

## Universidad de las fuerzas armadas espe

**Nombre:** Ariel Ilumiquinga

**Fecha:**06/02/2026

### Actividad

La empresa TecnoMega requiere un prototipo de API para compra/venta de productos tecnológicos, usando Redis como base NoSQL clave-valor para almacenar catálogos y transacciones.

### Objetivo general

Construir una API REST en Node.js + Express que:

1. Use middleware para parsear JSON (express.json()).
2. Use middleware response-time para medir tiempo de respuesta.
3. Implemente endpoints SET y GET contra Redis.
4. Cargue 10 registros por cada “tabla” desde un archivo JSON (clave-valor) y los almacene en Redis.

### Modelo de datos (tablas en Redis)

Se simulan 4 “tablas” (colecciones):

1. clientes
2. productos
3. pedidos
4. detalle\_pedido
5. Si el estudiante quiere agregar más colecciones lo puede realizar

### Reglas mínimas

- **productos:** deben incluir código, nombre, categoría, precio, stock.
- **clientes:** deben incluir cedula o dni, nombres, email, teléfono, edad, genero
- **pedidos:** deben incluir Código , clienteld, fecha, subtotal, iva, total, estado.
- **detalle\_pedido:** debe incluir Código, productold, cantidad, detalle, precioUnit.

### Entregables

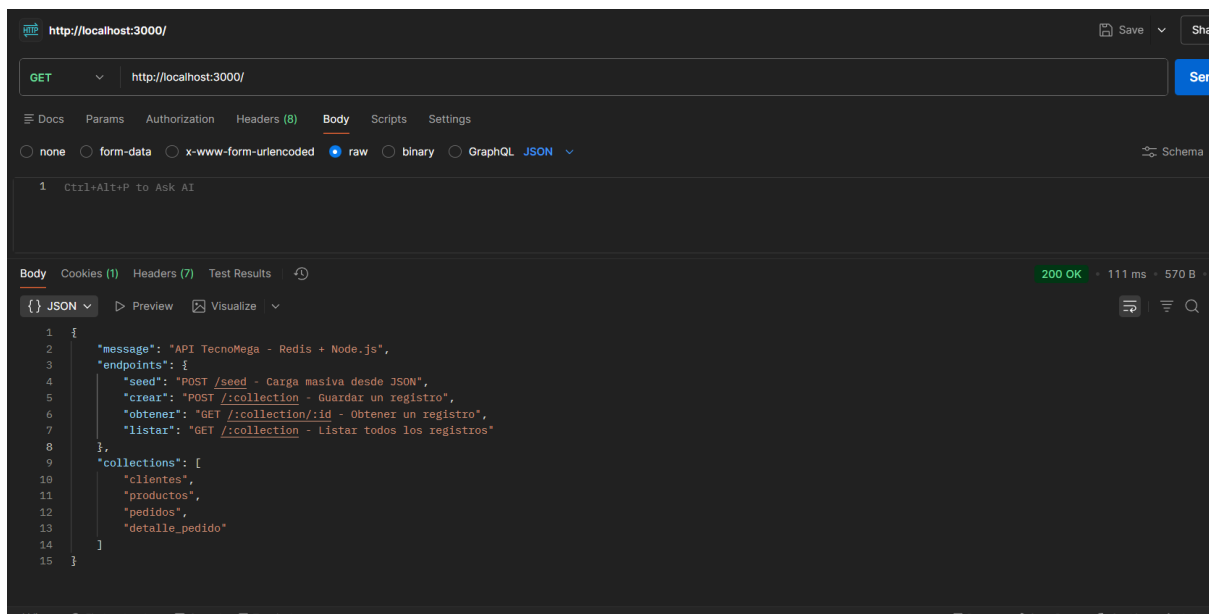
1. Repositorio con:
  - index.js (API)

- data/tecnomega.json (10 registros por tabla mínimo)
2. Evidencia:
    - Capturas de Postman/Thunder Client
    - Salida del tiempo de respuesta (Response-Time)

