
- 소프트웨어 개발의 생산성을 향상시킨다.

2. ④

전송 계층(Transport Layer)에 대한 설명이다.

3. ④

- 정적 모델링: Class Diagram, Object Diagram, Component Diagram, Deployment Diagram
- 동적 모델링: State Diagram, Sequence Diagram, Collaboration Diagram, Activity Diagram

4. ③

SCRUM 방법론에 대한 설명이다.

5. ③

경계 클래스 유형에는 UI 클래스, SI 클래스, DI 클래스가 있다.

6. ①

① 직관성: 누구나 쉽게 이해하고 사용할 수 있어야 한다.

7. ④

와이어프레임에 대한 설명이다. 와이어프레임 도구에는 손그림, 파워포인트, 키노트, 스케치, 일러스트, 포토샵이 있다.

8. ①

ISO/IEC 9126 품질 요구사항 중 기능성(Functionality)에 대한 설명이다.

9. ①

UI 프로토타입에 대한 설명이다.

10. ①

* UI 설계 단계

1. 문제 정의(시스템의 목적 기술, 문제 정의)
2. 사용자 모델 정의(사용자 특성 결정)
3. 작업 분석(문제 정제, 사용자의 특성, 특징 세분화)
4. 컴퓨터 오브젝트 및 기능 정의(어떤 컴퓨터 UI를 통해 표현할 것인지 정의)
5. 사용자 인터페이스 정의(상호작용하는 오브젝트 선택, 시스템 상태 명확히 정의)
6. 디자인 평가

11. ③

기능적 응집-순차적 응집-교환적 응집-절차적 응집-시간적 응집-논리적 응집-우연적 응집

12. ③

③ 입력/출력 데이터는 정적(Static)요소에 속한다.

13. ④

아키텍처 구축에서 요구 사항 분석 단계는 소프트웨어 개발의 요구 사항 분석 단계와 같다. 다만 품질 속성과 같은 비기능적인 요구 사항에 더 많은 관심을 둔다. 이 단계에서는 다음과 같은 내용을 다룬다.

- 요구 사항 취득, 식별, 명세, 분류, 검증
- 기능적/비기능적 요구 사항 분류 및 명세

14. ④

정보은닉(Information Hiding)에 대한 설명이다.

15. ①

① Adapter Pattern은 구조 패턴에 속한다.

16. ②

인터페이스 요구사항은 크게 시스템 인터페이스와 사용자 인터페이스(UI) 요구사항으로 나누어 작성합니다.

17. ①

송신 시스템에 대한 설명이다.

18. ②

② 비기능적 요구사항 중 보안에 관련된 사항이다.

* 기능적 요구사항

- 시스템이 제공해야 하는 기능들을 기술하는 것이다.
- 시스템이 특정 입력에 어떻게 반응하는지와 시스템이 특정 상황에서 어떻게 행동하는지를 기술한다.

* 비기능적 요구사항

- 시스템의 전체적 품질이나 기능적 요구사항의 구현 시 고려해야하는 제약 사항이다.
 - 소프트웨어뿐만 아니라 시스템 특성, 품질에 대한 요구, 시스템에 대한 제약사항들이다.
- 19. ④**
- 요구사항 개발 절차
- : 요구사항 도출→요구사항 분석→요구사항 명세→요구사항 확인
- 20. ①**
- ① 연계 데이터 오류의 경우 데이터를 보정하고 재전송한다.
- 21. ②**
- 스택(Stack)에 대한 설명이다.
- 22. ②**
- ② 액세스 효율 향상을 위한 물리적 저장 방법이다.
- 23. ②**
- 선언부(Declare)에 대한 설명이다.
- 24. ①**
- SQL 처리 단계
1. 구문분석: 사용자가 요청한 SQL문이 실행계획이 있는지 확인한다.
 2. 실행: 실행 계획에 따라서 메모리 영역의 데이터베이스 버퍼 캐시 영역에 해당 데이터가 있는지 확인한다.
 3. 추출: 서버 프로세스는 데이터버퍼 캐시영역에서 관련 테이블 데이터를 읽어서 클라이언트로 보내준다.
- 25. ③**
- * 클린코드의 특징
 - 가독성이 뛰어나다.
 - 간단하고 작다.
 - 의존성을 최대한 줄였다.
 - 코드의 의도와 목적이 명확하다.
 - 코드가 타인에 의해 변경이 용이하다.
 - 코드에 중복이 없거나 적다.
 - 개체가 한 가지 작업만 수행한다.
- 26. ③**
- 조건 커버리지(Condition Coverage)에 대한 설명이다.
- 27. ④**
- ④ 협업도구의 기능이다.
- * IDE 도구 기능
 - 개발 환경 지원
 - 컴파일 및 디버깅 기능 제공
 - 외부 연계모듈과 통합 기능 제공

28. ②

② 레드마인(Redmine)은 프로젝트 관리 도구이다.

