**Créez une base de données quelconque et tester les opérations CRUD Suivants :**

Ouvrir une base de données (existante ou non existante)

use dbsportifs;

créer une collection dans une base de données

db.createCollection("Sportif");

supprimer une base de données

use dbsportifs

db.dropDatabase() ;

supprimer une collection

db.nomCollection.drop()

# insert est obsolète (Depricated) utilisez insertOne insertMany bulkWrite à la place

db.Sportif.insert({

"\_id":"sp1",

"nom":"test",

"prenom":"test"

})

db.Sportif.insertOne({

"\_id":"sp2",

nom:"test",

"prenom":"test",

'age':15,

'actif':true

})

db.Sportif.insertMany([

{"\_id":"sp3",

nom:"test",

"prenom":"test",

'age':15,

'actif':true

},

{"\_id":"sp4",

nom:"test",

"prenom":"test",

'age':15,

'actif':true

},

{"\_id":"sp5",

nom:"test",

"prenom":"test",

'age':15,

'actif':true

},

])

/supprime un ou plusieurs document selon un filtre

db.Sportif.deleteOne({'actif':true});

db.Sportif.deleteMany({'actif':true});

//remplace/insère le champs envoyé pour le premier document du filtre

db.Sportif.updateOne({'actif':true},{$set:{'actif':false}});

db.Sportif.updateOne({'id':'sp1'},{$set:{'actif':false}});

//remplace/insère le champs envoyé pour tous les documents du filtre

db.Sportif.updateMany({'actif':true},{$set:{'actif':false}});

#replace le premier trouvé par le filtre par le document envoyé

db.Sportif.replaceOne({'actif':false},{'\_id':'myId','nom':'testttt','actif':true});

#replace many n'existe pas

db.Sportif.replaceMany({'actif':false},{'actif':true});

##executer plusieurs operation dans un seul script

db.Sportif.bulkWrite([

{

insertOne : {

"\_id":"sp29",

nom:"test",

"prenom":"test",

'age':15,

'actif':true

}

},

{

deleteOne : {"filter":{"\_id":"sp19"}}

}

]);

**Supprimez votre base de données de test et créez la base de données « dbsportifs » et ajoutez les insertions suivantes :**

db.getCollection("Sportif").insertOne(

{\_id:"sp0",

"nom":"Radi0",

"prenom":"Abdessalam0",

"genre":"homme",

"sport":{"description":"marathon"}

})

db.getCollection("Sportif").insertOne(

{\_id:"sp1",

"nom":"Radi",

"prenom":"Abdessalam",

"genre":"homme",

"sport":{"description":"marathon","olympique":"true"}

})

db.Sportif.insertOne(

{\_id:"sp2",

"nom":"Larbi",

"prenom":"Benmbarek",

"genre":"homme",

"sport":{"description":"football","olympique":"true"}

});

db.Sportif.insertMany([

{\_id:"sp3",

"nom":"ELGourch",

"prenom":"Mohamed",

"genre":"homme",

"sport":{"description":"cyclisme","olympique":"true"}

},

{\_id:"sp4",

"nom":"Bidouane",

"prenom":"Nezha",

"genre":"femme",

"sport":{"description":"athletisme","olympique":"true"}

}]);

db.Sportif.bulkWrite(

[{insertOne:{\_id:"sp5",

"nom":"ELGaraa",

"prenom":"Najat",

"genre":"femme",

"sport":{"description":"athletisme","olympique":"true"}

}},

{insertOne:

{\_id:"sp6",

"nom":"Rabii",

"prenom":"Mohamed",

"genre":"homme",

"sport":{"description":"box","olympique":"true"}

}},

{insertOne:

{\_id:"sp7",

"nom":"elguerrouj",

"prenom":"Hicham",

"genre":"homme",

"sport":{"description":"box","olympique":"true"}

}},

{insertOne:

{\_id:"sp8",

"nom":"Abissourour",

"prenom":"Sara",

"genre":"femme",

"sport":{"description":"volley ball","olympique":"true"}

}},

{insertOne:{\_id:"sp9","nom":"Belafrikh",

"prenom":"Amine",

"genre":"homme",

"sport":{"description":"Muay Thai","olympique":"false"}

}},

{insertOne:{\_id:"sp10",

"nom":"Moutawakil",

"prenom":"Naoual",

"genre":"femme",

"nbMedailles":4,

"sport":{"description":"athletisme","olympique":"true"}

}}]);

db.Sportif.insertOne(

{\_id:"spT",

"nom":"test",

"prenom":"test"

});

db.Sportif.updateOne({\_id:"spT"},{$set:{"nom":"testttt"}});

db.Sportif.updateMany({genre:"femme"},{$set:{"genre":"F"}});

db.Sportif.replaceOne({\_id:"spT"},{{nom:"test","genre":"M"}});

db.Sportif.deleteOne({\_id:"spT"});

db.Sportif.deleteMany({\_id:"spT"});

**Find & aggregate**

1. -Quels sont les sportifs de la base de données?