## DESARROLLO

### PRUEBA 1: Verifica que la API está corriendo

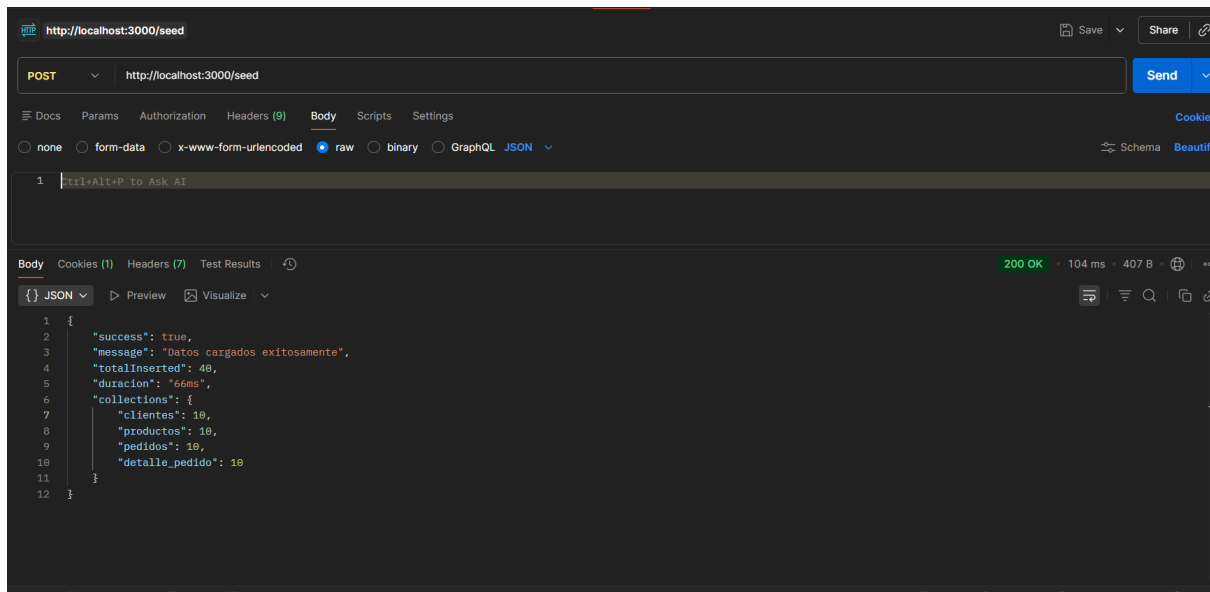
1. Método: **GET**
2. URL: <http://localhost:3000/>



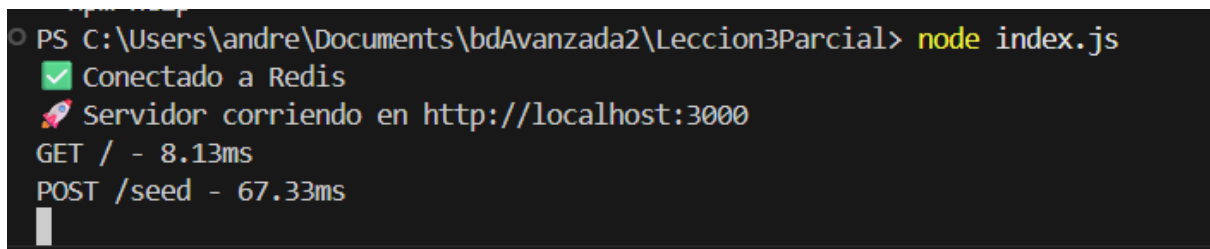
### PRUEBA 2: Cargar datos masivos (SEED)

carga todos los datos del JSON a Redis.