29. ③

③ 애플리케이션 패키징은 개발자가 아닌 사용자 중심으로 진행된다.

30. ②

컨텐츠 제공자(DRM Server)의 구성요소 중 패키지에 대한 설명이다.

31. ③

* ISO/IEC 12207 프로세스 구분

- 기본 생명 주기 프로세스: 획득(계약 준비), 공급(계약), 개발(S/W 구현), 운영, 유지보수 프로세스
- 지원 생명 주기 프로세스: 품질 보증, 검증, 확인, 활동 검토, 감사, 문서화, 형상 관리, 문제 해결 프로세스
- 조직 생명 주기 프로세스: 관리, 기반 구조, 개선, 훈련 프로세스

32. ③

CI(Continuous Integration, 지속적인 통합)에 대한 설명이다.

33. ②

통합 테스트에 대한 설명이다.

34. ④

- 화이트 박스 테스트 종류 : 기초 경로 검사(Basic Path Testing, McCabe 제안), 조건 검사(Condition Testing), 루프 검사(Loop Testing), 데이터 흐름 검사(Data Flow Testing)
- 블랙 박스 테스트 종류 : 동치분할검사(Equivalence Partitioning), 경계값 분석(Boundary Value Analysis), 원인-효과 그래프 검사(Cause-Effect Graphing Testing), 오류 예측검사(Fault Based Testing), 비교검사(Comparison Testing)

35. ①

테스트 하네스(Test Harness)에 대한 설명이다.

36. ①

MVC 구성요소 중 View에 대한 설명이다.

37. ④

* 인터페이스 기능 확인 방법

- 상세 인터페이스 기능은 인터페이스 설계서(정의서)를 보고 확인한다.
- 시스템 정적/동적 기능분석을 통하여 내/외부 인터페이스 기능을 확인한다.
- 인터페이스 되는 데이터 유형, 값의 범위, 예외처리 규칙을 확인한다.

38. ①

① Json, DB, XML 등 다양한 형태로 인터페이스 모듈의 표현이 가능하다.

39. ①

ESB(Enterprise Service Bus)에 대한 설명이다.

40. ②

* 인터페이스 보안 기능

- 민감정보 가상화 - 인증 보안 수행
- 이상 거래 감지 - 필드 암호/복호화
- 필드 필터링 - 암호화 키 전송
- 파일 암호/복호화 - 체크섬

41. ④

* 데이터베이스 정의

- 통합 데이터(Integrated Data)
- 저장 데이터(Stored Data)
- 운영 데이터(Operational Data)
- 공용 데이터(Shared Data)

42. ③

③ 외부 스키마에 대한 설명이다.

43. ③

테이블에 새로운 레코드를 삽입하고자 하면 INSERT 문을 사용한다.

```
INSERT INTO      테이블명(컬럼명1, ..., 컬럼명n)
VALUES           (값1, ..., 값n)
```

44. ④

- 버킷(Bucket): 동일한 해시 주소를 갖는 레코드(키, 주소)들이 저장될 메모리 블록

45. ④

FROM 성적: 성적 테이블을 읽어

GROUP BY 성명: 학생별

SELECT 성명, AVG(점수); 학생별 점수의 평균

46. ①

트리거(Trigger)에 대한 설명이다.

47. ④

- DDL: CREATE
- DML: SELECT, INSERT, UPDATE

48. ④

* 뷰의 특징

- 저장 장치내에 물리적으로 존재하지 않는다.
- 데이터 보정 등 임시적인 작업을 위한 용도로 사용한다.
- 기본 테이블과 같은 형태의 구조로 조작이 거의 비슷하다.

- 삽입, 내용 갱신 연산이 가능하지만 제한적이다.
- 검색은 일반 테이블과 같다.
- 논리적 독립성을 제공한다.
- 독자적인 인덱스를 가질 수 없다.
- Create를 이용하여 뷰를 생성한다.
- 필요한 데이터만 골라 뷰를 이용하여 처리한다.

49. ①

① 개체 타입은 사각형으로 표시된다.

50. ②

관계 해석에 대한 설명이다.

51. ④

④ 릴레이션을 분리하므로 연산 시간이 증가된다.

- 정규화를 하는 이유는 데이터의 중복을 방지하고 보다 효율적으로 데이터를 저장하기 위함이다. (릴레이션 분리 -> 삽입, 삭제, 갱신 이상의 발생 가능성을 줄이는 것)
- 정규화의 단점 : 연산 시간이 증가됨

52. ③

③ 스키마 정제(정규화)는 논리적 데이터베이스 설계에 관련된 사항이다.

