

1. Microservice Architecture의 특징으로 적합하지 않은 것은?

- 1) 중앙 통제할 수 있는 관리 서비스를 최소화하는 것이 목적이다
- 2) 하나의 서비스는 다른 서비스의 존재를 인식하지 않으며, 오로지 API 통해서만 서비스를 사용하도록 한다
- 3) 각 서비스마다 독립적으로 분리된 데이터 저장소를 갖는 것이 목표이며, 이를 위해서는 공통된 개발 언어를 사용하여 인터페이스를 통일해야 할 필요가 있다
- 4) Cloud Native 서비스가 제공되지 않으면, MSA로 개발된 각 서비스들의 관리가 어려워지기 때문에, Cloud Native와의 연동이 필수이다

2. Cloud Native 서비스가 잘못 연결된 것은?

- 1) Eureka: Naing Server의 역할로써, 서비스의 등록/검색/삭제를 관리한다
- 2) Configuration Server: 중앙에 통제할 수 있는 Configuration Server를 두어, 각 서비스에 필요한 설정 정보를 통제한다
- 3) Load Distribution: MSA의 서비스 간에 데이터 동기화를 위해 사용되는 메시지 서비스로써, Kafka서비스가 대표적이다
- 4) Fault Tolerance: 장애 발생 시 서비스의 사용 중지 및 서비스의 요청을 우회할 수 있는 기능을 제공한다

3. SOA와 MSA와의 차이점을 설명 한 것 중 적절하지 않은 것은?

- 1) 서비스 중심적인 애플리케이션 설계 방식이다
- 2) MSA와 SOA 모두 SOAP을 이용하여 서비스를 하게 되기 때문에, WSDL의 정의가 필수이다
- 3) SOA는 재사용을 통한 비용 절감 효과를 가질 수 있지만, MSA에서는 서비스 간의 공유를 최소화하는 함으로써 고객의 요구사항에 보다 유연하게 대응할 수 있다
- 4) 비즈니스에 보다 집하여 개발하는 방식이 SOA이고, 작은 서비스 하나에 집중하는 방식이 MSA이다

4. REST API 특징으로 적합하지 않은 것은?

- 1) HTTP 프로토콜을 이용한 서비스이다
- 2) 서비스에서 제공하려는 Resource들에 고유한 이름을 부여하여 서비스 하는 것이다
- 3) HTTP Method를 사용하기 때문에, 동일한 URI라 하더라도 다른 Method에 의해 다른 처리가 가능하다
- 4) RESTful API 형식에 맞는 이름을 부여할 때, 동사 위주로 사용하여 붙이도록 한다

5. RESTful을 리소스의 이름을 정하려 할 때 추천되는 내용이 아닌 것은?

- 1) Consumer First
- 2) Include Secure Info in URI
- 3) Use Plurals
- 4) Use Nouns for Resources

6. 12 Factors에 대해 잘못 설명 된 것은?

- 1) Codebase: 각 서비스의 기본이 되는 인터페이스에 대한 정의를 하기 위해 하나의 Base 코드를 갖는 것
- 2) Dependencies: MSA에 필요한 라이브러리 및 각 라이브러리의 버전에 대한 정보를 통하는 것
- 3) Configuration: MSA에서 사용되는 Configuration 정보는 하나의 Repository에 관리 하는 것
- 4) Build, release, Run: 각 서비스는 독립적으로 빌드, 배포, 실행 될 수 있음을 보장하는 것

7. 서비스의 기획, 설계, 개발, 배포 및 운영에 관련 모든 사항을 하나의 팀에서 수행하기 위해 설명되는 용어는?

- 1) SOA
- 2) Agile
- 3) Two Pizza Team
- 4) DevOps

8. MSA를 구현하기 위한 과정 중 데이터의 분리는 가장 어려운 주제이다. 다음 중 각 서비스간의 분리된 데이터를 동기화하기 위한 방법으로 적절한 방법은?

- 1) CUD와 R을 분리하여 도메인 데이터를 관리
- 2) Message Queue 서비스의 이벤트를 통해 데이터를 복사
- 3) 각 서비스에서 사용되는 모든 Database에 동시에 데이터를 저장
- 4) 스케줄러를 통해 일정 주기마다 데이터를 복제

9. Cloud Native의 기술환경으로 설명할 수 있는 4가지 요소를 잘 나타낸 것은?

- 1) CI/CD, DevOps, Microservices, Containers
- 2) Spring Cloud, Netflix OSS, Eureka, Configuration Server
- 3) Eureka, Zuul, Ribbon, Kafka
- 4) Plan, Create, Verify, Release

10. Spring Boot Project 기동 시 제일 먼저 호출되는 클래스는 () 어노테이션이 선언되어 있는 클래스이다. 괄호 안에 들어갈 알맞은 어노테이션은? (주관식)

11. HTTP POST method로 사용자 등록을 요청하려고 한다. 서버로부터 전달 받는 Status Code로 알맞은 것은? (주관식)

12. Spring Boot와 Spring Security를 이용하여 인증처리를 하려고 한다. application.properties 파일에 인증 관련 정보를 등록하려고 할 때 알맞은 필드는?

- 1) security.username, security.password
- 2) spring.login.user.name, spring.login.user.password

- 3) security.user.id, security.user.password
- 4) spring.security.user.name, spring.security.user.password

13. HATEOAS의 특징으로 잘못 설명 된 것은?

- 1) Hypermedia As The Engine Of Application State의 약자이다
- 2) RESTful API를 사용하는 클라이언트가 전적으로 서버와 동적인 상호작용이 가능하도록 하는 것을 말한다
- 3) 클라이언트 요청에 따라, 각 기능마다 URI를 링크 시켜줌으로써, 동적인 API 제공이 가능하다
- 4) Level0 단계의 RESTful를 설계할 때 기본으로 추가되는 기능이다

14. Spring Boot에서 설정 정보 관련 된 내용을 지정하기 위한 클래스에 선언되어야 하는 Annotation은?

- 1) @SpringBootTest
- 2) @SpringBootConfiguration
- 3) @Configuration
- 4) @GlobalConfigure

15. "/users/{user_id}" 와 같은 형식의 RESTful API를 호출하려고 한다. 이때 구현하는 Method의 해당 매개변수에 반드시 선언되어야 하는 Annotation은?

