

MongoDB Queries

Pablo Benayas Penas

Mayo 2020

Introducción

El objetivo de este trabajo es elegir una colección de datos de internet, importar dicho archivo a mongo, realizar consultas y acabar con una conclusión de resultados.

Para ello, he elegido el listado de jugadores de FIFA 19, el cual ha sido extraído en kaggle¹.

Las campos que componen cada jugador de fútbol son:

- ID unique: código identificador del jugador
- Name: nombre
- Age: edad
- Nationality: nacionalidad
- Overall: puntuación general
- Potential: puntuación potencial en las próximas temporadas
- Club: club actual
- Value (M€): valor actual en el mercado (millones de euros)
- Wage (K€): salario actual (miles de euros)
- Preferred Foot: derecha/izquierda
- International Reputation: puntuación en escala de 5
- Body Type: complexión física (algunos tipos de cuerpo han sido definidos con nombres propios como 'Cristiano Ronaldo' o 'Akinfenwa').
- Position: posición en la pista. Posibles posiciones:

¹ <https://www.kaggle.com/karangadiya/fifa19/data>

	POSITION
GK	Goalkeeper
RB	Right Back
CB	Center Back
LB	Left Back
RWB	Right Wing Back
LWB	Left Wing Back
CDM	Center Defensive Midfielder
CM	Center Midfielder
CAM	Center Attacking Midfielder
RM	Right Midfielder
LM	Left Midfielder
RW	Right Wing
LW	Left Wing
CF	Center Forward
ST	Striker

- Release Clause (M€): cláusula para liberar jugador cuando el contrato sigue vigente (millones de euros).

Importar colección en MongoDB

Pasos:

1). Cargo mongod

```
C:\> Command Prompt - mongod
Microsoft Windows [Version 10.0.18362.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\pablo>cd C:\Program Files\MongoDB\Server\4.2\bin

C:\Program Files\MongoDB\Server\4.2\bin>mongod
2020-05-10T07:03:43.402+0200 I CONTROL [main] Automatically disabling TLS 1.0, to
2020-05-10T07:03:43.911+0200 W ASIO [main] No TransportLayer configured during
2020-05-10T07:03:43.912+0200 I CONTROL [initandlisten] MongoDB starting : pid=786
2020-05-10T07:03:43.912+0200 I CONTROL [initandlisten] targetMinOS: Windows 7/Wir
2020-05-10T07:03:43.912+0200 I CONTROL [initandlisten] db version v4.2.5
2020-05-10T07:03:43.912+0200 I CONTROL [initandlisten] git version: 2261270b51ea31
```

2). Importo archivo en la base de datos llamada 'fifa_db'

```
C:\> Command Prompt - mongo
Microsoft Windows [Version 10.0.18362.778]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\pablo>cd C:\Program Files\MongoDB\Server\4.2\bin

C:\Program Files\MongoDB\Server\4.2\bin>mongoimport --type csv -d fifa_db -c fifa --headerline --drop fifa.csv
2020-05-10T07:33:13.201+0200 connected to: mongodb://localhost/
2020-05-10T07:33:13.202+0200 dropping: fifa_db.fifa
2020-05-10T07:33:13.734+0200 18207 document(s) imported successfully. 0 document(s) failed to import.
```

3). Compruebo que el dataset ha sido importado correctamente:

3.1). Cargo mongo

```
C:\Program Files\MongoDB\Server\4.2\bin>mongo
MongoDB shell version v4.2.5
connecting to: mongodb://127.0.0.1:27017/?compressors=disabled&gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("1f9d7c53-b720-4c65-809f-6be6d73fb7fa") }
MongoDB server version: 4.2.5
Server has startup warnings:
2020-04-17T14:04:06.577+0200 I CONTROL [initandlisten]
2020-04-17T14:04:06.577+0200 I CONTROL [initandlisten] ** WARNING: Access control is not enabled for the database.
2020-04-17T14:04:06.577+0200 I CONTROL [initandlisten] ** Read and write access to data and configuration is
unrestricted.
2020-04-17T14:04:06.577+0200 I CONTROL [initandlisten]
---
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).

The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.

To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
---
```

3.2). Elijo la base de datos donde está guardado y hago consulta de comprobación.

```
> use fifa_db
switched to db fifa_db
> db.fifa.find().limit(1)
{ "_id" : ObjectId("5eb792190513981295622674"), "" : 0, "ID" : 158023, "Name" : "L. Messi", "Age" : 31, "Nationality" :
"Argentina", "Overall" : 94, "Potential" : 94, "Club" : "FC Barcelona", "Value (M€)" : 110.5, "Wage (K€)" : 565, "Preferred Foot" : "Left", "International Reputation" : 5, "Body Type" : "Messi", "Position" : "RF", "Release Clause (M€)" : 226.5 }
> _
```

Consultas

// Comprobación inicial

db.fifa.find()

//1) Agrupar el número de jugadores con una puntuación por encima de 90, por nacionalidad.

