HW6 - 1.js: ----------------------------------------------------------------------------------------------------------------------

students =

[

{

lastname: "duck",

firstname: "donald",

dateOfBirth: new Date("1993-03-17"),

email: "dduck@disney.com",

grade: "C",

major: "MATH",

quizzes : [43, 71, 53, 100, 0, 32, 31, 41],

report : { score: 71, ta: "Clark Richards" },

exams :

[

{ score: 58, ta: "Nancy Hew", comments: "Need to step it up" },

{ score: 69, ta: "Clark Richards", comments: "Still not passing" },

{ score: 45, ta: "Nancy Hew" },

{ score: 33, ta: "Nancy Hew" },

{ score: 61, ta: "Clark Richards", comments: "Stop laying eggs on these exams" }

]

}

]

db.students.insertMany(students);

db.students.find({ email : "dduck@disney.com" }, { \_id : 0 });

HW6 - 10.js: ----------------------------------------------------------------------------------------------------------------------

db.students.aggregate(

[

{

$project: {

\_id: 0, firstname: 1, lastname: 1,

avgExam : {$avg : "$exams.score"}

}

},

// { $unwind: "$exams" },

// {

// $group: {

// \_id: "$email",

// lastname: {$first: "$lastname"},

// firstname: {$first: "$firstname"},

// avgExam: {$avg : "$exams.score"}

// }

// },

{ $sort: {avgExam: -1} }

]

);

// print("\n\n");

// db.students.find(

// {},

// {\_id: 0, lastname: 1, firstname: 1, "exams.score" : 1}

// );

HW6 - 11.js: ----------------------------------------------------------------------------------------------------------------------

db.students.aggregate(

[

{

$match: {

$or:

[

//Probably should have written this in one large array...

//...instead of OR'ing grade multiple times. But this looks better :)

{grade: { $in: [ "C", "C+", "C-" ] }},

{grade: { $in: [ "D", "D+", "D-" ] }},

{grade: { $in: [ "F", "F+", "F-" ] }}

]

}

},

//The "count: {$sum: 1}" line is how you actually count items in each group!

{

$group: {

\_id : null,

count : {$sum : 1}

}

},

{

$project: {\_id: 0, count: 1}

}

]

);

// print("\n\n");

// db.students.find(

// { grade: { $in: [ "C", "C+", "C-", "D", "D+", "D-", "F", "F+", "F-" ] } },

// {\_id: 0, lastname: 1, firstname: 1, "grade" : 1}

// );

// //NOTE: This works good too. However, DON'T USE $group AND $count IN THE SAME PIPELINE.

// // It can lead to trouble.

// db.students.aggregate(

// [

// {$match: {grade: {$in: ["C", "D", "F"]}}},

// {$count: "count"}

// ]

// );

HW6 - 12.js: ----------------------------------------------------------------------------------------------------------------------

// print("BEFORE\n");

// db.students.find(

// {email: "mickey@disney.com"},

// // {lastname: "mouse", firstname: "mickey"},

// {\_id: 0, lastname: 1, firstname: 1, "grade" : 1, "report.ta": 1}

// );

// print("\n\n");

db.students.updateMany(

{

email: "mickey@disney.com"

//lastname: "mouse", firstname: "mickey"

},

{

$set: {grade: "A", "report.ta": "Clark Richards"}

},

);

// print("\nAFTER\n");

// db.students.find(

// {email: "mickey@disney.com"},

// // {lastname: "mouse", firstname: "mickey"},

// {\_id: 0, lastname: 1, firstname: 1, "grade" : 1, "report.ta": 1}

// );

db.students.find(

{email: "mickey@disney.com"},

{\_id: 0 }

);

HW6 - 13.js: ----------------------------------------------------------------------------------------------------------------------

// print("BEFORE\n");

// db.students.find(

// {lastname: "vander zanden", firstname: "smiley"},

// {\_id: 0, lastname: 1, firstname: 1, exams : 1}

// ).pretty();

// print("\n\n");

db.students.updateMany(

{

email: "smiley@utk.edu"

//lastname: "vander zanden", firstname: "smiley"

},

{

$push: {exams: {score: 95, ta: "Jeff Hardy"} }

},

);

// print("\nAFTER\n");

// db.students.find(

// {lastname: "vander zanden", firstname: "smiley"},

// {\_id: 0, lastname: 1, firstname: 1, exams : 1}

// ).pretty();

db.students.find(

{email: "smiley@utk.edu"},

//{lastname: "vander zanden", firstname: "smiley"},

{\_id: 0 }

);

HW6 - 14.js: ----------------------------------------------------------------------------------------------------------------------

// print("BEFORE\n");

// db.students.find(

// {lastname: "mouse", firstname: "minnie"},

// {\_id: 0, lastname: 1, firstname: 1, exams : 1}

// ).pretty();

