**UD8. MongoDB Parte 2. Consultas avanzadas**

**Ejercicio 1.1**

use pruebas

db.dropDatabase()

use pruebas

db.createCollection(“películas”)

var pel1 = {\_id: 789, titulo: "Divergente", anyo: 2014, recauda: 288.7, cuesta: 85, saldo: 203.7, rentabilidad: 340, ranking: 0, cod\_saga: "DIVER"}

var pel2 = {\_id: 963, titulo: "Insurgente", anyo: 2015, recauda: 297.2, cuesta: 110, saldo: 187.2, rentabilidad: 270, ranking: 0, cod\_saga: "DIVER"}

var pel3 = {\_id: 874, titulo: "Leal", anyo: 2016, recauda: 179.2, cuesta: 110, saldo: 69.2, rentabilidad: 163, ranking: 0, cod\_saga: "DIVER"}

var pel4 = {\_id: 151, titulo: "El planeta de los simios", anyo: 1968, recauda: 35.2, cuesta: 5.4, saldo: 27.1, rentabilidad: 602, ranking: 0, cod\_saga: "SIMIOS1"}

var pel5 = {\_id: 666, titulo: "El Exorcista", anyo: 1973, recauda: 441.3, cuesta: 12}

var pel6 = {\_id: 215, titulo: "Regreso al planeta de los simios", anyo: 1970, recauda: 18.9, cuesta: 3, saldo: 15.9, rentabilidad: 630, ranking: 0, cod\_saga: "SIMIOS1"}

var pel7 = {\_id: 687, titulo: "Huida del planeta de los simios", anyo: 1971, recauda: 12.3, cuesta: 2.5, saldo: 9.8, rentabilidad: 492, ranking: 0, cod\_saga: "SIMIOS1"}

var pel8 = {\_id: 278, titulo: "La rebelion de los simios", anyo: 1972, recauda: 9.7, cuesta: 1.7, saldo: 8, rentabilidad: 571, ranking: 0, cod\_saga: "SIMIOS1"}

var pel9 = {\_id: 987, titulo: "La batalla por el planeta de los simios", anyo: 1973, recauda: 8.8, cuesta: 1.8, saldo: 7, rentabilidad: 489, ranking: 0, cod\_saga: "SIMIOS1"}

var pel10 = {\_id: 345, titulo: "El planeta de los simios", anyo: 2001, recauda: 362.2, cuesta: 100, saldo: 262.2, rentabilidad: 362, ranking: 0}

var pel11 = {\_id: 428, titulo: "El origen del planeta de los simios", anyo: 2011, recauda: 481.8, cuesta: 93, saldo: 388.8, rentabilidad: 518, ranking: 0, cod\_saga: "SIMIOS2"}

var pel12 = {\_id: 887, titulo: "El amanecer del planeta de los simios", anyo: 2014, recauda: 707.4, cuesta: 170, saldo: 537.4, rentabilidad: 416, ranking: 0, cod\_saga: "SIMIOS2"}

var pel13 = {\_id: 587, titulo: "La guerra del planeta de los simios", anyo: 2017, recauda: 490.7, cuesta: 150, saldo: 340.7, rentabilidad: 327, ranking: 0, cod\_saga: "SIMIOS2"}

db.peliculas.insertMany([pel1, pel2, pel3, pel4, pel5, pel6, pel7, pel8, pel9, pel10, pel11, pel12, pel13])

**Ejercicio 1.2**

var filtro = {\_id: 666}

var j\_set = {saldo: 429.3, rentabilidad: 3678}

var accion = {$set: j\_set}

db.peliculas.updateOne(filtro, accion)

*Para comprobar el valor cambiado:*

var proyec = {titulo: 1, anyo:1 , saldo: 1, rentabilidad: 1}

db.peliculas.find(filtro, proyec)

A computer screen shot of a black background

Description automatically generated

**Ejercicio 1.3**

Ranking 5:

var cond = {$exists: true}

var filtro1 = {rentabilidad: cond}

var cond2 = {$gte: 500}

var filtro2 = {rentabilidad: cond2}

var filtrocomb = {$and:[filtro1, filtro2]}

var valor = {ranking: 5}

var accion = {$set: valor}

db.peliculas.updateMany(filtrocomb, accion)

Ranking 4:

var cond = {$exists: true}

var filtro1 = {rentabilidad: cond}

var cond2 = {$gte: 400, $le: 500}

var filtro2 = {rentabilidad: cond2}

var filtrocomb = {$and:[filtro1, filtro2]}

var valor = {ranking: 4}

var accion = {$set: valor}

db.peliculas.updateMany(filtrocomb, accion)

