Importando Banco CSV

mongoimport -d pokemon -c pokemons --type csv --file D:\veloxSZ(Gian)\pokemon-challenge\pokemon.csv –headerline

mongoimport -d pokemon -c combats --type csv --file D:\veloxSZ(Gian)\pokemon-challenge\combats.csv --headerline

mongoimport -d pokemon -c tests --type csv --file D:\veloxSZ(Gian)\pokemon-challenge\tests.csv --headerline

-d - database

-c - collection

--type - Tipo do arquivo que vai ser importado

--file – Caminho do Arquivo

--headerline – ignorar a linha do cabeçalho

&& - importa vários bancos de uma vez

db.pokemons.find({HP: {$gte:40}},

{Name: 1,

HP: 1,

\_id:0}).sort({Name:1}).skip(100).limit(100)

Busca todos os pokemons onde o HP é maior ou igual a 40, pula 100 registros e mostra os próximos 100, mostrando as colunas nome e HP.

Inserindo Consulta Variável

var items = db.pokemons.find({HP: {$gte:40}},

{Name: 1,

HP: 1,

\_id:0}).sort({Name:1}).skip(100).limit(100)

Para executar:

Items

Aggreate

Match – encontra tudo que é igual

Group – Agrupa

db.funcionarios.aggregate([{"$match":{"Empresa":"Google"}},

{"$group":{"\_id":"$Nome"}}])

Project- apelido

db.funcionarios.aggregate([{"$match":{"Empresa":"Google"}},

{"$group":{"\_id":"$Nome"}},

{"$project":{

nome:"$\_id",

\_id:0

}

}])

Sum – soma - count

db.funcionarios.aggregate([{"$match":{"Empresa":"Google"}},

{"$group":{"\_id":"$Nome"}},

{"$project":{

nome:"$\_id",

\_id:0

}

}])

Count

db.funcionarios.aggregate([{"$match":{"Empresa":"Google"}},

{"$count":"total"}

])

Apresentar campos específicos

db.funcionarios.aggregate([{"$match":{"Empresa":"Google"}},

{"$project":{

"Nome":1,

"Cargo":1,

\_id:0

}

}])

Definir nome personalizado

db.funcionarios.aggregate([{"$match":{"Empresa":"Google"}},

{"$project":{

"Nome":1,

"Ocupação":"$Cargo",

\_id:0

}

}])