// print("\n\n");

db.students.updateMany(

{

email: "mmouse@disney.com"

//lastname: "mouse", firstname: "minnie"

},

{

$set:

{

"exams.0.score": 86,

"exams.2.score": 100,

"exams.2.ta": "Nancy Hew"

// //This actually replaced the entire object instead of the specific object members. Ooops

// "exams.0": {score: 86},

// "exams.2": {score: 100, ta: "Nancy Hew"}

}

},

);

// print("\nAFTER\n");

// db.students.find(

// {lastname: "mouse", firstname: "minnie"},

// {\_id: 0, lastname: 1, firstname: 1, exams : 1}

// ).pretty();

db.students.find(

{email: "mmouse@disney.com"},

//{lastname: "mouse", firstname: "minnie"},

{\_id: 0 }

);

HW6 - 15.js: ----------------------------------------------------------------------------------------------------------------------

// print("BEFORE\n");

// db.students.find(

// {lastname: "ape", firstname: "great"},

// {\_id: 0, lastname: 1, firstname: 1}

// ).pretty();

// print("\n\n");

db.students.deleteMany(

{

email: "gape@warner.com"

//lastname: "ape", firstname: "great"

}

);

// print("\nAFTER\n");

// db.students.find(

// {lastname: "ape", firstname: "great"},

// {\_id: 0, lastname: 1, firstname: 1}

// ).pretty();

db.students.find(

{email: "gape@warner.com"},

//{lastname: "ape", firstname: "great"},

{\_id: 0 }

);

HW6 - 2.js: ----------------------------------------------------------------------------------------------------------------------

db.students.find(

//{grade: "A"},

{grade: { $in: [ "A", "A+", "A-" ] } },

{\_id: 0, firstname: 1, lastname: 1}

);

// print("\n\n");

// db.students.find(

// {},

// {\_id: 0, lastname: 1, firstname: 1, grade: 1}

// );

HW6 - 3.js: ----------------------------------------------------------------------------------------------------------------------

db.students.find(

{ dateOfBirth: {$gte: new Date("2000-01-01"), $lt: new Date("2001-01-01")} },

{ \_id: 0, firstname: 1, lastname: 1, dateOfBirth: 1, grade: 1 }

);

HW6 - 4.js: ----------------------------------------------------------------------------------------------------------------------

db.students.find(

{ grade: { $in: [ "A", "A+", "A-" ] }, quizzes: { $elemMatch: {$lt: 50} } },

{ \_id: 0, firstname: 1, lastname: 1, quizzes: 1 }

);

HW6 - 5.js: ----------------------------------------------------------------------------------------------------------------------

db.students.find(

{ exams: { $elemMatch: {ta: "Jeff Hardy"} } },

{ \_id: 0, firstname: 1, lastname: 1, exams: 1 }

).sort(

{ lastname: 1, firstname: -1 }

);

HW6 - 6.js: ----------------------------------------------------------------------------------------------------------------------

db.students.find(

{$or:

[

{ "report.score" : {$gte: 60, $lte: 80}, "report.ta" : "Nancy Hew" },

{ "report.score" : {$gte: 80, $lte: 100}, "report.ta" : "Nancy Hew" }

]

},

{ \_id: 0, firstname: 1, lastname: 1, "report.score": 1 }

//{ \_id: 0, lastname: 1, firstname: 1, "report.score": 1 , "report.ta": 1}

);

HW6 - 7.js: ----------------------------------------------------------------------------------------------------------------------

db.students.find(

{ "report.comments" : { $exists : false } },

{ \_id: 0, firstname: 1, lastname: 1, report: 1 }

);

// print("\n\n");

// db.students.find(

// {},

// {\_id: 0, lastname: 1, firstname: 1, report : 1}

// );

HW6 - 8.js: ----------------------------------------------------------------------------------------------------------------------

db.students.aggregate(

[

{

$group: {

\_id : "$major",

count : {$sum : 1}

}

}

]

);

// print("\n\n");

// db.students.find(

// {},

// {lastname: 1, firstname: 1, major : 1}

// ).sort({major : 1});

HW6 - 9.js: ----------------------------------------------------------------------------------------------------------------------

db.students.aggregate(

[

{

$project: {

\_id: 0, firstname: 1, lastname: 1,

min : {$min : "$quizzes"},

max : {$max : "$quizzes"},

avgQuiz : {$avg : "$quizzes"}

}

}

// { $unwind: "$quizzes"},

// {

// $group: {

// \_id: "$email",

// lastname: {$first: "$lastname"},

// firstname: {$first: "$firstname"},

// min : {$min : "$quizzes"},

// max : {$max : "$quizzes"},

// avgQuiz : {$avg : "$quizzes"}

// }

// },

]

);

// print("\n\n");

// db.students.find(

// {},

// {\_id: 0, lastname: 1, firstname: 1, quizzes : 1}

// );