

CURSO : Base de Datos Avanzado II (2398)
PROFESOR : JESUS CANALES GUANDO
CICLO : Cuarto
SECCIÓN : T4CJ
GRUPO : 01
FECHA : Miércoles 23 de abril del 2025
DURACIÓN : 90 minutos

NOTA

ALUMNO (A) : Ronald Jaime Duran

EXAMEN DE LABORATORIO 2 (T2)

Consideraciones generales:

- Considerar el orden, la limpieza y la claridad de las respuestas.
- NO está permitido el uso o consulta de cuadernos, separatas, libros o cualquier material de la asignatura durante el desarrollo de la evaluación. El desarrollo es estrictamente personal.
- Todas las **respuestas** deberán aparecer dentro del espacio de cada pregunta, dichas **respuestas** deben evidenciar el script y sus resultados.
- Puede utilizar Mongo Shell, Compass o Studio 3T.
- Toda copia parcial o total será amonestado con la nota 99, **sin** posibilidad a recuperación.
- El desarrollo del presente examen no está permitido enviar por correo, sólo es permitido por el Blackboard.

LOGRO DE LA EVALUACIÓN

Tenga presente que el logro, es que usted, aplique operadores en la implementación de consultas de documentos, así como el uso de condicionales y ordenamientos.

CONSOLIDADO

Pregunta	Puntaje		Llenar solo en caso de Recalificación justificada	
	Máximo	Obtenido	Sustento	Puntaje
1	04			
2	04			
3	06			
4	06			
	20			
Nota Recalificada				

NOTA: Para evaluar su desarrollo debe pegar su script y captura de pantalla mostrando el resultado de ejecución como evidencia por cada pregunta. En toda captura de su desarrollo debe figurar sus nombres completos, por ejemplo "Por Juan Pérez". Asegurarse que toda respuesta sea legible.

Implemente la base de datos NoSQL en **MongoDB** llamada **"UrbanStyle_Store"** con las colecciones: **clientes, productos, vendedores y sucursales** con sus documentos. **Script proporcionado por el docente.**

Implementar consultas de documentos de las colecciones:

PREGUNTA 1.-

Obtenga de acuerdo al caso las consultas solicitadas de la colección **"clients"**, aplicando:

Operadores de comparación: \$eq, \$gt, \$gte, \$in, \$lt, \$lte, \$ne, \$nin u otros:

Id	Descripción	Puntaje
1	Muestre los clientes que no residan en Breña ni en La Molina. Ordene en descendente por apellidos. Sólo debe mostrar los campos: nombres, apellidos, email e historial de compras.	
Script:	<pre> use("UrbanStyle_store"); db.clients.find({ direccion: { \$not: { \$regex: /(la molina breña)/i } } }, { _id:0, nombre: 1, apellidos: 1, email:1, telefono:1, direccion:1, preferencias:1, historialCompras: 1 }).sort({nombre: -1}); </pre>	1

Captura:

```
1  [
2    {
3      "nombre": "Roberto",
4      "apellidos": "López",
5      "email": "roberto.lopez@outlook.com",
6      "telefono": "941456987",
7      "direccion": "Avenida Central 90, Miraflores",
8      "preferencias": [
9        "Caballero",
10       "Dama"
11     ],
12     "historialCompras": [
13       {
14         "producto": "Camisa Formal",
15         "fecha": {
16           "$date": "2024-03-01T00:00:00Z"
17         },
18         "cantidad": 2
19       },
20       {
21         "producto": "Blusa Casual",
22         "fecha": {
23           "$date": "2024-03-10T00:00:00Z"
24         },
25         "cantidad": 1
26       }
27     ]
28   },
29   {
30     "nombre": "Patricia",
31     "apellidos": "Núñez",
32     "email": "patricia.nunez@yahoo.com",
33     "telefono": "989123654",
34     "direccion": "Calle Flores 78, San Martín de Porres",
35     "preferencias": [
36       "Niño"
37     ],
38     "historialCompras": [
39       {
40         "producto": "Conjunto Deportivo",
41         "fecha": {
42           "$date": "2024-02-25T00:00:00Z"
43         },
44         "cantidad": 2
45       },
46       {
47         "producto": "Pantalón Niño",
48         "fecha": {
49           "$date": "2024-03-05T00:00:00Z"
50         },
51         "cantidad": 1
52       }
53     ]
54   },
55   {
56     "nombre": "Lucía",
57     "apellidos": "Fernández",
58     "email": "lucia.fernandez@outlook.com",
59     "telefono": "933398746",
60     "direccion": "Avenida Sol 456, Miraflores",
61     "preferencias": [
62       "Dama",
63       "Niño"
64     ],
65     "historialCompras": [
66       {
67         "producto": "Falda de Verano",
68         "fecha": {
69           "$date": "2024-03-15T00:00:00Z"
70         },
71         "cantidad": 1
72       },
73       {
74         "producto": "Conjunto Deportivo",
75         "fecha": {
76           "$date": "2024-04-05T00:00:00Z"
77         },
78         "cantidad": 1
79       }
80     ]
81   },
82 ]
```