- 1) @PathVariable
- 2) @GetMapping
- 3) @Parameters
- 4) @Value

16. AMQP의 특징으로 바르게 설명 된 것을 2개 선택하시오.

- 1) Spring Cloud Config Server에서 각각의 애플리케이션들의 속성 값을 최신 데이터로 업데이트하기 위해 사용되는 프로토콜로써 RabbitMQ 서버와 연동되어 사용된다.
- 2) 초당 100k 이상의 이벤트 처리가 가능하면 Publisher에 의해 전달 된 메시지는 Topic에 보관 된 후에 Consumer에게 전달된다.
- 3) 메시지 지향 미들웨어를 위한 개방형 표준 프로토콜로써, P2P 방식으로 메시지를 송수신한다.
- 4) Scalar 언어로 개발한 오픈소스 메시지 브로커 프로젝트로써, 비동식 처리를 위한 메시지를 송수신 한다.

17. MSA에서 다양한 언어와 다양한 스토리지를 이용하여 서비스를 개발할 수 있는 것을 나타내는 용어는?

- 1) DevOps
- 2) IntelliJ
- 3) Polyglot
- 4) Spring Boot

18. Spring Cloud에서 각 마이크로서비스들의 등록과 해지에 관련된 작업을 처리하기 위해 Netflix Eureka 를 이용하여 Discovery Service를 구현할 수 있다. Eureka Discovery Service에 등록된 마이크로서비스들의 Gateway 역할과 Load Balancer 역할을 하는 것으로 적절한 것은?

- 1) Spring Cloud Config Server
- 2) Spring Cloud Gateway
- 3) Spring Cloud Eureka
- 4) Spring Cloud Bootstrap

19. 마이크로서비스들 간의 통신을 위해 사용할 수 있는 방법으로 적절하게 연결된 것은?

- 1) OpenFeign, RestTemplate
- 2) RestTemplate, WSDL
- 3) Circuit Breaker, Hystrix
- 4) RabbitMQ, Kafka

20. 사용자 마이크로서비스(User microservice)를 여러 instance에서 실행하기 위해 PORT번호를 달리하려고 한다. application.yml(또는 application.properties) 파일에서 PORT를 어떻게 설정해야 하는가? (주관식)

21. Spring Boot Application에서 환경 설정 정보가 변경되었을 때, 이것을 반영하기 위한 방법으로 적절하지 못한 것은?

- 1) Spring Boot Application을 재기동 하였다.
- 2) 변경된 환경 설정을 반영하기 위해 Spring Cloud Config Server를 재기동하였다.
- 3) Actuator의 refresh를 POST 방식으로 호출하였다.
- 4) Spring Cloud Bus와 RabbitMQ 서버를 연동하여 일괄적으로 업데이트 하도록 설정하였다.

22. Spring Cloud Discovery로써 Netflix Eureka Server를 사용하여고 한다. 다음 설명 중 잘못된 것은?

- 1) 애플리케이션을 Eureka Server에 등록하기 위해서는 Eureka Discovery Client 라이브러리만 추가하면 된다.
- 2) Eureka Dashboard를 통해 등록된 애플리케이션의 이름, IP, Port 번호를 확인할 수 있다.
- 3) Eureka Server에 등록 된 애플리케이션을 삭제하기 위해서는 HTTP DELETE /unregister 를 전달해야 한다.
- 4) Eureka Server를 기동하기 이전에 실행 중인 애플리케이션들에 대해서는 등록할 수 없다.

23. 다음은 RestTemplate을 이용하여 마이크로서비스 간에 통신을 구현하려는 예제의 일부분이

다. 빈칸에 들어갈 적절한 코드를 선택하시오.

```
String orderUrl = String.format(env.getProperty("order_service.url"), userId);
ResponseEntity<List<ResponseOrder>> orderListResponse =
    restTemplate.exchange(orderUrl, code, requestEntity: null,
        new ParameterizedTypeReference<List<ResponseOrder>>() {
        });

List<ResponseOrder> ordersList = orderListResponse.getBody();
```

- 1) HttpMethod.POST
- 2) HttpMethod.GET
- 3) null
- 4) RequestOrder.class

24. Spring Cloud OpenFeign을 이용하여 마이크로서비스 간에 통신을 처리하려고 한다. 다음과 같이 코드를 작성하였을 때, 사각형으로 표시된 부분의 코드가 의미하는 내용을 설명하시오. (주관식)

```
@FeignClient(name="order-service", configuration = FeignErrorDecoder.class)
public interface OrderServiceClient {

    @GetMapping("/order-service/{userId}/orders")
    List<ResponseOrder> getOrders(@PathVariable String userId);

}
```

25. Spring Boot Application에서 Context에 의해서 생성된 Bean을 주입 받기 위한 방법 3가지에 대해 설명하시오. (주관식)