1. Método: **POST**
2. URL: <http://localhost:3000/seed>



## Terminal de vs:



## PRUEBA 3: Listar todos los clientes (GET)

1. Método: **GET**
2. URL: <http://localhost:3000/clientes>

GET http://localhost:3000/clientes Send

Docs Params Authorization Headers (8) Body Scripts Settings Cookies

Body Cookies (1) Headers (7) Test Results

200 OK 19 ms 1.71 KB

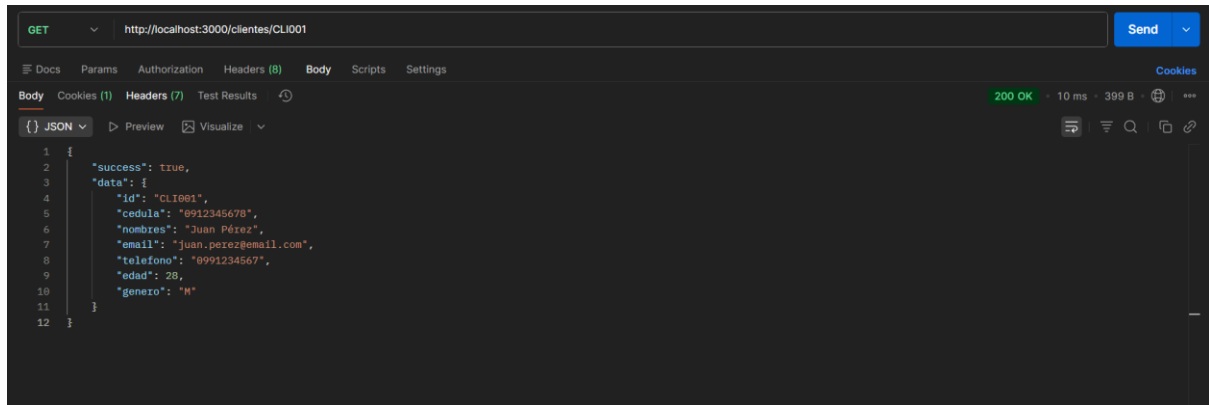
JSON Preview Visualize

```
1 {
2   "success": true,
3   "collection": "clientes",
4   "count": 10,
5   "data": [
6     {
7       "id": "CLI001",
8       "cedula": "0912345678",
9       "nombres": "Juan Pérez",
10      "email": "juan.perez@gmail.com",
11      "telefono": "0991234567",
12      "edad": 28,
13      "genero": "M"
14    },
15    {
16      "id": "CLI002",
17      "cedula": "0923456789",
18      "nombres": "Maria González",
19      "email": "maria.gonzalez@gmail.com",
20      "telefono": "0992345678",
21      "edad": 32,
22      "genero": "F"
23    },
24    {
25      "id": "CLI003",
26      "cedula": "0934567890",
27      "nombres": "Carlos Rodríguez",
28      "email": "carlos.rodriguez@gmail.com",
29      "telefono": "0993456789",
30      "edad": 25,
31      "genero": "M"
32    },
33    {
34      "id": "CLI004",
35      "cedula": "0945678901",
36      "nombres": "Ana Martínez",
37      "email": "ana.martinez@gmail.com",
38      "telefono": "0994567890",
39      "edad": 30,
40      "genero": "F"
41    },
42    {
43      "id": "CLI005",
44      "cedula": "0956789012",
45      "nombres": "Luis Fernández",
46      "email": "luis.fernandez@gmail.com",
47      "telefono": "0995678901",
48      "edad": 35,
49      "genero": "M"
50    },
51    {
52      "id": "CLI006",
53      "cedula": "0967890123",
54      "nombres": "Sofía López",
55      "email": "sofia.lopez@gmail.com",
56      "telefono": "0996789012",
57      "edad": 27,
58      "genero": "F"
59    },
60    {
61      "id": "CLI007",
62      "cedula": "0978901234",
63      "nombres": "Diego Torres",
64      "email": "diego.torres@gmail.com",
65      "telefono": "0997890123",
66      "edad": 29,
67      "genero": "M"
68    },
69    {
70      "id": "CLI008",
71      "cedula": "0989012345",
72      "nombres": "Valentina Ruiz",
73      "email": "valentina.ruiz@gmail.com",
74      "telefono": "0998901234",
75      "edad": 24,
76      "genero": "F"
77    },
78    {
79      "id": "CLI009",
80      "cedula": "0990123456",
81      "nombres": "Andrés Silva",
82      "email": "andres.silva@gmail.com",
83      "telefono": "0999012345",
84      "edad": 31,
85      "genero": "M"
86    },
87    {
88      "id": "CLI010",
89      "cedula": "0901234567",
90      "nombres": "Camila Herrera",
91      "email": "camila.herrera@gmail.com",
92      "telefono": "0990123456",
93      "edad": 26,
94      "genero": "F"
95    }
96  ]
97 }
```