Ranking 3:

var cond = {$exists: true}

var filtro1 = {rentabilidad: cond}

var cond2 = {$gte: 300, $lt: 400}

var filtro2 = {rentabilidad: cond2}

var filtrocomb = {$and:[filtro1, filtro2]}

var valor = {ranking: 3}

var accion = {$set: valor}

db.peliculas.updateMany(filtrocomb, accion)

Ranking 2:

var cond = {$exists: true}

var filtro1 = {rentabilidad: cond}

var cond2 = {$gte: 200, $lt: 300}

var filtro2 = {rentabilidad: cond2}

var filtrocomb = {$and:[filtro1, filtro2]}

var valor = {ranking: 2}

var accion = {$set: valor}

db.peliculas.updateMany(filtrocomb, accion)

Ranking 1:

var cond = {$exists: true}

var filtro1 = {rentabilidad: cond}

var cond2 = {$gte: 100, $lt: 200}

var filtro2 = {rentabilidad: cond2}

var filtrocomb = {$and:[filtro1, filtro2]}

var valor = {ranking: 1}

var accion = {$set: valor}

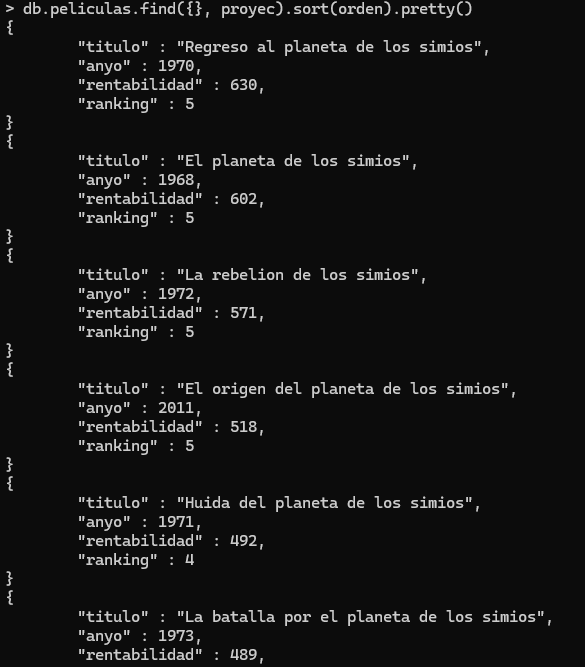
db.peliculas.updateMany(filtrocomb, accion)

Mostrar las películas:

var proyec = {\_id: 0, titulo: 1, anyo: 1, rentabilidad: 1, ranking: 1}

var orden = {ranking: -1, rentabilidad: -1}

db.peliculas.find({}, proyec).sort(orden)



… La última es Insurgente

**Ejercicio 1.4**

Hacer copias de la colección:

db.peliculas.aggregate([{$out: “SAGA\_DIVER”}])

db.peliculas.aggregate([{$out: “SAGA\_SIMIOS1”}])

db.peliculas.aggregate([{$out: “SAGA\_SIMIOS2”}])

show collections 🡪 para ver si se han creado las colecciones

Borrar las películas que no correspondan a cada saga:

var cond = {$exists: false}

var filtro1 = {cod\_saga: cond}

var cond2 = {$ne: "DIVER"}

var filtro2 = {cod\_saga: cond2}

var filtrocomb = {$or:[filtro1, filtro2]}

db.SAGA\_DIVER.deleteMany(filtrocomb)

var cond = {$exists: false}

var filtro1 = {cod\_saga: cond}

var cond2 = {$ne: "SIMIOS1"}

var filtro2 = {cod\_saga: cond2}

var filtrocomb = {$or:[filtro1, filtro2]}

db.SAGA\_SIMIOS1.deleteMany(filtrocomb)

var cond = {$exists: false}

var filtro1 = {cod\_saga: cond}

var cond2 = {$ne: "SIMIOS2"}

var filtro2 = {cod\_saga: cond2}

var filtrocomb = {$or:[filtro1, filtro2]}

db.SAGA\_SIMIOS2.deleteMany(filtrocomb)

Eliminar los campos especificados:

var campos = {recauda: "", cuesta: "", saldo: "", rentabilidad: "", ranking: "", cod\_saga: ""}

var accion = {$unset: campos}

db.SAGA\_DIVER.updateMany({}, accion)