1

```
1 {
2   "nombre": "Laura",
3   "apellidos": "Martínez",
4   "email": "laura.martinez@yahoo.com",
5   "telefono": "993654789",
6   "direccion": "Calle Jardines 12, San Borja",
7   "preferencias": [
8     "Dama",
9     "Niño"
10  ],
11  "historialCompras": [
12    {
13      "producto": "Vestido de Verano",
14      "fecha": {
15        "$date": "2024-02-15T00:00:00Z"
16      },
17      "cantidad": 1
18    },
19    {
20      "producto": "Conjunto Deportivo",
21      "fecha": {
22        "$date": "2024-03-05T00:00:00Z"
23      },
24      "cantidad": 2
25    }
26  ]
27 },
28 {
29   "nombre": "Alberto",
30   "apellidos": "Gómez",
31   "email": "alberto.gomez@outlook.com",
32   "telefono": "964789321",
33   "direccion": "Calle Águilas 56, San Luis",
34   "preferencias": [
35     "Caballero"
36  ],
37  "historialCompras": [
38    {
39      "producto": "Camisa Casual",
40      "fecha": {
41        "$date": "2024-02-20T00:00:00Z"
42      },
43      "cantidad": 1
44    },
45    {
46      "producto": "Sudadera Deportiva",
47      "fecha": {
48        "$date": "2024-03-10T00:00:00Z"
49      },
50      "cantidad": 3
51    }
52  ]
53 }
54 ]
```

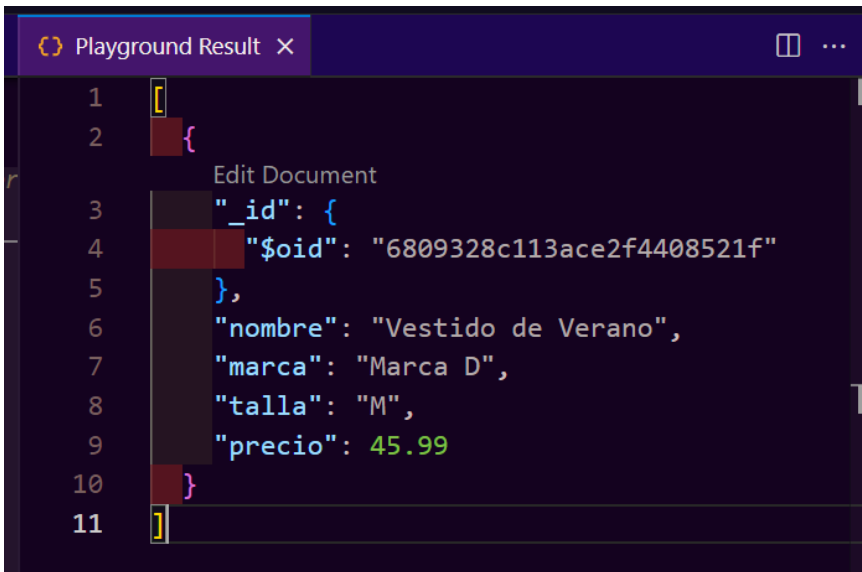
2	Muestre los clientes que tienen como parte de sus preferencias ropa de niños. Considere visualizar sólo los campos: nombres, apellidos, fecha de registro y preferencias. Ordene por fecha de registros de manera ascendente.	
Script:	<pre> use("UrbanStyle_store"); db.clients.find({ preferencias: { \$in: ["Niño"] } }, { _id: 0, nombre: 1, apellidos: 1, fechaRegistro: 1, preferencias:1 }).sort({ "fechaRegistro": 1 }) </pre>	1
Captura:		1



```
1  [  
2    {  
3      "nombre": "Patricia",  
4      "apellidos": "Núñez",  
5      "fechaRegistro": {  
6        "$date": "2024-01-18T00:00:00Z"  
7      },  
8      "preferencias": [  
9        "Niño"  
10     ]  
11   },  
12   {  
13     "nombre": "Miguel",  
14     "apellidos": "Díaz",  
15     "fechaRegistro": {  
16       "$date": "2024-01-20T00:00:00Z"  
17     },  
18     "preferencias": [  
19       "Caballero",  
20       "Niño"  
21     ]  
22   },  
23   {  
24     "nombre": "Laura",  
25     "apellidos": "Martínez",  
26     "fechaRegistro": {  
27       "$date": "2024-01-25T00:00:00Z"  
28     },  
29     "preferencias": [  
30       "Dama",  
31       "Niño"  
32     ]  
33   },  
34   {  
35     "nombre": "Lucía",  
36     "apellidos": "Fernández",  
37     "fechaRegistro": {  
38       "$date": "2024-03-05T00:00:00Z"  
39     },  
40     "preferencias": [  
41       "Dama",  
42       "Niño"  
43     ]  
44   }  
45 ]
```

