

# MongoDB - Esercitazione

## (Create, Update, Delete)

Per questa esercitazione, utilizzare la collection "classe". Non è necessario crearla (viene creata automaticamente da MongoDB all'inserimento di un documento).

### Create

Creare e inserire nella collezione un documento personale con i seguenti dati

- nome
- cognome
- età
- interessi: un array con almeno 2 valori di tipo stringa
- cittàVisitate: un array con almeno 3 valori tipo oggetti, ciascuno dei quali contiene:
  - nome
  - stato
- un campo a piacere

Ad esempio:

```
{
  nome: "Enrico",
  cognome: "Gallinucci",
  età: 28,
  interessi: ["calcetto", "viaggiare", "serie tv"],
  cittàVisitate: [
    { nome: "Cesena", stato: "Italia" },
    { nome: "Parigi", stato: "Francia" },
    { nome: "San Francisco", stato: "USA" }
  ],
  colorePreferito: "Giallo"
}
```

### Update

NB: ricordarsi di impostare l'opzione **multi: true** nei casi in cui si vogliono aggiornare più documenti.

1. Inserire in TUTTI i documenti un campo "crediti", inizializzato a 0
  - a. **db.classe.update({}, {\$set: {crediti: 0}}, {multi: true})**
2. Incrementare di 1 il valore dei propri crediti
  - a. **db.classe.update({nome: "Enrico"}, {\$inc: {crediti: 1}})**
3. Cercare i documenti in cui è specificato "calcio" tra gli interessi
  - a. **db.classe.find({interessi: "calcio"})**
4. Diminuire di 1 i crediti di tali documenti
  - a. **db.classe.update({interessi: "calcio"}, {\$inc: {crediti: -1}}, {multi: true})**
5. Cercare i documenti in cui è specificato "viaggi" o "viaggiare" tra gli interessi
  - a. **db.classe.find({interessi: {\$in: ["viaggi", "viaggiare"]}})**
6. Aumentare di 1 i crediti di tali documenti

- a. **db.classe.update({interessi: {\$in: ["viaggi", "viaggiare"]}}, {\$inc: {crediti: 1}}, {multi: true})**
- 7. Cercare i documenti in cui l'età è maggiore di 40
  - a. **db.classe.find({età: {\$gt: 40}})**
- 8. Aumentare di 1 i crediti di tali documenti
  - a. **db.classe.update({età: {\$gt: 40}}, {\$inc: {crediti: 1}}, {multi: true})**
- 9. Cercare i documenti in cui esiste un campo che corrisponde al proprio campo a piacere
  - a. **db.classe.find({colorePreferito: {\$exists: 1}})**
- 10. Aumentare di 1 i crediti di tali documenti
  - a. **db.classe.update({prova: {\$exists: 1}}, {\$inc: {crediti: 1}}, {multi: true})**
- 11. Ordinare i documenti per crediti decrescente e verificare chi è il primo
  - a. **db.classe.find().sort({crediti: -1})**

12. Rimuovere il proprio campo a piacere da tutti i documenti
  - a. **db.classe.update({}, {\$unset: {colorePreferito: 1}}, {multi: true})**
13. Usare il \$push per aggiungere un campo "numeri" con valore 4 in tutti i documenti
  - a. **db.classe.update({}, {\$push: {numeri: 4}}, {multi: true})**
14. Usare il \$push per aggiungere i valori 8, 15 e 16 al campo "numeri" in tutti i documenti
  - a. **db.classe.update({}, {\$push: {numeri: {\$each: [8, 15, 16]} }}, {multi: true})**
15. Usare il \$push per aggiungere i valori 23 e 42 solo al proprio documento
  - a. **db.classe.update({nome: "Enrico"}, {\$push: {numeri: {\$each: [23, 42]} } })**
16. Usare l'\$addToSet per aggiungere i valori 23 e 42 a tutti i documenti
  - a. **db.classe.update({}, {\$addToSet: {numeri: {\$each: [23, 42]} } }, {multi: true})**
17. Verificare che il proprio documento non contenga numeri ripetuti
  - a. **db.classe.find({nome: "Enrico"})**
18. Estrarre il primo valore dall'array numeri del proprio documento
  - a. **db.classe.update({nome: "Enrico"}, {\$pop: {numeri: -1}})**
19. Estrarre l'ultimo valore dall'array numeri del proprio documento
  - a. **db.classe.update({nome: "Enrico"}, {\$pop: {numeri: 1}})**
20. Estrarre i valori compresi tra 8 e 23 dall'array numeri di tutti i documenti
  - a. **db.classe.update({}, {\$pull: {numeri: {\$gte: 8, \$lte: 23}} }, {multi: true})**
21. Impostare a 2017 il campo "anno" nel primo elemento dell'array cittàVisitate nel proprio documento
  - a. **db.classe.update({nome: "Enrico"}, {\$set: {"cittàVisitate.0.anno": 2017}})**
22. Impostare a true il campo "sede" nell'elemento dell'array cittàVisitate che contiene il campo "nome" valorizzato con "Cesena"
  - a. **db.classe.update({"cittàVisitate.nome": "Cesena"}, {\$set: {"cittàVisitate.\$.sede": true}}, {multi: true})**
23. Aggiungere il valore "MongoDB" all'array interessi nel proprio documento usando l'opzione upsert
  - a. **db.classe.update({nome: "Enrico"}, {\$push: {interessi: "MongoDB"}}, {upsert: true})**
24. Aggiungere il valore "MongoDB" all'array interessi del documento con nome "Bill" usando l'opzione upsert
  - a. **db.classe.update({nome: "Bill"}, {\$push: {interessi: "MongoDB"}}, {upsert: true})**

## Delete

24. Rimuovere tutti i documenti in cui l'array numeri è vuoto.
  - a. **db.classe.remove({numeri: {\$size: 0}})**