Find and replace Console Terminal Runner Start Proxy Cookies Vault Trash

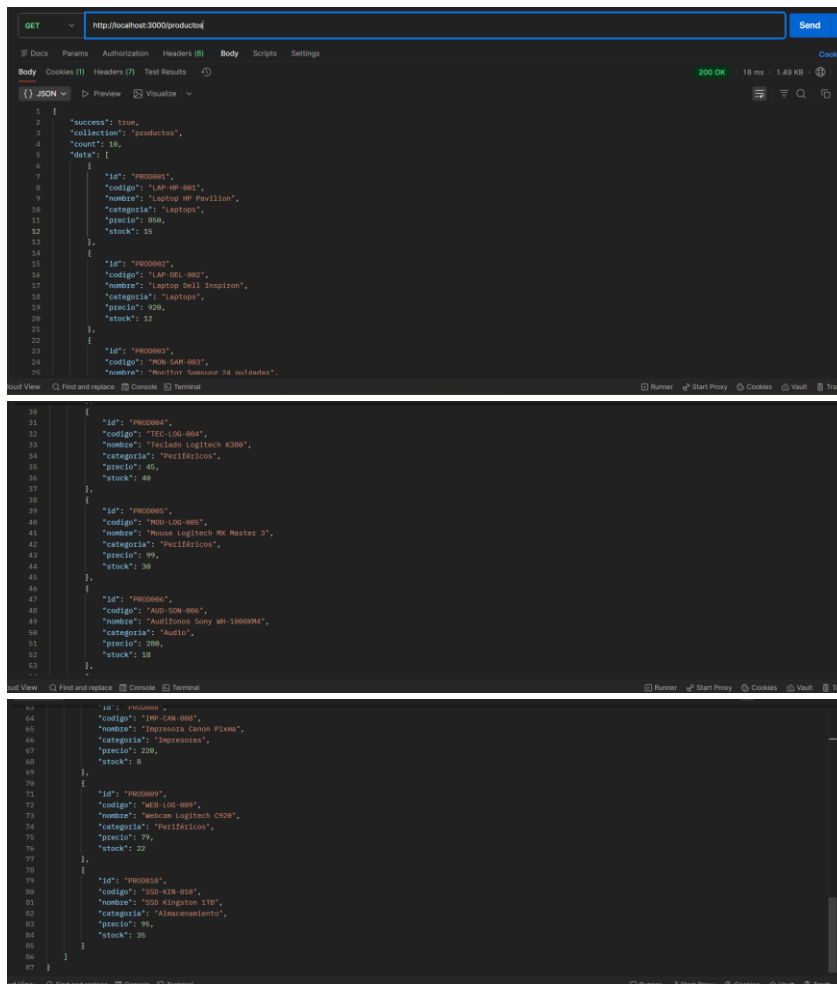
## PRUEBA 4: Obtener un cliente específico (GET)

1. Método: **GET**
2. URL: <http://localhost:3000/clientes/CLI001>



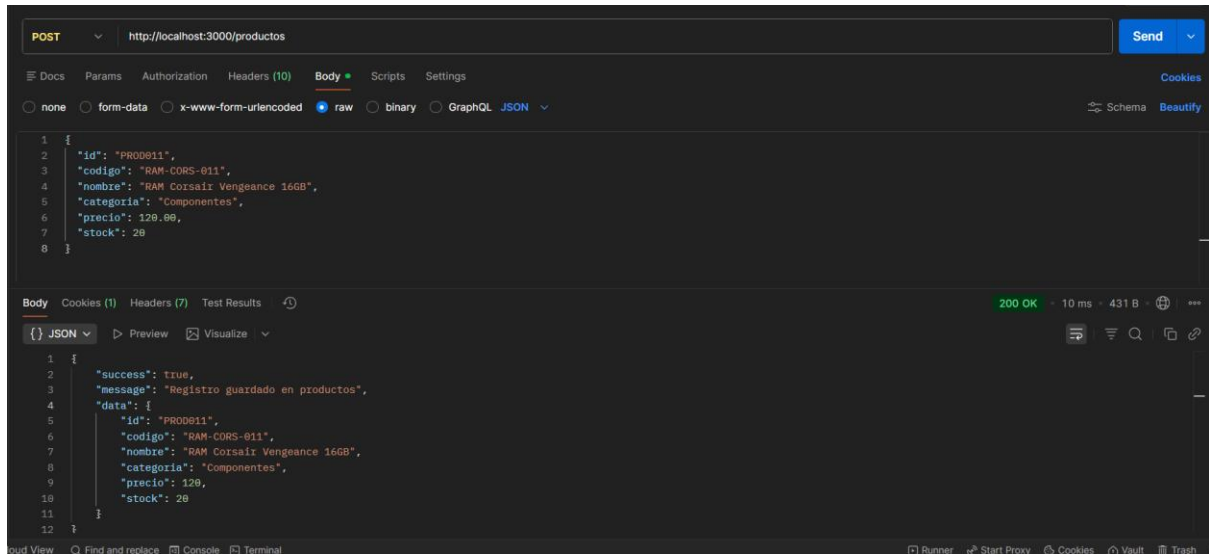
## PRUEBA 5: Listar todos los productos (GET)

