

PRÀCTICA 2 : INTRODUCCIÓ AL MONGODB

1) PROJECTAR CAMPS:

```
db.inventory.insert( [
  { item: "journal", status: "A", size: { h: 14, w: 21, uom: "cm" }, instock: [ { warehouse:
"A", qty: 5 } ] },
  { item: "notebook", status: "A", size: { h: 8.5, w: 11, uom: "in" }, instock: [ {
warehouse: "C", qty: 5 } ] },
  { item: "paper", status: "D", size: { h: 8.5, w: 11, uom: "in" }, instock: [ { warehouse:
"A", qty: 60 } ] },
  { item: "planner", status: "D", size: { h: 22.85, w: 30, uom: "cm" }, instock: [ {
warehouse: "A", qty: 40 } ] },
  { item: "postcard", status: "A", size: { h: 10, w: 15.25, uom: "cm" }, instock: [ {
warehouse: "B", qty: 15 }, { warehouse: "C", qty: 35 } ] }
]);
```

The following example returns all fields from all documents in the inventory collection where the status equals "A":

Return All Fields:

```
db.inventory.find( { status: "A" } )
```

Return the Specified Fields and the `_id` Field:

```
db.inventory.find( { status: "A" }, { item: 1, status: 1 } )
```

Suppress `_id` Field:

```
db.inventory.find( { status: "A" }, { item: 1, status: 1, _id: 0 } )
```

Return All But the Excluded Fields:

```
db.inventory.find( { status: "A" }, { status: 0, instock: 0 } )
```

```
db.inventory.find(
  { status: "A" },
  { item: 1, status: 1, "size.uom": 1 }
)
```

```
db.inventory.find(
  { status: "A" },
  { "size.uom": 0 }
)
```

Project Specific Array Elements in the Returned Array

For fields that contain arrays, MongoDB provides the following projection operators:

```
$elemMatch,
$slice
$.
```

The following example uses the `$slice` projection operator to return just the last element in the `instock` array:

```
db.inventory.find( { status: "A" }, { name: 1, status: 1, instock: { $slice: -1 } } )
```

`$elemMatch`, `$slice`, and `$` are the only way to project specific elements to include in the returned array. For instance, you cannot project specific array elements using the array index; e.g. `{ "instock.0": 1 }` projection will not project the array with the first element.

2) ORDENAR RESULTATS

```
db.inventory.find( {}, {item:1} ).sort({item:-1})
```

```
db.inventory.find( {}, {item:1, status:1} ).sort({item:-1, status:1})
```

3) VALORS NULL o PROPIETATS INNEXISTENTS

```
db.inventory.insert([
  { _id: 1, item: null },
  { _id: 2 }
])

db.inventory.find( { item: null } )

db.inventory.find( { item : { $type: 10 } } )

db.inventory.find( { item : { $exists: false } } )
```

4) CURSORS

```
var myCursor = db.inventory.find( { status: "A"} );
myCursor
----

var myCursor = db.inventory.find( { status: "A"} );

while (myCursor.hasNext()) {
  print(tojson(myCursor.next()));
}

----

var myCursor = db.inventory.find( { status: "A"} );

while (myCursor.hasNext()) {
  printjson(myCursor.next());
}

----

var myCursor = db.inventory.find( { status: "A"} );

myCursor.forEach(printjson);

----

var myCursor = db.inventory.find( { status: "A" } );
var documentArray = myCursor.toArray();
var myDocument = documentArray[3];
myDocument
```