Nota: para cada pregunta deberás resolverla utilizando

1. “aggregate framework”
2. “map-reduce”

Utiliza la base de datos GRADES.JSON para contestar las siguientes preguntas:

1. ¿Cuál es el total de alumnos inscritos?

var mapFunction1 = function() {

emit(this.student\_id, 1);

};

var reduceFunction1 = function(student\_id, values) {

return Array.sum(values);

};

db.partial.mapReduce(

mapFunction1,

reduceFunction1,

{

out:"TotalDeAlumnos"

}

)

db.TotalDeAlumnos.count()

1. ¿Cuántos cursos se han impartido?

var mapFunction1 = function() {

emit(this.class\_id, 1);

};

var reduceFunction1 = function(class\_id, values) {

return Array.sum(values);

};

db.partial.mapReduce(

mapFunction1,

reduceFunction1,

{

out:"TotalDeClases"

}

)

db.TotalDeClases.count()

1. Encuentra, para cada alumno, su promedio obtenido en cada una de las clases en las que fue inscrito (promedia exámenes, quizes, tareas y todas las actividades realizadas que contenga un grade)

var mapFunction1 = function() {

this.scores.forEach((value)=>{

emit({student\_id:this.student\_id,class\_id: this.class\_id,entregas:this.scores.length},value.score);

});

};

var reduceFunction1 = function(name, score) {

return (Array.sum(score)/(name.entregas)) ;

};

db.partial.mapReduce(

mapFunction1,

reduceFunction1,

{

out:"PromedioDeAlumno"

}

)

db.PromedioDeAlumno.find().pretty()

1. ¿Cuál fue la materia que tiene la calificación más baja (muestra el id de la materia, el id del alumno y la calificación)?

var mapFunction1 = function() {

this.scores.forEach((value)=>{

emit({student\_id:this.student\_id,class\_id: this.class\_id,entregas:this.scores.length},value.score);

});

};

var reduceFunction1 = function(name, score) {

return (Array.sum(score)/(name.entregas)) ;

};

db.partial.mapReduce(

mapFunction1,

reduceFunction1,

{

out:"PromedioDeAlumno"

}

)

db.PromedioDeAlumno.find().sort({value:1}).limit(1).pretty()

1. ¿Cuál es la materia en la que se han dejado más tareas?

var mapFunction1 = function() {  
 emit(this.class\_id,this.scores.length);  
};  
var reduceFunction1 = function(name,score) {  
 return (score.reverse()[0]);  
};  
db.partial.mapReduce(  
 mapFunction1,  
 reduceFunction1,  
 {  
 out:"TotaldeTareas"  
 }  
)

db.TotaldeTareas.find().sort({value:-1}).limit(1).pretty()

1. ¿Qué alumno se inscribió en más cursos?

var mapFunction1 = function() {  
 emit(this.student\_id,1);  
};  
var reduceFunction1 = function(name,values) {  
 return (Array.sum(values));  
};  
db.partial.mapReduce(  
 mapFunction1,  
 reduceFunction1,  
 {  
 out:"ClasesTomadas"  
 }  
)

db.ClasesTomadas.find().sort({value:-1}).limit(1).pretty()

1. ¿Cuál fue el curso que tuvo más reprobados?

var mapFunction1 = function() {

this.scores.forEach((value)=>{

emit({student\_id:this.student\_id,class\_id: this.class\_id,entregas:this.scores.length},value.score);

});

};

var reduceFunction1 = function(name, score) {

return (Array.sum(score)/(name.entregas)) ;

};

db.partial.mapReduce(

mapFunction1,

reduceFunction1,

{

out:"PromedioDeAlumno"

}

)

var mapFunction1 = function() {

if(this.value<70){emit(this.\_id.student\_id,1);}};

var reduceFunction1 = function(name,values) {  
 return (Array.sum(values));  
};

db.PromedioDeAlumno.mapReduce(

mapFunction1,

reduceFunction1,

{

out:"ReprobadasDeAlumno"

}

)

db.ReprobadasDeAlumno.find().sort({value:-1}).limit(1).pretty()

Notas:

* códigos similares equivalen a 0
* el profesor no resuelve dudas (todo está en google)