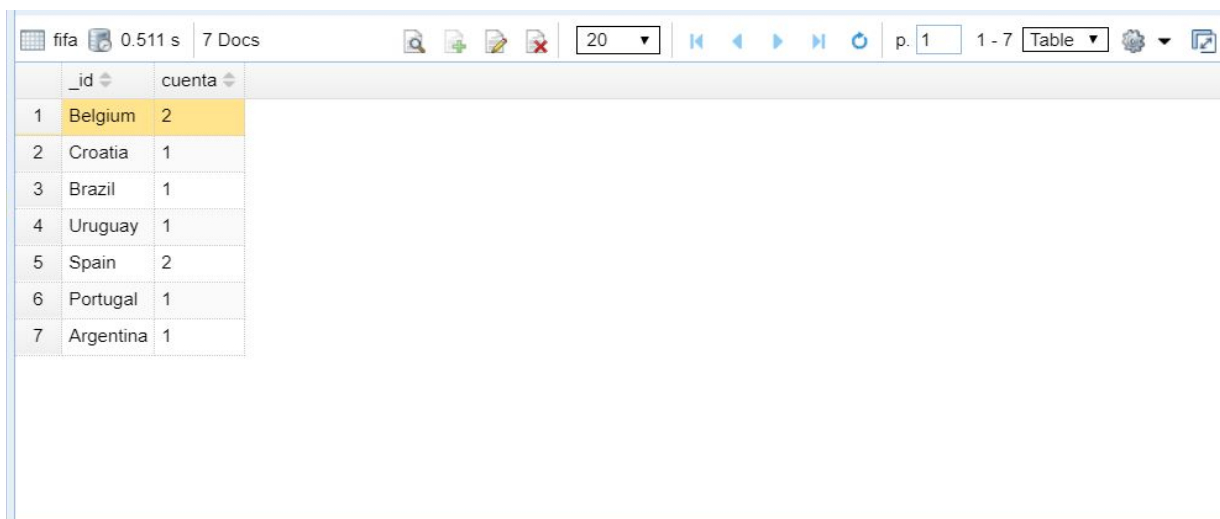
var fase1 = { \$match: { "Overall": { "\$gt": 90 } } }

var fase2 = { \$group: { "_id": "\$Nationality", "cuenta": { \$sum: 1 } } }

var fase3 = { \$sort: { "count" : -1 } }

var etapas = [fase1, fase2, fase3]

db.fifa.aggregate(etapas)



	_id	cuenta
1	Belgium	2
2	Croatia	1
3	Brazil	1
4	Uruguay	1
5	Spain	2
6	Portugal	1
7	Argentina	1

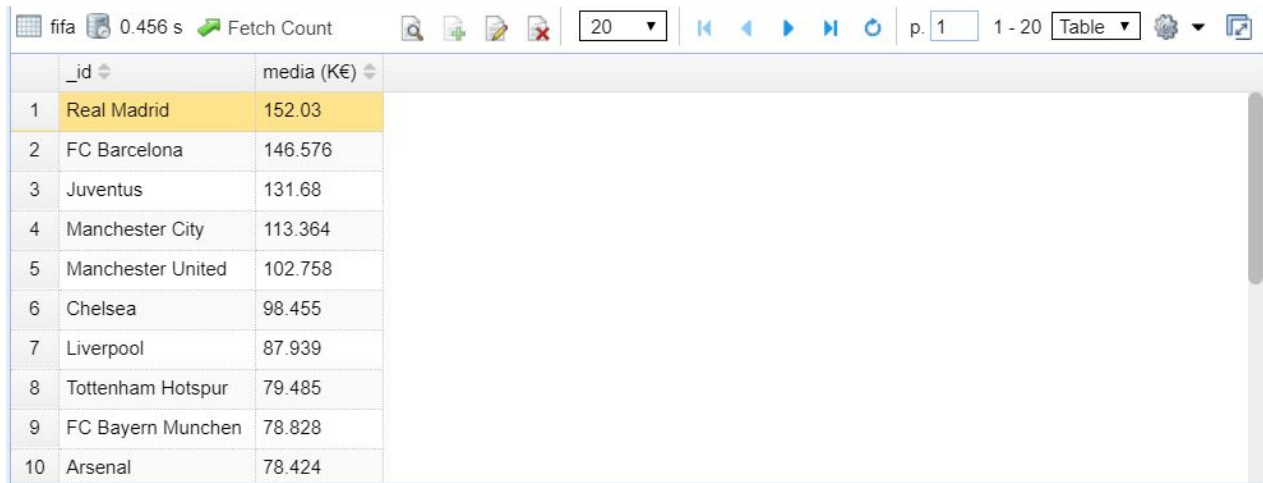
//2 Media de salario por equipo de fútbol. Ordenar de más a menos.