1. Método: **GET**
2. URL: <http://localhost:3000/productos>



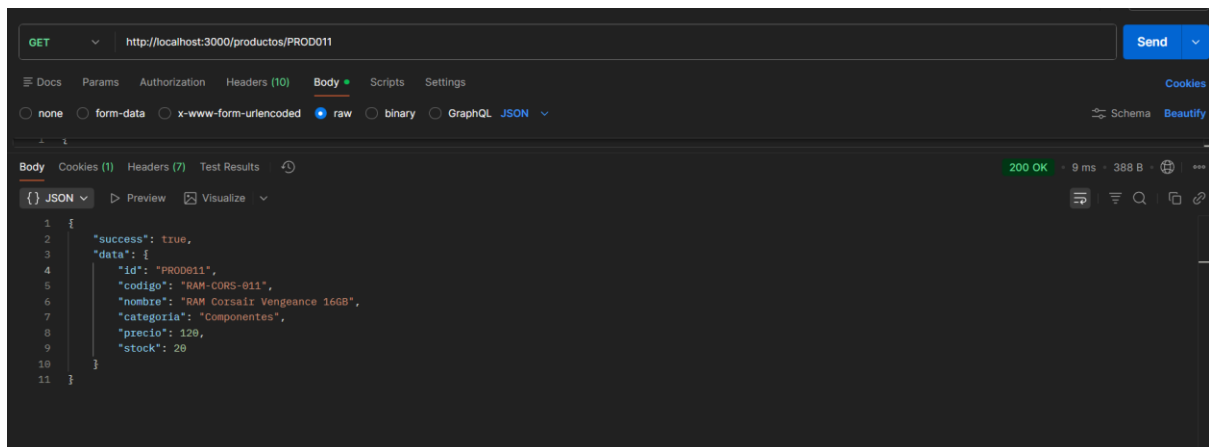
## PRUEBA 6: Crear un nuevo producto (POST)

1. Método: **POST**
2. URL: <http://localhost:3000/productos>



## PRUEBA 7: Verificar que el nuevo producto se guardó

1. Método: **GET**
2. URL: <http://localhost:3000/productos/PROD011>



## PRUEBA 8: Listar todos los pedidos

1. Método: **GET**
2. URL: <http://localhost:3000/pedidos>

GET http://localhost:3000/pedidos Send

Docs Params Authorization Headers (10) Body Scripts Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Schema Beautify