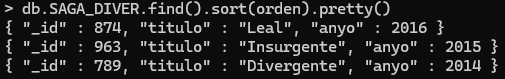
db.SAGA\_SIMIOS1.updateMany({}, accion)

db.SAGA\_SIMIOS2.updateMany({}, accion)

Mostrar todas las películas de las 3 colecciones:

var orden = {anyo: -1}

db.SAGA\_DIVER.find().sort(orden)



db.SAGA\_SIMIOS1.find().sort(orden)

A screen shot of a computer

Description automatically generated

db.SAGA\_SIMIOS2.find().sort(orden)

A computer screen shot of white text

Description automatically generated

**Ejercicio 1.5**

Crear nuevas colecciones:

db.peliculas.aggregate([{$out: “mas\_rentables”}])

db.peliculas.aggregate([{$out: “menos\_rentables”}])

Eliminar las películas cuyo ránking no correspondan con las colecciones anteriores:

var cond = {$ne: 5}

var filtro = {ranking: cond}

db.mas\_rentables.deleteMany(filtro)

var cond = {$ne: 1}

var filtro = {ranking: cond}

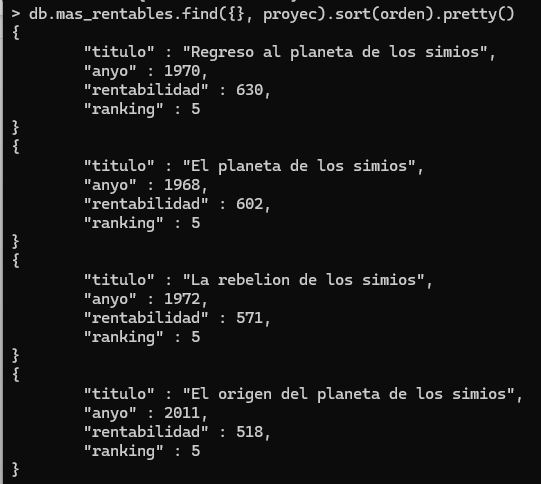
db.menos\_rentables.deleteMany(filtro)

Mostrar las películas de las dos colecciones ordenadas por ranking descendiente:

var orden = {rentabilidad: -1}

var proyec = {\_id: 0, titulo: 1, anyo: 1, rentabilidad: 1, ranking: 1}

db.mas\_rentables.find({}, proyec).sort(orden)



db.menos\_rentables.find({}, proyec).sort(orden)



**Ejercicio 1.6**

Introducir las películas en la colección “películas”:

var anillo1 = {\_id: 05, titulo: "El señor de los anillos. La comunidad del anillo", anyo: 2001, recauda: 898, cuesta: 93, saldo: 805, rentabilidad: 965, ranking: 0, cod\_saga: "ANILLOS"}

var anillo2 = {\_id: 24, titulo: "El señor de los anillos. Las dos torres", anyo: 2002, recauda: 947, cuesta: 94, saldo: 805, rentabilidad: 1007, ranking: 0, cod\_saga: "ANILLOS"}

var anillo3 = {\_id: 32, titulo: "El señor de los anillos. El retorno del rey", anyo: 2003, recauda: 1147, cuesta: 94, saldo: 805, rentabilidad: 1220, ranking: 0, cod\_saga: "ANILLOS"}

db.peliculas.insertMany([anillo1, anillo2, anillo3])

Asignarles ranking:

var cond = {$exists: true}

var filtro1 = {rentabilidad: cond}

var cond2 = {$gte: 500}

var filtro2 = {rentabilidad: cond2}

var filtrocomb = {$and:[filtro1, filtro2]}

var valor = {ranking: 5}

var accion = {$set: valor}

db.peliculas.updateMany(filtrocomb, accion)

Crear nueva colección y dejar sólo esas películas:

db.peliculas.aggregate([{$out: “SAGA\_ANILLOS”}])

var cond = {$ne: "ANILLOS"}

var filtro = {cod\_saga: cond}

var cond2 = {$exists: false}

var filtro2 = {cod\_saga: cond2}

var combi = {$or:[filtro, filtro2]}

db.SAGA\_ANILLOS.deleteMany(combi)

Mostrar películas:

var proyec = {\_id: 0, titulo: 1, anyo: 1, cod\_saga: 1}

var orden = {anyo: 1}

db.SAGA\_ANILLOS.find({}, proyec).sort(orden)

