**📌 Ejercicio 1 – Libros ↔ Autores**

**Enunciado**: Unir libros con autores y mostrar título, nombre y nacionalidad.

db.libros.aggregate([

{

$lookup: {

from: "autores",

localField: "autores",

foreignField: "nombre",

as: "infoAutores"

}

},

{ $unwind: "$infoAutores" },

{

$project: {

\_id: 0,

titulo: 1,

autor: "$infoAutores.nombre",

nacionalidad: "$infoAutores.nacionalidad"

}

}

])

**Resultado esperado (ejemplo):**

{ "titulo": "The Hobbit", "autor": "J.R.R. Tolkien", "nacionalidad": "Reino Unido" }

{ "titulo": "Harry Potter y la Piedra Filosofal", "autor": "J.K. Rowling", "nacionalidad": "Reino Unido" }

**📌 Ejercicio 2 – Películas ↔ Empleados (críticos)**

**Enunciado**: Unir películas con empleados cuando criticas.autor = nombre.

db.peliculas.aggregate([

{ $unwind: "$criticas" },

{

$lookup: {

from: "empleados",

localField: "criticas.autor",

foreignField: "nombre",

as: "empleadoCritico"

}

},

{ $unwind: { path: "$empleadoCritico", preserveNullAndEmptyArrays: true } },

{

$project: {

\_id: 0,

pelicula: "$titulo",

autorCritica: "$criticas.autor",

comentario: "$criticas.comentario",

legajo: "$empleadoCritico.legajo"

}

}

])

**Resultado esperado (ejemplo):**

{ "pelicula": "Titanic", "autorCritica": "Lucía Gómez", "comentario": "Emotiva", "legajo": 10002 }

{ "pelicula": "El Padrino", "autorCritica": "Carlos", "comentario": "Una obra maestra", "legajo": null }

*(Si el autor no está en empleados, legajo queda null)*

**📌 Ejercicio 3 – Empleados ↔ Libros (ciudad)**

**Enunciado**: Relacionar empleados con libros de la misma ciudad.

db.empleados.aggregate([

{

$lookup: {

from: "libros",

localField: "ciudadTrabajo",

foreignField: "editorial.ciudad",

as: "librosCiudad"

}

},

{

$project: {

\_id: 0,

nombre: 1,

ciudadTrabajo: 1,

cantidadLibros: { $size: "$librosCiudad" }

}

}

])

**Resultado esperado (ejemplo):**

{ "nombre": "Juan Perez", "ciudadTrabajo": "Bogotá", "cantidadLibros": 1 }

{ "nombre": "Carla Ruiz", "ciudadTrabajo": "Mar del Plata", "cantidadLibros": 1 }

**📌 Ejercicio 4 – Lookup encadenado**

**Enunciado**: Empleado → crítica → película → director.

db.empleados.aggregate([

{

$lookup: {

from: "peliculas",

localField: "nombre",

foreignField: "criticas.autor",

as: "peliculasCriticadas"

}

},

{ $unwind: "$peliculasCriticadas" },

{

$project: {

\_id: 0,

empleado: "$nombre",

critica: "$peliculasCriticadas.criticas",

director: "$peliculasCriticadas.director"

}

}

])

**Resultado esperado (ejemplo):**

{ "empleado": "Lucía Gómez", "pelicula": "Titanic", "director": "James Cameron", "critica": [...] }

**📌 Ejercicio 5 – Desafío Final**

**Enunciado**: Top 3 empleados más críticos.

db.peliculas.aggregate([

{ $unwind: "$criticas" },

{

$lookup: {

from: "empleados",

localField: "criticas.autor",

foreignField: "nombre",

as: "empleadoCritico"

}

},

{ $unwind: { path: "$empleadoCritico", preserveNullAndEmptyArrays: true } },

{

$group: {

\_id: "$empleadoCritico.nombre",

totalCriticas: { $sum: 1 }

}

},

{ $sort: { totalCriticas: -1 } },

{ $limit: 3 }

])

**Resultado esperado (ejemplo):**

{ "\_id": "Lucía Gómez", "totalCriticas": 2 }

{ "\_id": "Juan Perez", "totalCriticas": 1 }

{ "\_id": null, "totalCriticas": 4 } // críticas hechas por autores que no son empleados

**📌 5 Ejercicios Simples de $lookup**

**🔹 Ejercicio 1 – Libros y Editoriales**

Relacionar libros con su editorial (ya está embebida, pero usalo como práctica).  
👉 Mostrar título del libro y ciudad de la editorial.

db.libros.aggregate([

{

$lookup: {

from: "libros", // misma colección (self-join de práctica)

localField: "editorial.ciudad",

foreignField: "editorial.ciudad",

as: "editorialRelacionada"

}

},

{ $project: { \_id: 0, titulo: 1, ciudad: "$editorial.ciudad" } }

])

**🔹 Ejercicio 2 – Películas y Premios**

Unir películas con otras películas que tengan el mismo valor en premios.oscar.  
👉 Mostrar título y si ganó o no Oscar.

db.peliculas.aggregate([

{

$lookup: {

from: "peliculas",

localField: "premios.oscar",

foreignField: "premios.oscar",

as: "otrasConOscar"