var fase1 = { \$group: { "_id": "\$Club", "media (K€)": { '\$avg': '\$Wage (K€)' } } }

var fase2 = { \$sort: { "media (K€)" : -1 } }

var etapas = [fase1, fase2]

db.fifa.aggregate(etapas)



Query result showing a list of football clubs and their average media value in thousands of euros (K€).

	_id	media (K€)
1	Real Madrid	152.03
2	FC Barcelona	146.576
3	Juventus	131.68
4	Manchester City	113.364
5	Manchester United	102.758
6	Chelsea	98.455
7	Liverpool	87.939
8	Tottenham Hotspur	79.485
9	FC Bayern Munchen	78.828
10	Arsenal	78.424

//3 Posición en el campo más común de los jugadores de nacionalidad Francesa (Delantero = Striker (ST)).

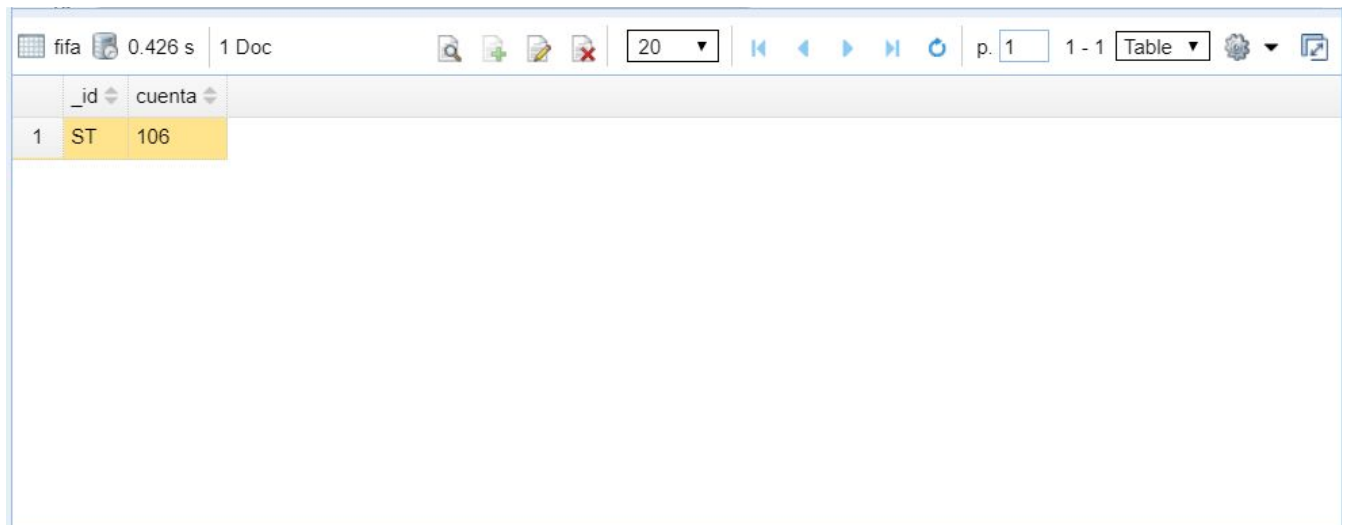
```
var fase1= { $match: { "Nationality": "France" } }
```

```
var fase2= { $group: { "_id": "$Position", "cuenta": { $sum: 1 } } }
```

```
var fase3= { $sort: { "cuenta" : -1 } }
```

```
var etapas= [ fase1, fase2, fase3 ]
```

db.fifa.aggregate(etapas).limit(1)