Body Cookies (1) Headers (7) Test Results 200 OK 20 ms 1.69 KB

{ } JSON Preview Visualize

```
1 {
2   "success": true,
3   "collection": "pedidos",
4   "count": 10,
5   "data": [
6     {
7       "id": "PED001",
8       "codigo": "ORD-2025-001",
9       "clienteId": "CLI001",
10      "fecha": "2025-01-15",
11      "subtotal": 850,
12      "iva": 192,
13      "total": 952,
14      "estado": "completado"
15    },
16    {
17      "id": "PED002",
18      "codigo": "ORD-2025-002",
19      "clienteId": "CLI002",
20      "fecha": "2025-01-16",
21      "subtotal": 225,
```

```
26 {
27   "id": "PED003",
28   "codigo": "ORD-2025-003",
29   "clienteId": "CLI003",
30   "fecha": "2025-01-17",
31   "subtotal": 920,
32   "iva": 110.4,
33   "total": 1030.4,
34   "estado": "completado"
35 },
36 {
37   "id": "PED004",
38   "codigo": "ORD-2025-004",
39   "clienteId": "CLI004",
40   "fecha": "2025-01-18",
41   "subtotal": 379,
42   "iva": 45.48,
43   "total": 424.48,
44   "estado": "en_proceso"
45 },
46 {
```

```
46 {
47   "id": "PED005",
48   "codigo": "ORD-2025-005",
49   "clienteId": "CLI005",
50   "fecha": "2025-01-19",
51   "subtotal": 650,
52   "iva": 78,
53   "total": 728,
54   "estado": "completado"
55 },
56 {
57   "id": "PED006",
58   "codigo": "ORD-2025-006",
59   "clienteId": "CLI006",
60   "fecha": "2025-01-20",
61   "subtotal": 144,
62   "iva": 17.28,
63   "total": 161.28,
64   "estado": "pendiente"
65 },
66 }
```

```
65     },
66     {
67       "id": "PED007",
68       "codigo": "ORD-2025-007",
69       "clienteId": "CLI007",
70       "fecha": "2025-01-21",
71       "subtotal": 280,
72       "iva": 33.6,
73       "total": 313.6,
74       "estado": "completado"
75     },
76     {
77       "id": "PED008",
78       "codigo": "ORD-2025-008",
79       "clienteId": "CLI008",
80       "fecha": "2025-01-22",
81       "subtotal": 220,
82       "iva": 26.4,
83       "total": 246.4,
84       "estado": "en_proceso"
85     }
86   ]
87 }
88
89 {
90   "id": "PED009",
91   "codigo": "ORD-2025-009",
92   "clienteId": "CLI009",
93   "fecha": "2025-01-23",
94   "subtotal": 174,
95   "iva": 20.88,
96   "total": 194.88,
97   "estado": "completado"
98 }
99
100 {
101   "id": "PED010",
102   "codigo": "ORD-2025-010",
103   "clienteId": "CLI010",
104   "fecha": "2025-01-24",
105   "subtotal": 95,
106   "iva": 11.4,
107   "total": 106.4,
108   "estado": "pendiente"
109 }
110 ]
111 }
```

## PRUEBA 9: Listar todos los detalles de pedido

1. Método: **GET**
2. URL: [http://localhost:3000/detalle\\_pedido](http://localhost:3000/detalle_pedido)

```
GET http://localhost:3000/detalle_pedido
200 OK 17 ms 1.7 KB
{
  "success": true,
  "collection": "detalle_pedido",
  "count": 10,
  "data": [
    {
      "id": "DET001",
      "codigo": "DET-001",
      "pedidoId": "PED001",
      "productoId": "PROD001",
      "cantidad": 1,
      "detalle": "Laptop HP Pavilion para trabajo",
      "precioUnit": 850
    },
    {
      "id": "DET002",
      "codigo": "DET-002",
      "pedidoId": "PED002",
      "productoId": "PROD003",
      "cantidad": 1,
      "detalle": "Monitor para oficina",
      "precioUnit": 45
    },
    {
      "id": "DET003",
      "codigo": "DET-003",
      "pedidoId": "PED002",
      "productoId": "PROD004",
      "cantidad": 1,
      "detalle": "Teclado inalámbrico",
      "precioUnit": 45
    },
    {
      "id": "DET004",
      "codigo": "DET-004",
      "pedidoId": "PED003",
      "productoId": "PROD002",
      "cantidad": 1,
      "detalle": "Laptop Dell para estudiante",
      "precioUnit": 920
    },
    {
      "id": "DET005",
      "codigo": "DET-005",
      "pedidoId": "PED004",
      "productoId": "PROD005",
      "cantidad": 1,
      "detalle": "Mouse inalámbrico",
      "precioUnit": 15
    }
  ]
}
```

```
40     "precioUnit": 920
41   },
42   {
43     "id": "DET005",
44     "codigo": "DET-005",
45     "pedidoId": "PED004",
46     "productoId": "PROD006",
47     "cantidad": 1,
48     "detalle": "Audífonos con cancelación de ruido",
49     "precioUnit": 280
50   },
51   {
52     "id": "DET006",
53     "codigo": "DET-006",
54     "pedidoId": "PED004",
55     "productoId": "PROD005",
56     "cantidad": 1,
57     "detalle": "Mouse ergonómico",
58     "precioUnit": 99
59   },
60   {
61     "id": "DET007",
62     "codigo": "DET-007",
63     "pedidoId": "PED005",
64     "productoId": "PROD007",
65     "cantidad": 1,
66     "detalle": "Tablet para diseño",
67     "precioUnit": 650
68   },
69   {
70     "id": "DET008",
71     "codigo": "DET-008",
72     "pedidoId": "PED006",
73     "productoId": "PROD004",
74     "cantidad": 2,
75     "detalle": "Teclados para equipo",
76     "precioUnit": 45
77   },
78   {
79     "id": "DET009",
80     "codigo": "DET-009",
81     "pedidoId": "PED006",
82     "productoId": "PROD009",
83     "cantidad": 2,
84     "detalle": "Webcams para reuniones",
85     "precioUnit": 79
86   },
87   {
88     "id": "DET010",
89     "codigo": "DET-010",
90     "pedidoId": "PED007",
91     "productoId": "PROD006",
92     "cantidad": 1,
93     "detalle": "Audífonos premium",
94     "precioUnit": 280
95   }
96 ]
97 }
```