}

},

{ $project: { \_id: 0, titulo: 1, oscar: "$premios.oscar" } }

])

**🔹 Ejercicio 3 – Empleados y Tareas**

Unir empleados con otros empleados que compartan al menos una tarea.  
👉 Mostrar nombre del empleado y cantidad de compañeros con misma tarea.

db.empleados.aggregate([

{

$lookup: {

from: "empleados",

localField: "tareas",

foreignField: "tareas",

as: "companeros"

}

},

{ $project: { \_id: 0, nombre: 1, cantidadCompaneros: { $size: "$companeros" } } }

])

**🔹 Ejercicio 4 – Libros y Años de Publicación**

Unir libros con otros libros publicados en el mismo año.  
👉 Mostrar título y cuántos libros salieron ese año.

db.libros.aggregate([

{

$lookup: {

from: "libros",

localField: "anio",

foreignField: "anio",

as: "mismoAnio"

}

},

{ $project: { \_id: 0, titulo: 1, anio: 1, cantidad: { $size: "$mismoAnio" } } }

])

**🔹 Ejercicio 5 – Películas y Directores**

Unir películas con otras películas del mismo director.  
👉 Mostrar título y cantidad de películas que hizo ese director.

db.peliculas.aggregate([

{

$lookup: {

from: "peliculas",

localField: "director",

foreignField: "director",

as: "otrasDelDirector"

}

},

{ $project: { \_id: 0, titulo: 1, director: 1, total: { $size: "$otrasDelDirector" } } }

])

=========================================finales===============================

**📌 6 Ejercicios Combinados**

**🔹 Ejercicio 1 – Empleados y Libros en la misma ciudad**

Unir empleados con libros publicados en su ciudadTrabajo.  
👉 Mostrar empleado, ciudad y cantidad de libros encontrados.

db.empleados.aggregate([

{

$lookup: {

from: "libros",

localField: "ciudadTrabajo",

foreignField: "editorial.ciudad",

as: "librosCiudad"

}

},

{

$project: {

\_id: 0,

empleado: "$nombre",

ciudad: "$ciudadTrabajo",

cantidadLibros: { $size: "$librosCiudad" }

}

}

])

**🔹 Ejercicio 2 – Películas y Empleados críticos**

Unir películas con empleados que escribieron críticas.  
👉 Mostrar película, autor de crítica y si ese autor está en empleados.

db.peliculas.aggregate([

{ $unwind: "$criticas" },

{

$lookup: {

from: "empleados",

localField: "criticas.autor",

foreignField: "nombre",

as: "empleadoCritico"

}

},

{

$project: {

\_id: 0,

pelicula: "$titulo",

autor: "$criticas.autor",

comentario: "$criticas.comentario",

esEmpleado: { $gt: [{ $size: "$empleadoCritico" }, 0] }

}

}

])

**🔹 Ejercicio 3 – Top 3 directores más criticados por empleados**

1. Desarmar críticas.
2. Relacionar con empleados (si el autor es empleado).
3. Agrupar por director y contar.
4. Ordenar y limitar a 3.

db.peliculas.aggregate([

{ $unwind: "$criticas" },

{

$lookup: {

from: "empleados",

localField: "criticas.autor",

foreignField: "nombre",

as: "empleado"

}

},

{ $match: { "empleado": { $ne: [] } } },

{

$group: {

\_id: "$director",

totalCriticas: { $sum: 1 }

}

},

{ $sort: { totalCriticas: -1 } },

{ $limit: 3 }

])

**🔹 Ejercicio 4 – Libros por década**

1. Agrupar libros por década (anio).
2. Mostrar cantidad de libros por década.

db.libros.aggregate([

{

$project: {

titulo: 1,

decada: { $subtract: ["$anio", { $mod: ["$anio", 10] }] }

}

},

{

$group: {

\_id: "$decada",

cantidadLibros: { $sum: 1 }

}

},

{ $sort: { \_id: 1 } }

])

**🔹 Ejercicio 5 – Regex en ciudades (con $lookup)**

Unir empleados con libros cuando ciudadTrabajo **contiene "mar"** (ej: "Mar del Plata").  
👉 Mostrar empleado y títulos de libros de esa ciudad.

db.empleados.aggregate([

{

$lookup: {

from: "libros",

let: { ciudad: "$ciudadTrabajo" },

pipeline: [

{ $match: { "editorial.ciudad": { $regex: /mar/i } } }

],

as: "librosRelacionados"

}

},

{

$project: {

\_id: 0,

empleado: "$nombre",

ciudad: "$ciudadTrabajo",

librosRelacionados: "$librosRelacionados.titulo"

}

}

])

**🔹 Ejercicio 6 – Regex en críticas de películas**

Unir películas con empleados que hicieron críticas que **contienen la palabra “obra”**.  
👉 Mostrar empleado, película y el comentario.

db.peliculas.aggregate([

{ $unwind: "$criticas" },

{ $match: { "criticas.comentario": { $regex: /obra/i } } },

{

$lookup: {

from: "empleados",

localField: "criticas.autor",

foreignField: "nombre",

as: "empleado"

}

},

{

$project: {

\_id: 0,

pelicula: "$titulo",

comentario: "$criticas.comentario",

autor: "$criticas.autor",

empleado: "$empleado.nombre"

}

}

])