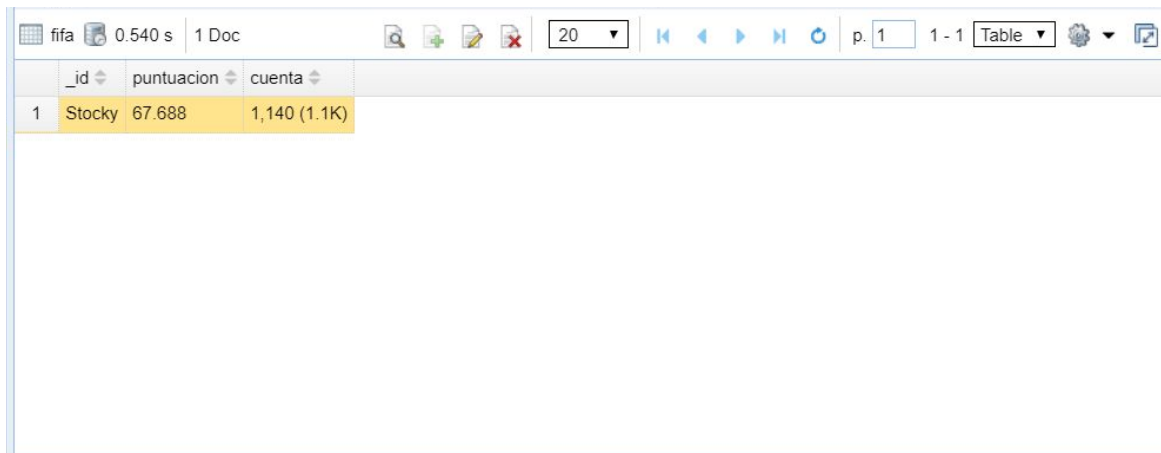
Query result showing the most common position for French players.

	_id	cuenta
1	ST	106

//4 Tipo de cuerpo que tiene más de un jugador y que tiene la mejor puntuación en 'Overall'

```
var fase1 = { $group: { "_id": "$Body Type", "puntuacion": { '$avg': '$Overall' },  
"cuenta": { $sum: 1 } } }  
var fase2 = { $match: { "cuenta" : { "$gt": 1 } } }  
var fase3 = { $sort: { "puntuacion" : -1 } }  
var etapas= [ fase1, fase2, fase3 ]
```

db.fifa.aggregate(etapas).limit(1)



_id	Stocky	puntuacion	cuenta
1	Stocky	67.688	1,140 (1.1K)

//5 Salario de los jugadores Españoles por posición y pierna buena. Ordenar de más salario a menos y añadir número de jugadores por categoría.

```
var fase1= { $match: { "Nationality": "Spain" } }  
var fase2= { $group: { "_id": { "Position": "$Position", "Preferred Foot": "$Preferred  
Foot" }, "media (K€)": { '$avg': '$Wage (K€)' }, "número de jugadores en categoría": {  
$sum: 1 } } }  
var fase3= { $sort: { "media (K€)" : -1 } }  
var etapas= [ fase1, fase2, fase3 ]
```

db.fifa.aggregate(etapas)

fifa

0.421 s

Fetch Count

20

p. 1

1 - 20

Table

	_id		media (K€)	número de jugadores en categoría
	Position	Preferred Foot		
1	RW	Left	66.5	6
2	LW	Right	53.125	8
3	RW	Right	52.667	9
4	LCM	Left	43.125	8
5	LWB	Right	34.5	2
6	RCM	Left	32.8	5
7	RCB	Right	32.412	34
8	CF	Right	31	4
9	LB	Right	30	1

//6 Países con mayor número de jugadores que tienen un salario superior a 20K. Mostrar media de edad, número de jugadores por categoría y salario medio.

var fase1= { \$match: { "Wage (K€)": { "\$gt":20 } } }

var fase2= { \$group: { "_id": "\$Nationality", "media edad": { '\$avg': '\$Age' }, "número de jugadores en categoría": { '\$sum': 1 }, "salario medio": { '\$avg': '\$Wage (K€)' } } }

var fase3= { \$sort: { "número de jugadores en categoría" : -1 } }

var etapas= [fase1, fase2, fase3]

db.fifa.aggregate(etapas)

fifa

0.473 s

Fetch Count

50

p.