PREGUNTA 2.-

Recupere documentos solicitados de la colección “productos”, según sea el caso utilizando:
Operadores lógicos: \$and, \$or, \$not, \$nor, u otros:

Id	Descripción	Puntaje
1	Listar de los productos _id, nombre, marca, talla y precio de las prendas, filtre cuya sucursal sea “Sucursal Norte” y el precio sea mayor igual a 100.	
Script:	<pre>use("UrbanStyle_store"); db.products.find({ \$and: [{ sucursal: {\$eq: "Sucursal Norte" }}, { precio: {\$gte: 45}}] }, { _id: 1, nombre: 1, marca: 1, talla: 1, precio: 1 });</pre>	1
Captura:	<p>No había productos con precio >= a 100 y que sean de la sucursal norte Por lo que ajuste el precio a >= 45 para que se pueda ver el resultado</p> 	1

2	Mostrar los productos cuyo precio sea mayor a 55 o color "azul", no considerar de la temporada "Invierno"	
Script:	<pre> use("UrbanStyle_store"); db.products.find({ \$and: [{ "especificaciones.temporada": {\$ne: "Invierno"} }, { \$or: [{ precio: {\$gt: 55} }, { "especificaciones.color" : { \$eq: "Azul"} },] }] }); use("UrbanStyle_store"); db.products.find(); </pre>	1
Captura:		1


```

1  [
2  {
3    "_id": {
4      "$oid": "6809328c113ace2f4408521d"
5    },
6    "nombre": "Blusa Casual",
7    "marca": "Marca B",
8    "tipo": "Dama",
9    "talla": "S",
10   "precio": 25.99,
11   "cantidad": 150,
12   "fechaRegistro": {
13     "$date": "2024-01-12T00:00:00Z"
14   },
15   "sucursal": "Sucursal Sur",
16   "especificaciones": {
17     "material": "Poliéster",
18     "color": "Azul",
19     "temporada": "Otoño-Invierno"
20   }
21 },
22 {
23   "_id": {
24     "$oid": "6809328c113ace2f4408521e"
25   },
26   "nombre": "Pantalón Niño",
27   "marca": "Marca C",
28   "tipo": "Niño",
29   "talla": "L",
30   "precio": 20.99,
31   "cantidad": 120,
32   "fechaRegistro": {
33     "$date": "2024-01-15T00:00:00Z"
34   },
35   "sucursal": "Sucursal Este",
36   "especificaciones": {
37     "material": "Denim",
38     "color": "Azul",
39     "temporada": "Todo el año"
40   }
41 },
42 {
43   "_id": {
44     "$oid": "6809328c113ace2f44085223"
45   },
46   "nombre": "Camisa Casual",
47   "marca": "Marca H",
48   "tipo": "Caballero",
49   "talla": "L",
50   "precio": 28.99,
51   "cantidad": 110,
52   "fechaRegistro": {
53     "$date": "2024-04-05T00:00:00Z"
54   },
55   "sucursal": "Sucursal Norte",
56   "especificaciones": {
57     "material": "Lino",
58     "color": "Azul",
59     "temporada": "Primavera"
60   }
61 },
62 {
63   "_id": {
64     "$oid": "6809328c113ace2f44085224"
65   },
66   "nombre": "Conjunto Deportivo",
67   "marca": "Marca I",
68   "tipo": "Niño",
69   "talla": "L",
70   "precio": 55.99,
71   "cantidad": 65,
72   "fechaRegistro": {
73     "$date": "2024-04-20T00:00:00Z"
74   },
75   "sucursal": "Sucursal Oeste",
76   "especificaciones": {
77     "material": "Algodón",
78     "color": "Rojo",
79     "temporada": "Todo el año"
80   }
81 }
82 ]

