1. **Calculate the average salary for jobs in each country and work type combination.**

```
db.LinkedinJobAnalysis.aggregate([
  {
    $group: {
      _id: { country: "$company_details.country", work_type: "$formatted_work_type" },
      avg_salary: { $avg: "$max_salary" }
    }
  },
  {
    $project: {
      country: "$_id.country",
      work_type: "$_id.work_type",
      avg_salary: 1,
      _id: 0
    }
  }
]);
```

Explanation : This query performs an aggregation operation on the `LinkedinJobAnalysis` collection. It groups the data by `country` and `work_type`, and calculates the average `max_salary` for each group. The result is a list of countries, work types, and their corresponding average salaries.

2. **Identify companies with the highest average number of job views for their postings.**

```
db.LinkedinJobAnalysis.aggregate([
  {
    $group: {
      _id: "$company_details.company_name",
      avg_views: { $avg: "$views" }
    }
  },
  {
    $sort: { avg_views: -1 }
  },
  {
```

```
        $limit: 5
    }
]);
```

Explanation : The query groups the data by company_name and calculates the average number of views for each company. The results are then sorted in descending order of avg_views. Finally, it limits the output to the top 5 companies with the highest average views.

3. **Find job titles and locations with the highest median salary, ordered by median salary in descending order.**

```
db.LinkedinJobAnalysis.aggregate([
    {
        $sort: { "med_salary": -1 }
    },
    {
        $project: {
            job_title: 1,
            location: 1,
```

```
            med_salary: 1
        }
    },
    {
        $limit: 10
    }
]);
```

```
> db.LinkedinJobAnalysis.aggregate([
    {
        $sort: { "med_salary": -1 }
    },
    {
        $project: {
            job_title: 1,
            location: 1,
            med_salary: 1
        }
    },
    {
        $limit: 10
    }
]);
< {
    _id: ObjectId("655ac79fc851ecee6a4ca5d4"),
    job_title: 'Accounting Specialist'
  }
  {
    _id: ObjectId("655ac79fc851ecee6a4ca5d8"),
    job_title: 'Outside Sales Representative'
  }
  {
    _id: ObjectId("655ac79fc851ecee6a4ca587"),
    job_title: 'IT Operations ServiceNow Admin'
  }
  {
    _id: ObjectId("655ac79fc851ecee6a4ca5bc"),
    job_title: 'Assistant Manager'
  }
  {
    _id: ObjectId("655ac79fc851ecee6a4ca598"),
    job_title: 'Legal Assistant Paralegal'
  }
  {
    _id: ObjectId("655ac79ec851ecee6a4ca57a"),
    job_title: 'Office Associate'
  }
```

Explanation : The query sorts the data in descending order based on the `med_salary` field. The query then projects or selects the `job_title`, `location`, and `med_salary` fields from the sorted data. Finally, it limits the output to the top 10 records with the highest median salaries.

4. **List Job Titles with High Views-to-Applications Ratio (Retrieve job titles with a high ratio of views to applications, indicating high interest relative to the number of applications.)**

```
db.LinkedinJobAnalysis.aggregate([
  {
    $match: {
      applies: { $gt: 0 }
    }
  },
  {
    $project: {
      job_title: 1,
      views_to_applications_ratio: { $divide: ["$views", "$applies"] }
```

```
        }
    },
    {
        $sort: { views_to_applications_ratio: -1 }
    },
    {
        $limit: 10
    }
]);
```

```
> db.LinkedinJobAnalysis.aggregate([
    {
        $match: {
            applies: { $gt: 0 }
        }
    },
    {
        $project: {
            job_title: 1,
            views_to_applications_ratio: { $divide: ["$views", "$applies"] }
        }
    },
    {
        $sort: { views_to_applications_ratio: -1 }
    },
    {
        $limit: 10
    }
]);
< {
    _id: ObjectId("655ac7cdc851ecee6a4d1a15"),
    job_title: 'Executive Assistant',
    views_to_applications_ratio: 428.3333333333333
  }
  {
    _id: ObjectId("655ac7ccc851ecee6a4d1353"),
    job_title: 'Senior Director Business Operations',
    views_to_applications_ratio: 279.8333333333333
  }
  {
    _id: ObjectId("655ac7cec851ecee6a4d1d20"),
    job_title: 'Web Developer',
    views_to_applications_ratio: 230
  }
  {
    _id: ObjectId("655ac7cdc851ecee6a4d1bd3"),
    job_title: 'Human Resources Assistant',
    views_to_applications_ratio: 213
  }
```

Explanation : This query first filters the data to include only those records where `applies` is greater than 0. Then, it calculates the ratio of `views` to `applies` for each job and includes the `job_title` in the output. The results are sorted in descending order based on the `views_to_applications_ratio`. Finally, it limits the output to the top 10 records with the highest views-to-applications ratio.

5. **Identify Job Titles with the Highest Increase in Applications (Find job titles with the highest percentage increase in the number of applications compared to the previous data collection.)**

```
db.LinkedinJobAnalysis.aggregate([
  {
    $sort: { "applies": -1 }
  },
  {
    $group: {
      _id: "$job_title",
      previous_applies: { $first: "$applies" },
      current_applies: { $first: "$applies" }
```

```
        }
    },
    {
        $project: {
            job_title: "$_id",
            percentage_increase: { $multiply: [{ $divide: [{ $subtract: ["$current_applies", "$previous_applies"] }, "$previous_applies"] }, 100] }
        }
    },
    {
        $sort: { percentage_increase: -1 }
    },
    {
        $limit: 10
    }
]);
```

```
> db.LinkedinJobAnalysis.aggregate([
    {
        $sort: { "applies": -1 }
    },
    {
        $group: {
            _id: "$job_title",
            previous_applies: { $first: "$applies" },
            current_applies: { $first: "$applies" }
        }
    },
    {
        $project: {
            job_title: "$_id",
            percentage_increase: { $multiply: [{ $divide: [{ $subtract: ["$current_applies", "$previous_applies"] }, "$previous_applies"] }, 100] }
        }
    },
    {
        $sort: { percentage_increase: -1 }
    },
    {
        $limit: 10
    }
]);
< {
  _id: 'Estimator / Business Development ',
  job_title: 'Estimator / Business Development ',
  percentage_increase: 0
}
{
  _id: 'Database Engineer',
  job_title: 'Database Engineer',
  percentage_increase: 0
}
{
  _id: 'Principle Specialist, Security',
  job_title: 'Principle Specialist, Security',
  percentage_increase: 0
}
```

Explanation : The query sorts the documents based on the number of job applications in descending order. Then, it groups the documents by job title and retains the first document's number of applies as both the previous and current applies for each group. After that, it calculates the percentage increase in job applications for each job title. The documents are then sorted in descending order based on this calculated percentage increase. Finally, the output is limited to the top 10 documents with the highest percentage increase in job applications.