**db.Sportif.find()**

1. Quels sont les sportifs ( nom et prenom) de genre « Homme »?

**db.Sportif.find(**

**{genre:'homme'},**

**{nom:1,prenom:1,\_id:0})**

1. Quel(s) sont les sportifs (identifiant , nom, prenom et genre) qui pratiquent le cyclisme?

**db.Sportif.find(**

**{"sport.description":'cyclisme'},**

**{nom:1,prenom:1,genre:1})**

1. Quels sports ne sont pas olympiques?

db.Sportif.find({$or:[{"sport.olympique":'false'},{"sport.olympique":{$exists:false}}]},{"\_id":0,"sport.description":1 })

sans doublons

db.Sportif.distinct("sport.discription",{$or:[{"sport.olympique":'false'},{"sport.olympique":{$exists:false}}]})

)

1. -Quel est le premier sportif qui jouent au football?

**db.Sportif.findOne({"sport.description" : "football"})**

1. -Afficher les sportifs (nom et prenom) triés par ordre alphabétique des noms,

**db.Sportif.find({}, {nom:1,prenom:1,\_id:0}).sort({nom:1})**

1. Afficher les descriptions des sports triées par ordre alphabétique décroissant,

**db.Sportif.find({},{"sport.description":1}).sort({"sport.description":-1})**

1. -Afficher le nombre de sportifs de la base de données

**db.Sportif.find().count()**

1. -Afficher les descriptions des différents sport sans doublant

**db.Sportif.distinct("sport.description");**

1. -Afficher les deux premières femmes sportives de la base de données

**db.Sportif.find({"genre":"femme"}).limit(2)**

1. -Afficher les sportifs qui ont 3 médailles

**db.Sportif.find({nbMedailles: 3})**

1. -Afficher les noms et prénoms des sportifs qui ont plus que deux médailles

**db.Sportif.find({nbMedailles:{$gt:2}},{\_id:0,nom:1,prenom:1})**

1. -Afficher les noms et prénoms des sportifs qui pratiquent les sports suivants: box, athlétisme et cyclisme

**db.Sportif.find({'sport.description':{$in:['box','athlétisme','cyclisme']}},{nom:1,prenom:1,\_id:0})**

1. -Afficher les noms et prénoms des sportifs qui n’ont pas de médailles

**db.Sportif.find({$or:[{nbMedailles:{$exists:false}},{nbMedailles:0}]},{\_id:0,nom:1,prenom:1})**

1. -Retourner les sportifs qui pratiquent le cyclisme

**db.Sportif.find({"sport.description":"cyclisme"})**

avec aggregate

**db.Sportif.aggregate([{$match:{"sport.description":"cyclisme"}}])**

1. -Calculer la somme des médailles par sport,

**db.Sportif.aggregate([  
{$group :{\_id:'$sport.description' , somme :{$sum : '$nbMedailles' }}}])**

ou bien

**varGroup = {$group :{\_id:'$sport.description' , somme :{$sum : '$nbMedailles' }}}**

**db.Sportif.aggregate([varGroup])**

1. afficher les sport qui ont plus que 3 médailles

**db.Sportif.aggregate([varGroup, {$match: {somme: {$gt: 3}}}])**

1. Afficher les sports (femme) qui ont 2 médailles ou plus

**db.Sportif.aggregate([{$match:{genre:"femme"}},varGroup, {$match: {somme: {$gte: 2}}}])**

1. -Afficher le maximum de médailles par sport

**db.Sportif.aggregate([{$group :{\_id:'$sport.description' , max :{$max : '$nbMedailles' }}}])**

1. -Mettre à jour le champ nbMedailles du sportif « Rabii Mohamed » pour y mettre la valeur 2

**db.Sportif.updateOne({"nom":"Rabii","prenom":"Mohamed"},{$set:{"nbMedailles":2}})**

1. -Pour tous les sportifs, ajouter un champ « nationalité » avec la valeur « marocaine »

**db.Sportif.updateMany({},{$set:{"nationnalité":"Marocaine"}})**

1. -Supprimer les sportifs qui pratiquent le sport « Muay Thai »

**db.Sportif.deleteMany({"sport.description":"Muay Thai"})**

1. -Supprimer les sportifs qui ne pratiquent pas le sport « Muay Thai »

**db.Sportif.deleteMany({"sport.description":{$ne : "Muay Thai"}})**