```


--	--	--

PREGUNTA 3.-

Recupere documentos según sea el caso utilizando los métodos: sort(), limit(), skip(), \$and así como **Operadores de formación (array):** \$all, \$elemMatch, \$size, \$slice u otros métodos:

Id	Descripción	Puntaje
1	Utilizando la colección vendedores mostrar nombres, ventas mensuales y sede en la que labora, la lista debe mostrar después del tercer documento, filtre por sede que sea la "Sucursal Sur". Ordene por cualquier campo.	
Script:	<pre>use("UrbanStyle_store"); db.sellers.find({ sucursal: {\$eq: "Sucursal Sur"} }, { _id: 0, nombre: 1, "metas.ventasMensuales": 1, sucursal: 1 }).skip(3).sort({ nombre: 1});</pre>	1

<p>Captura:</p>	<p>Profesor la consulta sin skip() obtiene 3 documentos</p> <pre> 1 [2 { 3 "nombre": "Carlos", 4 "sucursal": "Sucursal Sur", 5 "metas": { 6 "ventasMensuales": 9500 7 } 8 }, 9 { 10 "nombre": "Fernando", 11 "sucursal": "Sucursal Sur", 12 "metas": { 13 "ventasMensuales": 9800 14 } 15 }, 16 { 17 "nombre": "Laura", 18 "sucursal": "Sucursal Sur", 19 "metas": { 20 "ventasMensuales": 12000 21 } 22 } 23] </pre> <p>pero al utilizar skip omite esos 3 documentos, dando así vacío como respuesta</p> <pre> 1 [] </pre>	<p>1</p>
<p>2</p>	<p>Listar sólo los nombres, apellidos y fecha de ingreso (sin “_id”), ordenada en descendente por apellidos, recuperar los 4 primeros documentos. Filtre sólo los vendedores cuya cantidad de clientes nuevos que atienden sean menores e iguales que 20.</p>	
<p>Script:</p>	<pre> db.sellers.find({ "metas.clientesNuevos": { \$lte: 20} }, { _id: 0, nombre: 1, apellidos: 1, fechaIngreso:1, }).sort({apellidos: -1}).limit(4); </pre>	<p>1</p>

Captura:	 <pre> 1 [2 { 3 "nombre": "María", 4 "apellidos": "López", 5 "fechaIngreso": { 6 "\$date": "2023-10-15T00:00:00Z" 7 } 8 }, 9 { 10 "nombre": "Fernando", 11 "apellidos": "Jiménez", 12 "fechaIngreso": { 13 "\$date": "2024-01-10T00:00:00Z" 14 } 15 }, 16 { 17 "nombre": "Diego", 18 "apellidos": "García", 19 "fechaIngreso": { 20 "\$date": "2023-08-25T00:00:00Z" 21 } 22 } 23] </pre>	1
3	Mostrar todos aquellos vendedores en los que dentro de su campo “ventas mensuales” sea mayor e igual que 10000 y menor que 11000. Ordene en descendente por nombre.	
Script:	<pre> use("UrbanStyle_store"); db.sellers.find({ \$and: [{ "metas.ventasMensuales": { \$gte: 10000} }, { "metas.ventasMensuales": { \$lt: 11000} }] }, { _id: 0, nombre: 1, apellidos: 1, "metas.ventasMensuales": 1 }).sort({ nombre: -1 }); </pre>	1