53. ①

- 부분 복제: 통합된 테이블을 본사에 가지고 있으면서 각 지사별로는 지사에 해당된 로우(Row)를 가지고 있는 형태
- 광역 복제: 통합된 테이블을 본사에 가지고 있으면서 각 지사에도 본사와 동일한 데이터를 모두 가지고 있는 형태

54. ④

④ 논리 백업(Logical Backup): 각 오브젝트를 SQL문 등으로 저장하여 백업한다.

55. ③

* 테이블 추가 기법

- 중복테이블 추가 - 통계테이블 추가
- 이력테이블 추가 - 부분테이블 추가

56. ④

④ 활용성에 대한 설명이다.

- 완전성: 데이터 모델의 구성 요소를 정의하는데 있어서 누락을 최소화하고, 요구사항 및 업무 영역 반영에 있어서 누락이 없음을 의미함

57. ②

- 색인 영역(Index): 트랙 색인(track index), 실린더 색인(cylinder index), 마스터 색인(master index)

58. ③

- 데이터 전환 절차

: 데이터 전환 계획서 작성→체크 리스트 작성→데이터 변환 프로그램 개발→수행 결과 검증→결과 보고서 작성

59. ①

① 원천 데이터와 목적 데이터베이스의 속성(Column) 간 대응 관계는 1:1대응보다 N:M 대응 관계가 더 많다.

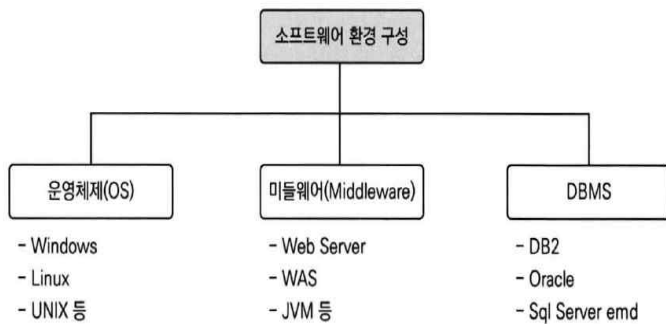
60. ④

④ 오류 데이터를 정당한 데이터로 변경하고 불필요한 데이터를 삭제 또는 초기화한다.

61. ③

③ 프로그램을 개발할 때 가장 많이 사용되는 도구는 구현 도구(Implementation Tool)이다.

62. ①



63. ④

④ 모듈의 독립성(Module Independence): 낮은 결합도와 높은 응집도를 가진다.

64. ②

② 응집도(Cohesion)는 높을수록 좋다.

65. ④

인증(Authentication)에 대한 설명이다.

66. ②

② 유효하지 않은 데이터의 경우도 처리해서 비정상적인 동작 중단이 발생하지 않아야 한다.

* 배치프로그램 필수요소

- 대용량 데이터: 대용량의 데이터를 처리할 수 있어야 한다.
- 자동화: 심각한 오류 상황 외에는 사용자의 개입 없이 동작해야 한다.
- 견고함: 유효하지 않은 데이터의 경우도 처리해서 비정상적인 동작 중단이 발생하지 않아야 한다.
- 안정성: 어떤 문제가 생겼는지, 언제 발생했는지 등을 추적할 수 있어야 한다.
- 성능: 주어진 시간 내에 처리를 완료할 수 있어야 하고, 동시에 동작하고 있는 다른 애플리케이션을 방해하지 말아야 한다.

67. ③

③ 대/소문자를 구분하여 작성한다.

* 변수명 작성 규칙

- 영문자, 숫자, _를 사용
- 첫 글자는 영문자 또는 _ 사용하며, 숫자는 올 수 없다. (1ab → X)
- 공백이나 *, +, -, / 등의 특수문자를 사용할 수 없다. (gisafirst → X → 수정 → gisa_first)
- 대/소문자 구분 (gisafirst와 Gisafirst는 다른 변수명)
- 예약어(ex. char, double, for, while 등)를 변수명으로 사용할 수 없음.
- 변수 선언 시 문장 끝에 반드시 세미콜론(;)을 붙여야 한다.
- 글자 수에 제한이 없다.

68. ③

③ 00000101 ^ 00000111 = 00000010

* 비트 연산자 종류

- &(비트 and): 두 값을 비트로 연산하여 모두 참이면 참(True)을 반환하고, 그렇지 않으면 거짓(False)을 반환한다.
- |(비트 or): 두 값을 비트로 연산하여 하나가 참이면 참(True)을 반환하고, 그렇지 않으면 거짓(False)을 반환한다.
- ^(비트 xor): 두 값을 비트로 연산하여 서로 다르면 참(True)을 반환하고, 그렇지 않으면 거짓(False)을 반환한다.
- ~(비트 not): 각 비트의 부정

69. ①

캡슐화(Encapsulation)에 대한 설명이다.