1

1 - 50

Table

	_id	media edad	número de jugadores en categoría	salario medio
1	Spain	26.873	228	55.092
2	England	26.022	226	48.075
3	Brazil	27.733	172	54.651
4	France	26.135	171	52.918
5	Germany	26.444	142	50.063
6	Italy	26.683	139	49.719
7	Argentina	28.422	116	49.784
8	Netherlands	26.317	60	44.567
9	Portugal	25.875	48	59.521
10	Turkey	27.429	42	41.286

//7 Jugador de edad superior a 39 con más reputación internacional, y dentro de los que tienen más reputación internacional, el que tenga más salario.

```
db.fifa.find({"Age": {'$gt':39}}).sort({'International Reputation':-1, 'Wage (K€)':-1}).limit(1)
```

	_id ↕	ID ↕	Name ↕	Age ↕	Nationality ↕	Overall ↕	Potential ↕	Club ↕	Value (M€)	
1	ObjectId("5eb79219051398129562269f")	41	1179	G. Buffon	40	Italy	88	88	Paris Saint-Germain	4

//8 país que tiene más de 100 jugadores y cuyos jugadores reciben el salario más bajo y país con jugadores con salarios más altos.

```
var fase1 = { $group: { "_id": "$Nationality", "media salarios (K€)": { '$avg': '$Wage (K€)' }, "número de jugadores en país": { '$sum': 1 } } }
```

```
var fase2 = { $match: { "número de jugadores en país" : { "$gt":100 } } }
```

```
var fase3 = { $sort: { "media salarios (K€)": -1 } }
```

```
var etapas= [ fase1, fase2, fase3 ]
```

```
db.fifa.aggregate( etapas ).limit(1) //max
```

fifa	0.492 s	1 Doc	20	p. 1	1 - 1	Table
_id	media salarios (K€)	número de jugadores en país				
1	Croatia	21.683	126			

db.fifa.aggregate(etapas).skip(db.fifa.aggregate(etapas).count()-1) //min

fifa	0.506 s	1 Doc	20	p. 1	1 - 1	Table
_id	media salarios (K€)	número de jugadores en país				
1	Korea Republic	3.066	335			

//9 (nombre, edad, nacionalidad, overall, potential, club) del jugador de menor edad con mayor potencial.

**db.fifa.find({}, {'_id':0,'Name':1,'Age':1,'Nationality':1,'Overall':1,
'Potential':1,'Club':1}).sort({"Age":1,'Potential':-1}).limit(1)**

fifa	0.483 s	1 Doc	20	p. 1	1 - 1	Table
	Name	Age	Nationality	Overall	Potential	Club
1	W. Geubbels	16	France	64	86	AS Monaco

//10 De los clubes de Inglaterra, seleccionar el club con más jugadores zurdos.

```
var fase1 = { $match: { "Nationality" : "England", "Preferred Foot" : "Left" } }
```

```
var fase2 = { $group: { "_id": "$Club", 'count':{ '$sum':1 } } }
```

```
var fase3 = { $sort: { 'count': -1 } }
```

```
var etapas= [ fase1, fase2, fase3 ]
```

```
db.fifa.aggregate( etapas ).limit(1)
```

fifa

0.501 s

1 Doc

20

p.

1

1 - 1

Table

	_id	count
1	Doncaster Rovers	9

//11 Imaginemos que queremos fichar a algún jugador: De los jugadores menores de 21 con salario superior a cero, encontrar aquellos con menor valor en el mercado pero mayor potencial.

```
db.fifa.find({'Wage (K€)': {'$gt': 0}, "Age" : {'$lt': 21}},
{'_id': 0, 'ID': 0}).sort({'Value (M€)': 1, 'Potential': -1 })
```

fifa 0.497 s Fetch Count 50 p. 1 1 - 50 Table									
	Name	Age	Nationality	Overall	Potential	Club	Value (M€)	Wa	
1	9,935 (9.9K)	C. Fruchtl	18	Germany	65	86	FC Bayern Munchen	1	
2	11,457 (11.5K)	W. Geubbels	16	France	64	86	AS Monaco	1	
3	10,526 (10.5K)	Chumi	19	Spain	65	82	FC Barcelona	1	
4	10,900 (10.9K)	T. Edwards	19	England	65	82	Stoke City	1	
5	10,220 (10.2K)	S. Niakate	18	France	65	81	Valenciennes FC	1	
6	11,002 (11.0K)	K. Soni	20	Cameroon	64	81	Girona FC	1	
7	11,835 (11.8K)	Kuki Zalazar	20	Spain	64	80	Real Valladolid CF	1	
8	9,061 (9.1K)	G. Kobel	20	Switzerland	66	79	TSG 1899 Hoffenheim	1	
9	9,446 (9.4K)	A. Maisonnial	20	France	66	79	FC Sion	1	
10	9,962 (10.0K)	M. Bakker	18	Netherlands	65	79	Ajax	1	

