**1. Find all job postings for a specific company by name:**

db.companies.aggregate([

{ $match: { name: "IBM" } },

{ $lookup: {

from: "job\_postings",

localField: "company\_id",

foreignField: "company\_id",

as: "job\_postings"

}}

])

A screenshot of a computer

Description automatically generated

**2. Get a list of companies and the number of job postings they have:**

db.companies.aggregate([

{ $lookup: {

from: "job\_postings",

localField: "company\_id",

foreignField: "company\_id",

as: "job\_postings"

}},

{ $project: {

name: 1,

number\_of\_job\_postings: { $size: "$job\_postings" }

}}

])

A screenshot of a computer

Description automatically generated

**3. Find job postings with a minimum salary above a certain threshold:**

db.job\_postings.aggregate([

{ $match: { min\_salary: { $gt: 50000 } } },

{ $lookup: {

from: "companies",

localField: "company\_id",

foreignField: "company\_id",

as: "company"

}},

{ $unwind: "$company" },

{ $project: {

job\_title: 1,

min\_salary: 1,

company\_name: "$company.name"

}}

])

A screenshot of a computer

Description automatically generated

**4. Find job postings in a specific city:**

db.companies.aggregate([

{ $match: { city: "Austin" } },

{ $lookup: {

from: "job\_postings",

localField: "company\_id",

foreignField: "company\_id",

as: "job\_postings"

}},

{ $unwind: "$job\_postings" },

{ $project: {

company\_name: "$name",

job\_title: "$job\_postings.job\_title",

salary: "$job\_postings.salary"

}}

])

A blue rectangular object with white border

Description automatically generated

**5.Find companies with more than a certain number of job postings:**

db.companies.aggregate([

{ $lookup: {

from: "job\_postings",

localField: "company\_id",

foreignField: "company\_id",

as: "job\_postings"

}},

{ $project: {

company\_name: "$name",

job\_posting\_count: { $size: "$job\_postings" }

}},

{ $match: { job\_posting\_count: { $gt: 10 } } }

])

A screenshot of a computer

Description automatically generated

**6. Aggregate job postings by industry and count them:**

db.companies.aggregate([

{ $lookup: {

from: "job\_postings",

localField: "company\_id",

foreignField: "company\_id",

as: "job\_postings"

}},

{ $unwind: "$job\_postings" },

{ $group: {

\_id: "$industry",

total\_job\_postings: { $sum: 1 }

}},

{ $sort: { total\_job\_postings: -1 } }

])

A screenshot of a computer

Description automatically generated

**7. List all companies with their highest and lowest salary job postings:**

db.companies.aggregate([

{ $lookup: {

from: "job\_postings",

localField: "company\_id",

foreignField: "company\_id",

as: "job\_postings"

}},

{ $project: {

company\_name: "$name",

highest\_salary\_posting: { $max: "$job\_postings.max\_salary" },

lowest\_salary\_posting: { $min: "$job\_postings.min\_salary" }

}}

])

A screenshot of a computer

Description automatically generated

**8. List companies and their job postings with a focus on specific job types (e.g., Full-time):**

db.companies.aggregate([

{ $lookup: {

from: "job\_postings",

localField: "company\_id",

foreignField: "company\_id",

as: "job\_postings"

}},

{ $unwind: "$job\_postings" },

{ $match: { "job\_postings.formatted\_work\_type": "Full-time" } },

{ $project: {

company\_name: "$name",

job\_title: "$job\_postings.job\_title",

work\_type: "$job\_postings.formatted\_work\_type"

}}

])

A computer screen shot of a computer screen

Description automatically generated

**9. Aggregate companies by state and count their job postings:**

db.companies.aggregate([

{ $lookup: {

from: "job\_postings",

localField: "company\_id",

foreignField: "company\_id",

as: "job\_postings"

}},

{ $unwind: "$job\_postings" },

{ $group: {

\_id: "$state",

total\_job\_postings: { $sum: 1 }

}},

{ $sort: { total\_job\_postings: -1 } }

])

A blue square with white text

Description automatically generated

**10. Aggregate job postings by job type (e.g., full-time, part-time) across all companies:**

db.job\_postings.aggregate([

{ $group: {

\_id: "$formatted\_work\_type",

total\_postings: { $sum: 1 }

}},

{ $sort: { total\_postings: -1 } }

])

A blue rectangle with white dots

Description automatically generated

SQL queries

1. List all job postings with their associated company name and industry.

db.job\_postings.aggregate([

{

$lookup: {

from: "companies",

localField: "company\_id",

foreignField: "company\_id",

as: "company\_details"

}

},

{

$unwind: "$company\_details"

},

{

$project: {

job\_id: 1,

job\_title: "$title",

company\_name: "$company\_details.name",

industry: "$company\_details.industry"

}

}

])

A blue square with white text

Description automatically generated

2. Find all companies that offer remote work positions and are in the 'Technology' industry.

db.job\_postings.aggregate([

{

$match: {

remote\_allowed: 1 // Assuming remote\_allowed field indicates remote jobs (1 for remote, 0 for non-remote)

}

},

{

$lookup: {

from: "companies",

localField: "company\_id",

foreignField: "company\_id",

as: "company\_details"

}

},

{

$unwind: "$company\_details"

},

{

$match: {

"company\_details.industry": /Technology/ // Assuming the industry is a string containing the word "Technology"

}

},

{

$group: {

\_id: "$company\_details.company\_id",

company\_name: { $first: "$company\_details.name" },

industry: { $first: "$company\_details.industry" }

}

}

])

A computer screen shot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

3. Find the average company size for each industry.

db.companies.aggregate([

{

$unwind: "$industry"

},

{

$group: {

\_id: "$industry",

averageSize: { $avg: "$employee\_count" }

}

}

])

A screenshot of a computer

Description automatically generated