Captura:	 <p>En los documentos no hubieron Arrays, por lo que no fue necesario utilizar consultas de arrays</p>	1

PREGUNTA 4.-

Se tiene el siguiente comprobante como documento comercial, colección “factura” (no debe crear ninguna BD ni colección ni documentos, SÓLO UNA CONSULTA):

Rojo Polo Paella Inc.
Carretera Muelle 38
37531 Ávila, Ávila

FACTURAR A
Leda Villareal
Virgen Blanca 63
08759 Burgos, Burgos

ENVIAR A
Leda Villareal
Cercas Bajos 68
47300 Cádiz, Cádiz

N° DE FACTURA
ES-001
FECHA
29/01/2019
N° DE PEDIDO
1730/2019
FECHA
29/01/2019
VENCIMIENTO

Total Factura **199.65 €**

CANT.	DESCRIPCIÓN	PRECIO UNITARIO	IMPORTE
1	Talla pequeña traje de luces en rojo	100.00	100.00
2	Mui grande churrolito	25.00	50.00
3	Equipaje de Fútbol	5.00	15.00
	Subtotal		165.00
	IVA 21.0%		34.65

Laura García

CONDICIONES Y FORMA DE PAGO
El pago se realizará en un plazo de 15 días

Banco Santander
IBAN: ES12 3456 7891
SWIFT/BIC: ABCDESM1XXX

Requerimiento: Usted como responsable del análisis de datos semiestructurados en mongoDB para un proyecto de TI debe tomar en cuenta la distribución de la información de dicha boleta para luego crear un reporte donde considere:

- Visualizar al menos 5 campos.
- Dos filtros.
- Un ordenamiento.

Id	Descripción	Puntaje
Script:	<pre> /*se lista aquellos productos en el que el precio unitario sea >= 50 el total de factura sea < 100.50 ordenado en ascendente por total_factura **/ db.productos.find({ \$and: [{ precio_unitario: {\$gte: 50 } }, { total_factura: {\$lt: 100.50} }] }, { _id: 1, descripcion:1, precio_unitario: 1, importe: 1, total_factura:1, cant_productos:1 }).sort({total_factura: -1}) </pre>	6

Rúbrica de evaluación

Rúbrica de la Pregunta 1 (Condicionales con operadores lógicos y ordenamiento)				
Puntos	Excelente	Bueno	Regular	Deficiente
4	Implementa las consultas correctamente según requerimiento. Demuestra evidencia.	Implementa las consultas según requerimiento. Existe alguna inconsistencia. Demuestra evidencia.	Implementa las consultas según requerimiento. Posee dos a tres errores.	No desarrolla requerimiento alguno o todo presenta errores.
	4 puntos	2 puntos	1 punto	0 puntos
Rúbrica de la Pregunta 2 (Condicionales con operadores and y or)				
Puntos	Excelente	Bueno	Regular	Deficiente
4	Implementa las consultas correctamente según requerimiento. Demuestra evidencia.	Implementa las consultas según requerimiento. Existe alguna inconsistencia. Demuestra evidencia.	Implementa las consultas según requerimiento. Posee dos a tres errores.	No desarrolla requerimiento alguno o todo presenta errores.
	4 puntos	2 puntos	1 punto	0 puntos
Rúbrica de la Pregunta 3 (Condicionales con array o vectores)				
Puntos	Excelente	Bueno	Regular	Deficiente
6	Implementa las consultas correctamente según requerimiento. Demuestra evidencia.	Implementa las consultas según requerimiento. Existe alguna inconsistencia. Demuestra evidencia.	Implementa las consultas según requerimiento. Posee dos a tres errores.	No desarrolla requerimiento alguno o todo presenta errores.
	6 puntos	4 puntos	2 puntos	0 puntos
Rúbrica de la Pregunta 4 (Análisis de datos semiestructurados y consulta)				
Puntos	Excelente	Bueno	Regular	Deficiente
6	Implementa la consulta correctamente según requerimiento.	Implementa la consulta según requerimiento. Posee un error.	Implementa la consulta según requerimiento. Posee dos a tres errores.	No desarrolla requerimiento alguno o todo presenta errores.
	6 puntos	4 puntos	2 puntos	0 puntos