//12 Otra forma de encontrar jugadores con potencial es identificando los países que tienen a los jugadores con mejor puntuación y por debajo de 21 años (si tienen más edad, el precio aumenta significativamente). Mostrar potencial medio y número de jugadores por categoría.

```
var fase1 = { $match: { "Age" : {'$lt': 21} } }
var fase2 = { $group: { "_id": "$Nationality", 'media de potencial': {'$avg': '$Potential'},
'numero de jugadores': {'$sum': 1} } }
var fase3 = { $sort: {'media de potencial': -1} }
var etapas= [ fase1, fase2, fase3 ]
```

```
db.fifa.aggregate( etapas )
```

	_id	media de potencial	numero de jugadores
1	Cape Verde	87	1
2	Burkina Faso	84	1
3	Korea DPR	83	1
4	Ukraine	79.273	11
5	Portugal	78.606	33
6	Brazil	78.583	36
7	Algeria	78.5	2
8	Uruguay	78.417	12
9	Morocco	78.4	10
10	Mali	78.333	6

//13 Crear un nuevo campo llamada “Informacion personal” que incluye las valores de 'Name', 'Age' y 'Nationality'.

```
var juntar = { '$concat': ['$Name', ", ", {$toString: "$Age"}, ", ", '$Nationality'] }
```

```
var fase1 = { '$addFields': { "Informacion Personal": juntar } }
```

```
var project1 = { "$project" : { "_id":0, "Name":1, "Age":1, "Nationality":1, "Overall":1, "Wage (K€)":1, "Informacion Personal":1 } }
```

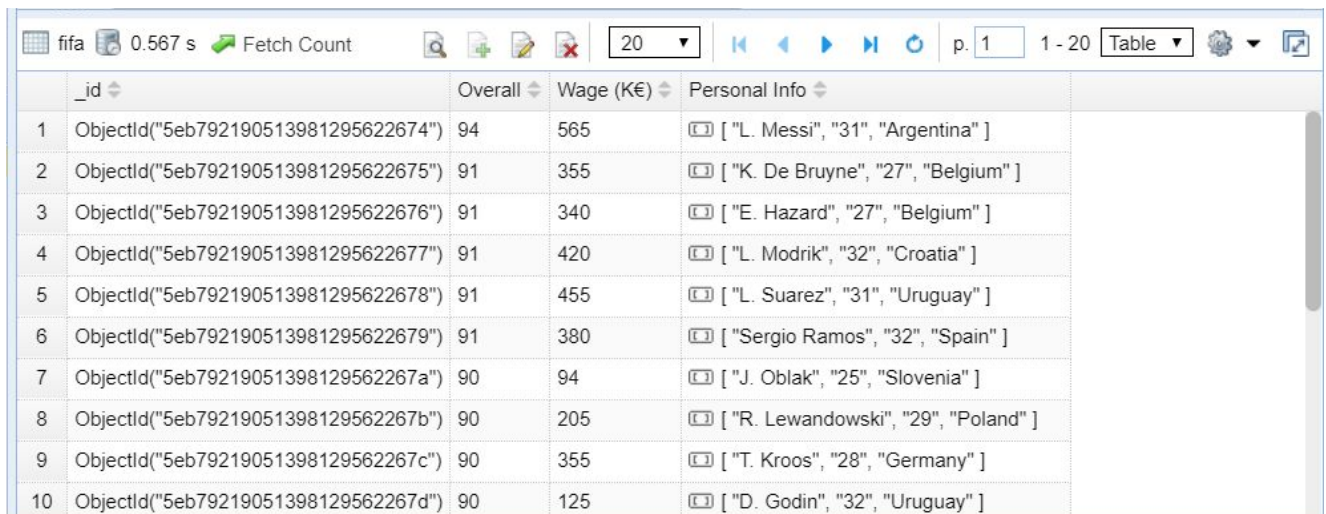
db.fifa.aggregate([fase1, project1])

fifa 0.536 s Fetch Count						
	Name	Age	Nationality	Overall	Wage (K€)	Informacion Personal
1	L. Messi	31	Argentina	94	565	L. Messi, 31, Argentina
2	K. De Bruyne	27	Belgium	91	355	K. De Bruyne, 27, Belgium
3	E. Hazard	27	Belgium	91	340	E. Hazard, 27, Belgium
4	L. Modrik	32	Croatia	91	420	L. Modrik, 32, Croatia
5	L. Suarez	31	Uruguay	91	455	L. Suarez, 31, Uruguay
6	Sergio Ramos	32	Spain	91	380	Sergio Ramos, 32, Spain
7	J. Oblak	25	Slovenia	90	94	J. Oblak, 25, Slovenia
8	R. Lewandowski	29	Poland	90	205	R. Lewandowski, 29, Poland
9	T. Kroos	28	Germany	90	355	T. Kroos, 28, Germany
10	D. Godin	32	Uruguay	90	125	D. Godin, 32, Uruguay