## Captura de la Terminal:

```
PROBLEMAS  SALIDA  CONSOLA DE DEPURACIÓN  TERMINAL  PUERTOS

PS C:\Users\andre\Documents\bdAvanzada2\leccion3Parcial> node index.js
GET / - 8.13ms
POST /seed - 67.33ms
GET /clientes - 10.12ms
GET /clientes/CLI001 - 2.10ms
GET /productos - 8.73ms
POST /productos - 0.72ms
POST /productos - 2.29ms
GET /productos/PROD011 - 1.32ms
GET /pedidos - 8.84ms
GET /detalle_pedido - 8.60ms
GET /detalle_pedido - 8.37ms

Lín. 370, col. 2  Espacios: 4  UTF-8  CRLF  {} JSON
```

## Evidencias Ubuntu

### Ver todas las claves almacenadas:

```
andre@BEYKER:~$ redis-cli
127.0.0.1:6379> ping
PONG
127.0.0.1:6379> KEYS *
 1) "clientes:CLI005"
 2) "pedidos:PED010"
 3) "productos:PROD011"
 4) "estudiante:"
 5) "estudiante:1724966849"
 6) "pedidos:PED003"
 7) "detalle_pedido:DET004"
 8) "clientes:CLI003"
 9) "estudiante:1850598010"
10) "productos:PROD003"
11) "detalle_pedido:DET005"
12) "estudiante:1725600819"
13) "clientes:CLI001"
14) "pedidos:PED001"
15) "detalle_pedido:DET002"
16) "clientes:index"
17) "productos:PROD004"
18) "detalle_pedido:index"
19) "empleado:1"
20) "estudiante:1754459160"
21) "detalle_pedido:DET001"
22) "clientes:CLI002"
23) "estudiante:1722651567"
24) "clientes:CLI007"
25) "clientes:CLI009"
26) "estudiante:1723167514"
27) "key"
28) "detalle_pedido:DET003"
29) "test"
30) "productos:PROD005"
31) "estudiante:1850677764"
32) "clientes:CLI008"
33) "user:1002"
34) "productos:PROD009"
35) "pedidos:PED007"
36) "pedidos:PED006"
37) "estudiante:1727641407"
```

### Ver un registro específico (JSON)

JSON almacenado del cliente

```
127.0.0.1:6379> GET clientes:CLI001
{"id":"CLI001","cedula":"0912345678","nombres":"Juan P\xe9rez","email":"juan.perez@email.com","telefono":"0991234567","edad":28,"genero":"M"}
```

### Ver el contenido de un índice (Set)

todos los IDs de clientes

```
127.0.0.1:6379> SMEMBERS clientes:index  
1) "CLI001"  
2) "CLI002"  
3) "CLI003"  
4) "CLI004"  
5) "CLI005"  
6) "CLI006"  
7) "CLI007"  
8) "CLI008"  
9) "CLI009"  
10) "CLI010"
```

### Contar cuántas claves hay

total de claves

```
127.0.0.1:6379> DBSIZE  
(integer) 72
```