**Gestion Des données - MongoDB**

# Exercice 3

Créez une base de données qui porte le nom de votre groupe « exemple (DEV203)

Importez le fichier movies.json dans une collection qui porte le même nom.

# Les Requêtes

Réalisez les requêtes suivantes et coller le code correspondant en dessous de la question

// 1.   les films dont le nom contient le mot

// «forest »

db.movies.find({title : /forest/i})

// 2.   les films dans lesquelles participe

// l’acteur « tom hanks »

db.movies.find({cast:"Tom Hanks"})

// 3.   les films du pays France

db.movies.find({countries:"France"})

// 4.   les films de France avant 1990

db.movies.find({countries:"France",year:{$lte:1990}})

// 5.   chercher les films dont la description

// contient le mot « mystery »

 db.movies.find({fullplot:/mystery/i})

// 6.   le nombre des films qui ont un rating

// imdb supérieur ou égale à  7

db.movies.find({"imdb.rating":{$gte:7}}).count()

// 7.   le nombre des films par acteur dans

// l’ordre croissant sur le nom d’acteur

 db.movies.aggregate([

    {$unwind : {path : "$cast"}},

    {$group : {"\_id":'$cast' , 'total de films':{$sum : 1}}},

    {$sort : {\_id : 1}}

])

// 8.   les films dont le titre commence par

// « the » « utiliser « $regex »

db.movies.find({"title":{$regex:/^the/i}})

// 9.   les différents type de films

db.movies.distinct("type")

// 10.  la liste des acteur qui ont participé

// dans plus que 10 films

 db.movies.aggregate([

    {$unwind : {path : "$cast"}},

    {$group : {\_id:'$cast' , total:{$sum : 1}}},

    {$match : {total : {$gt:10}}}

])

// 11.  le nombre de films produit par pays

db.movies.aggregate([{$unwind: {path:'$countries'}},

    {$group: {'\_id':'$countries', count :{$sum:1}}},

    {$sort: {\_id:-1}

}])

// 12.  le rating imdb moyenne des films de

// « Steven Spielberg»

 db.movies.aggregate([

   {$match:{cast:"Steven Spielberg"}},

    {$group: {'\_id':'average', moyenne :{$avg:'$imdb.rating'}}}

])

// 13.  les différents genres de films de

// « Steven Spielberg»

db.movies.aggregate([

    {$match:{directors:"Steven Spielberg"}},

    {$unwind:"$genres"},

    {$group:{\_id:"$genres"}}

 ])

// 14.  qle nombre des films par année du genre

//  « sci-fi »

db.movies.aggregate([

    {$match:{'genres':'Sci-Fi'}},

    {$group:{\_id:'$year',count:{$sum:1}}}

])

// 15.  Les films d’ « angelina jolie » qui

// ont rated « pg »

db.movies.find({cast:"Angelina Jolie",rated:"PG"});

// 16.  Le nombre de films par genre  -

// n’afficher que les 5 plus grands

 db.movies.aggregate([

    {$unwind:'$genres'},

    {$group:{\_id:"$genres",count:{$sum:1}}},

    {$sort:{count:-1}},

    {$limit:5}

 ])

// 17.  trier les films de « tom hanks » par

// imdb score dans l’ordre décroissant

  db.movies.find({cast:"Tom Hanks"}).sort({"imdb.rating":-1})

// 18.  le pays qui a le film qui a réalisé le

// plus de reviews

 db.movies.find(

    {},{"countries" :1, \_id:0}

 ).sort({"tomatoes.viewer.numReviews" :-1}).limit(1)

// 19.  donnez les cinq meilleurs directeurs de

// l’année 2000

db.movies.aggregate([

    {$match:{year:2000}},

    {$unwind:{path:"$directors"}},

    {$group:{\_id:"$directors",

    total:{$sum:"$imdb.rating"}}},

{$sort:{total:-1}},

{$limit:5}

])

// 20.  l’année dans laquelle on a produit le

// plus grand nombre de films

db.movies.aggregate([

    {$group:{\_id:"$year",total:{$sum:1}}},

    {$sort:{total:-1}},

{$limit:1}

])