//14 Crear el campo del ejercicio anterior usando arrays..

```
var fase1 = { '$project' : { 'Personal Info' : [ "$Name", { $toString: "$Age"},  
"$Nationality" ], "Overall":1, "Wage (K€)":1 } }
```

db.fifa.aggregate([fase1])



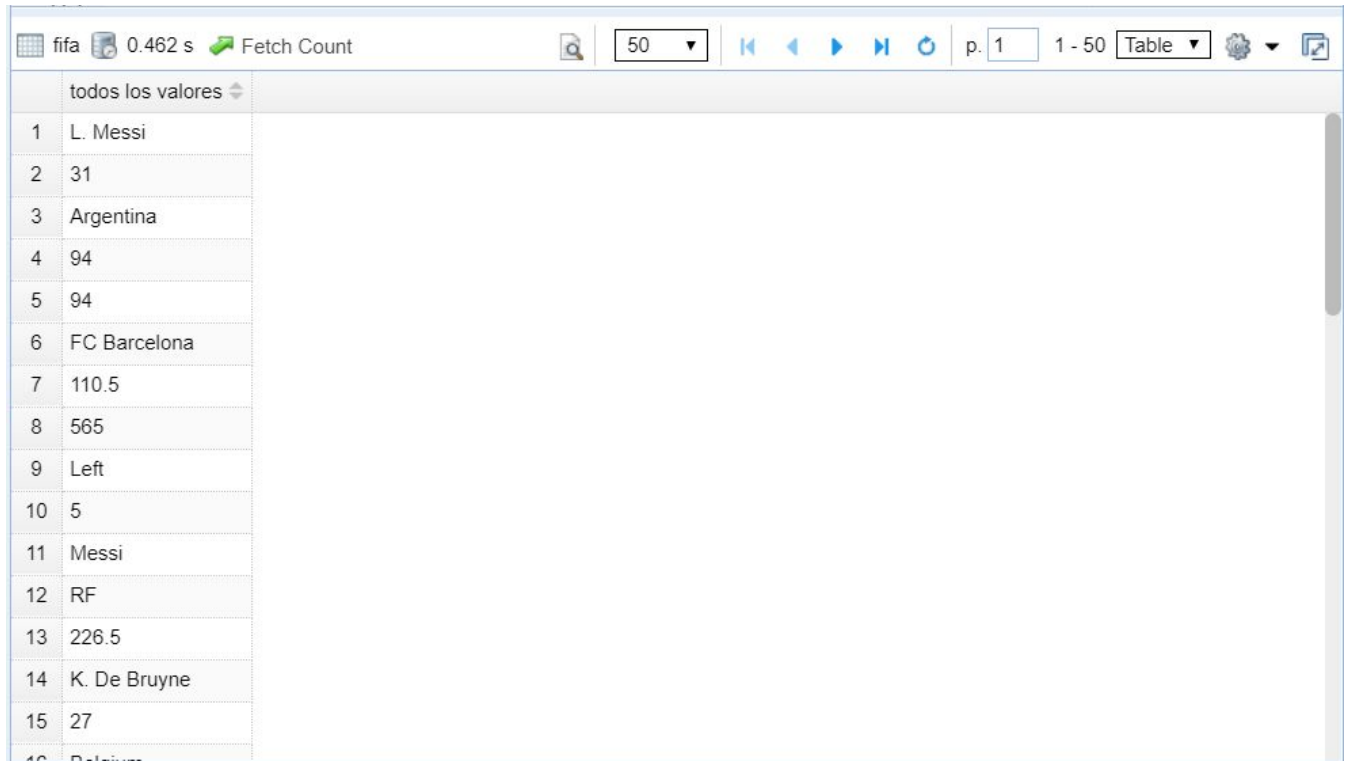
	_id	Overall	Wage (K€)	Personal Info
1	ObjectId("5eb792190513981295622674")	94	565	["L. Messi", "31", "Argentina"]
2	ObjectId("5eb792190513981295622675")	91	355	["K. De Bruyne", "27", "Belgium"]
3	ObjectId("5eb792190513981295622676")	91	340	["E. Hazard", "27", "Belgium"]
4	ObjectId("5eb792190513981295622677")	91	420	["L. Modrik", "32", "Croatia"]
5	ObjectId("5eb792190513981295622678")	91	455	["L. Suarez", "31", "Uruguay"]
6	ObjectId("5eb792190513981295622679")	91	380	["Sergio Ramos", "32", "Spain"]
7	ObjectId("5eb79219051398129562267a")	90	94	["J. Oblak", "25", "Slovenia"]
8	ObjectId("5eb79219051398129562267b")	90	205	["R. Lewandowski", "29", "Poland"]
9	ObjectId("5eb79219051398129562267c")	90	355	["T. Kroos", "28", "Germany"]
10	ObjectId("5eb79219051398129562267d")	90	125	["D. Godin", "32", "Uruguay"]