70. ②

파이썬(Python)에 대한 설명이다.

71. ④

④ 작동 순서를 구체적으로 작성하지 않기 때문에 오류가 적다.

72. ④

④ %c: 문자

* 서식 문자열

- %o: 8진수 정수
- %c: 문자
- %d: 10진수 정수
- %s: 문자열
- %x: 16진수 정수

73. ①

① 외부 라이브러리에 대한 설명이다.

74. ②

- 운영체제가 자원들을 관리하는 과정

시스템 내 모든 자원들의 상태 파악→어떤 프로세스에게 언제, 어떤 자원을 할당할 것인가를 결정하는 분배 정책 수립→자원을 배당하고 운영함으로써 수립된 정책을 수행→프로세스에 배당된 자원 회수

75. ①

① 프로세스를 분할하여 운영체제의 성능을 개선하려는 소프트웨어적 접근 방법이다.

76. ③

③ 페이지 수가 증가하며 페이지 맵핑 테이블이 커지고 맵핑 속도가 느리고 기억 공간 낭비가 발생한다.

77. ①

데이터 링크 계층(Data link layer)에 대한 설명이다.

78. ③

③ 단순한 헤더

★ UDP 주요 특징

- 비연결성
- 비신뢰성
- 순서화되지 않은 Datagram 서비스
- 실시간 응용 및 멀티캐스팅 가능
- 단순한 헤더
- 데이터 전송 단위
- 최대 데이터 크기

79. ④

정적콘텐츠에 대한 설명이다.

80. ③

③ 다양한 사용자에게 다양한 형태의 인터페이스를 제공한다.

★ 데이터베이스의 기능

- 데이터 저장과 개발 및 유지보수 측면에서 중복성 통제
- 다중 사용자 간의 데이터 공유
- 권한 없는 사용자의 데이터접근 통제
- 다양한 사용자에게 다양한 형태의 인터페이스를 제공한다.
- 데이터 사이에 존재하는 복잡한 관련성 표현
- 데이터베이스의 무결성 보장
- 백업과 복구기능 제공

81. ③

★ 소프트웨어 개발 생명주기

: 타당성검토→분석→설계→개발→테스트→운영→유지보수→폐기

82. ③

③ 명세에 대한 유지보수 비용이 감소한다.

83. ②

역공학(Reverse Engineering)에 대한 설명이다.

84. ④

④ 공학(engineering) 프로세스에 대한 설명이다.

- 관리(management) 프로세스: 소프트웨어 생명주기에서 프로젝트 관리자에 의해 사용되는 프로세스로 구성된다.

85. ③

③ 특정 개념들의 추상화를 제공하는 여러 클래스와 컴포넌트로 구성된다.

86. ①

브리지(Bridge)에 대한 설명이다.

87. ②

- 개발자: 정보보호시스템 시장을 확대하고 소비자의 보안요구사항을 이해하여야 한다.
- 사용자: 정보보호시스템 선택의 폭을 확대하고 명확한 보안기능요구사항을 표현하여야 한다.
- 평가자: 표준화된 정보보호시스템을 평가기준으로 한다.

88. ④

임의 접근 통제(DAC)에 대한 설명이다.

89. ①

★ 보안기술 유형

- 인증(Authentication) - 기밀성(Confidentiality)
- 권한(Authorization) - Backup/Recovery
- 접근통제(Access Control) - 감사(Audit)

90. ④

Data Diet에 대한 설명이다.

91. ③

③ 테스트에 대한 설명이다.

- 유지보수: 지속적인 개선을 하며 보안을 패치한다.

92. ②

크로스사이트스크립트(XSS)의 공격 과정이다.

93. ①

피싱(Phishing)에 대한 설명이다.

94. ①

AES(Advanced Encryption Standard)에 대한 설명이다.

95. ②

- 오자 에러: 하나의 문자나 숫자의 오기로 발생한 에러이다.
(표기 예시: 12345 → 12346)

96. ①

랜섬웨어(Ransomware)에 대한 설명이다.

97. ④

④ 인가에 대한 설명이다.

- 책임: 해당 사용자가 무엇을 하였는지를 밝힌다.

98. ①

① 기밀성에 대한 설명이다.

- 가용성: 정보 자산에 대해 적절한 시간에 접근 가능한 것을 의미한다.

99. ①

① 시스템에 이상 징후가 발생했을 때, 시스템 관리자가 가장 먼저 확인해 보는 것이 로그파일이다.

100. ②

IPS(Intrusion Prevention System)에 대한 설명이다.