

## getEventsBySport

```
db.sports.aggregate([{$match : {"sportsType" : "Swimming"}},
  {$unwind : "$events"},
  {$group : {_id : "$events.eventType"}}]).toArray()
```

The result is:

```
olympic_games> db.sports.aggregate([{$match : {"sportsType" : "Swimming"}},
...   {$unwind : "$events"}, {$group : {_id : "$events.eventType"}}])
[
  { _id: "Swimming Men's 100 metres Butterfly" },
  { _id: "Swimming Women's 200 metres Individual Medley" },
  { _id: "Swimming Women's 100 metres Butterfly" },
  { _id: "Swimming Men's 10 kilometres Open Water" },
  { _id: "Swimming Men's 880 yard Freestyle" },
  { _id: "Swimming Men's 4 x 250 metres Freestyle Relay" },
  { _id: "Swimming Women's 400 metres Freestyle" },
  { _id: "Swimming Men's 200 metres Obstacle Course" },
  { _id: "Swimming Men's 100 Yard Backstroke" },
  { _id: "Swimming Men's 100 yard Freestyle" },
  { _id: "Swimming Men's 4 x 200 metres Freestyle Relay" },
  { _id: "Swimming Men's 100 metres Freestyle For Sailors" },
  { _id: "Swimming Women's 200 metres Butterfly" },
  { _id: "Swimming Men's 440 yard Freestyle" },
  { _id: "Swimming Men's 400 metres Freestyle" },
  { _id: "Swimming Women's 4 x 100 metres Medley Relay" },
  { _id: "Swimming Men's 400 metres Individual Medley" },
  { _id: "Swimming Men's 4 x 50 Yard Freestyle Relay" },
  { _id: "Swimming Women's 200 metres Backstroke" },
  { _id: "Swimming Men's 100 metres Breaststroke" }
]
```

## getGenderStatisticsBySportType

(in index .js -> i am using both sportsType and sportId this function accepts sportId)

```
db.athletes.aggregate([{$match : {"participations.sportId": 9}},
  {$group : {_id : "$sex", count : {$sum : 1}}},
  {$project : {_id : 0, sex : "$_id", count : 1}}])
```

The result is:

```
olympic_games> db.athletes.aggregate([{$match : {"participations.sportId": 9}},
...   {$group : {_id : "$sex", count : {$sum : 1}}},
...   {$project : {_id : 0, sex : "$_id", count : 1}}])
[ { count: 2905, sex: 'F' }, { count: 3560, sex: 'M' } ]
```

## getEvents

```
db.sports.aggregate([{$match : {"events.eventType": new RegExp("^" + "Swimming",
"i")}},
    {$unwind : "$events"},
    {$match : {"events.eventType": new RegExp("^" + "Swimming")}},
    {$group : { _id: "$events.eventType"}},
    {$project : { _id :0, eventType : "$_id"}}]).toArray()
```

The result is:

```
olympic_games> db.sports.aggregate([{$match : {"events.eventType": new RegExp("^" + "Swimming", "i")}},
...                               {$unwind : "$events"},
...                               {$match : {"events.eventType": new RegExp("^" + "Swimming")}},
...                               {$group : { _id: "$events.eventType"}},
...                               {$project : { _id :0, eventType : "$_id"}}]).toArray()
[
  { eventType: "Swimming Men's 880 yard Freestyle" },
  { eventType: "Swimming Men's 10 kilometres Open Water" },
  { eventType: "Swimming Women's 400 metres Freestyle" },
  { eventType: "Swimming Men's 4 x 250 metres Freestyle Relay" },
  { eventType: "Swimming Men's 200 metres Team Swimming" },
  { eventType: "Swimming Men's 100 Yard Backstroke" },
  { eventType: "Swimming Men's 100 yard Freestyle" },
  { eventType: "Swimming Men's 200 metres Obstacle Course" },
  { eventType: "Swimming Women's 200 metres Butterfly" },
  { eventType: "Swimming Men's 440 yard Freestyle" },
  { eventType: "Swimming Men's 4 x 200 metres Freestyle Relay" },
  { eventType: "Swimming Women's 200 metres Backstroke" },
  { eventType: "Swimming Men's 100 metres Breaststroke" },
  { eventType: "Swimming Women's 200 metres Freestyle" },
  { eventType: "Swimming Women's 50 metres Freestyle" },
  { eventType: "Swimming Men's 400 metres Freestyle" },
  { eventType: "Swimming Women's 4 x 100 metres Medley Relay" },
  { eventType: "Swimming Men's 400 metres Individual Medley" },
  { eventType: "Swimming Men's 4 x 50 Yard Freestyle Relay" },
  { eventType: "Swimming Men's 4,000 metres Freestyle" },
  { eventType: "Swimming Men's One Mile Freestyle" },
  { eventType: "Swimming Women's 10 kilometres Open Water" },
  { eventType: "Swimming Men's 200 metres Freestyle" },
  { eventType: "Swimming Men's Underwater Swimming" },
```

## getFemaleAthletesTallerThan180

```
olympic_games> db.athletes.find({"participations.height" : {"$gte": 180},
"sex":"F"}).count()
```

The result is:

```
olympic_games> db.athletes.find({"participations.height" : {"$gte": 180}, "sex":"F").count()  
3356
```

## UpdateAthleteAgeByIncreasingOne

```
olympic_games> db.athletes.update({"athleteId": 71, "participations.sportId":  
31}, {"$inc":{"participations.$age":1}})
```

The result is:

```
olympic_games> db.athletes.update({"athleteId": 71, "participations.sportId": 31}, {"$inc":{"participations.$age":1}})  
{  
  acknowledged: true,  
  insertedId: null,  
  matchedCount: 1,  
  modifiedCount: 1,  
  upsertedCount: 0  
}  
_id: ObjectId("619c64f79798404570a6f2f6")  
athleteId: 71  
name: 'Juan Antonio Aball Delgado',  
sex: 'M',  
participations: [  
  {  
    sportId: 31,  
    gameId: 1,  
    teamId: 23,  
    age: 28,  
    height: 172,  
    weight: 82,  
    medalType: 'Null'  
  }  
]  
}
```