//15 Crear un único campo que incluya todos los valores de cada jugador y los almacene en una array. Una vez hecha esta transformación, descomponer la array usando unwind para generar un campo con todos los valores por separado.

```
var fase1 = { '$project' : { 'todos los valores' : [ "$Name", { $toString: "$Age"},  
'$Nationality', { $toString: '$Overall'}, { $toString: '$Potential'}, '$Club',  
{ $toString: '$Value (M€)' }, { $toString: '$Wage (K€)' }, '$Preferred Foot',  
{ $toString: '$International Reputation' }, '$Body Type', '$Position',  
{ $toString: '$Release Clause (M€)' } ], '_id':0 } }
```

```
var fase2 = { "$unwind" : "$todos los valores" }
```

db.fifa.aggregate([fase1, fase2])



The screenshot shows a web interface for a MongoDB aggregation pipeline. The top bar indicates the database is 'fifa', the execution time is '0.462 s', and the 'Fetch Count' is 50. The results are displayed in a table with 15 rows. The first column contains an index from 1 to 15, and the second column contains the aggregated data for each row. The data is as follows:


	todos los valores
1	L. Messi
2	31
3	Argentina
4	94
5	94
6	FC Barcelona
7	110.5
8	565
9	Left
10	5
11	Messi
12	RF
13	226.5
14	K. De Bruyne
15	27

(Comprobación)

```
var fase1 = { '$project' : { 'todos los valores' : [ "$Name", { $toString: "$Age" },  
'$Nationality', { $toString: '$Overall' }, { $toString: '$Potential' }, '$Club',  
{ $toString: '$Value (M€)' }, { $toString: '$Wage (K€)' }, '$Preferred Foot',  
{ $toString: '$International Reputation' }, '$Body Type', '$Position',  
{ $toString: '$Release Clause (M€)' } ], '_id':0 } } // , "Overall":1, "Wage (K€)":1 } }
```

```
var fase2 = { "$unwind" : "$todos los valores" }
```

db.fifa.aggregate([fase1, fase2]).count()

	0.643 s
1	236691

db.fifa.find().count()

	0.460 s
1	18207

Conclusiones:

- Los jugadores zurdos que son delanteros lateral derecho, son los que significativamente reciben los mejores salarios en España.
- Gigi Buffon es el jugador con edad superior a 39, que recibe el mejor salario.
- La edad es una variable significativa a la hora de determinar el salario del jugador. La edad media de los jugadores que ganan más de 20.000 euros al mes era superior a 25 años en la gran mayoría de los países.
- En promedio, de los países con más de 100 jugadores profesionales, los jugadores Coreanos son los peor pagados, mientras que los nacionales de Croacia reciben los salarios más altos.
- En caso de que un club desee fichar a un nuevo jugador, ha de tener en cuenta que:
 - El jugador de menor edad con mayor potencial es 'W. Geubbels', tiene 16 años y es delantero en el Mónaco.
 - Si se analiza por nacionalidad, los jugadores de Cabo Verde tienen el mejor potencial medio.

- Con respecto a Mongo, el uso de 'aggregate()' fue especialmente útil para aquellas consultas complejas (la mayoría de ellas agrupaciones) que requerían dividir dicha consulta en varias fases.
- Es importante destacar que la estructura de una colección en Mongo puede ser alterada. Por ejemplo, los campos pueden ser fusionados por medio de arrays.
- En caso de que se quiera reducir una array a un único valor, esto también se puede llevar a cabo gracias a la función 'unwind'.

Como conclusión, MongoDB ofrece una gran flexibilidad para organizar y hacer modificaciones en las colecciones de datos.

Este es el final. ¡Muchas gracias!