Java 전문가 과정 mongoDB 평가 답지

(메일제목 : mongoDB평가-XXX)

성명 :이원석

|  |  |
| --- | --- |
| 문항 | 답안 |
| 1 | 유연성 : 스키마 선언 없이 필드의 추가 및 삭제가 자유로운 Schema-less 구조이다.  확장성 : 스케일 아웃에 의한 서버 확장이 용이하다.  고성능 : 대용량 데이터를 처리하는 성능이 뛰어나다.  가용성 : 여러 대의 백업 서버 구성이 가능하여 장애 발생 시에도 무중단 서비스가 가능하다 |
| 2 | Scale-UP은 SQL의 시스템 증설방식으로 관계를 가지는 테이블이 수평적으로 더 커지는 시스템 증설 방식이다. 이렇게 되면 기존에 사용하던 DB시스템 보다 고성능의 DB시스템이 필요할 수 있고 이는 덩치가 큰 DB시스템이 된다는 말이며 비용도 많이 증가될 뿐만 아니라 관리가 어렵다  Scale-OUT은 NoSQL의 증설방식으로 고성능의 DB를 갖추는 것이 아닌 여러 DB시스템으로 추가할 수 있는 증설 방식이다. 숫자는 무한대로 늘려갈 수 있으며 이는 Scale-UP이 무제한적으로 늘려갈 수 없는 것과 비교해 큰 장점이다. |
| 3 | JSON은 JavaScriptObjectNotation의 약자로 필드와 값의 쌍으로 이루어진 데이터 오브젝트를전달하기 위해 텍스트를 사용하는 개방형 표준 포맷이고, JavaScript에서 객체 생성 시 사용하는 표현식이다. MongoDB는 BSON으로 인코딩하거나 디코딩하는 것을 자동으로 처리한다. 즉, 데이터 입출력 시에 key-value를 JSON형식으로 저장하여 도큐먼트를 사용하나 데이터베이스 저장 시에는 이진포맷으로 인코딩한 BSON(Binary JSON)형식의 도큐먼트로 변환되어 저장된다. |
| 4 | 1. 컬렉션 (2) 도큐먼트 (3) 필드 |
| 5 | 1, 3 |
| 6 | 도큐먼트 |

|  |  |
| --- | --- |
| 7 | use examdb |
| 8 | db.restaurant.find().sort({"fprice":-1}) |
| 9 | db.restaurant.find({},{"\_id":false, "fname":true, "fprice":true}) |
| 10 | var c = db.restaurant.find({fprice:{$gt:8000}}).count()  print("가격이 8000원 초과인 음식들의 수는 "+c+"입니다") |
| 11 | db.restaurant.find({fprice:{$gte:3000,$lte:10000}},{\_id:true, fname:true, fmaterial:true}) |
| 12 | db.restaurant.find({"fmaterial.1":"B4"},{\_id:false,fname:1}) |
| 13 | db.restaurant.find({fprice:{$gte:10000}},{fcate:true, \_id:false}) |
| 14 | db.restaurant.find({"fcate.item":{$lte:10}},{fcate:true, \_id:false, fname:true}) |
| 15 | db.restaurant.update({fname:"food3"},{$set:{"fprice":13000,"fmaterial.0":"AA3","fcate.type":"TT3"}})  db.restaurant.find().limit(5) |

수고하셨습니다.(^^)