

# OPTIMIZOVANI UPITI MONGO

## UPIT 1

---

```
1
2  const YEAR = 2020;
3  const STATE = "TX";
4
5  db.OnSiteReleaseAggregate.aggregate([
6
7    { $match: { year: YEAR, "facility.State": STATE } },
8
9
10   { $sort: { OnSiteReleaseTotalForAllChemicals: -1 } },
11
12
13   { $limit: 10 },
14
15
16   { $project: {
17     _id: 0,
18     FacilityId: "$facility.FacilityId",
19     FacilityName: "$facility.FacilityName",
20     City: "$facility.City",
21     State: "$facility.State",
22     IndustrySector: "$facility.IndustrySector",
23     CompanyName: "$facility.ParentCompany.CompanyName",
24     TotalRelease: "$OnSiteReleaseTotalForAllChemicals"
25   }}
26 ]);
27
28
29
30
31
32
33
34
35
36
37
38
39
40
```

```
1
2
3  const YEAR = 2024;
4
5  db.OnSiteReleaseAggregate.aggregate([
6
7    { $match: { year: YEAR } },
8
9
10   { $unwind: "$allReleases" },
11   { $unwind: "$allReleases.chemical.releasesForChemical" },
12
13
14   { $project: {
15     FacilityId: "$facility.FacilityId",
16     ParentCompanyId: "$facility.ParentCompany.CompanyId",
17     ParentCompanyName: "$facility.ParentCompany.CompanyName",
18     Air: "$allReleases.chemical.releasesForChemical.Air",
19     Water: "$allReleases.chemical.releasesForChemical.Water",
20     Land: { $add: ["$allReleases.chemical.releasesForChemical.Landfills",
21                   "$allReleases.chemical.releasesForChemical.Underground"] }
22   }},
23
24
25   { $group: {
26     _id: "$FacilityId",
27     ParentCompanyId: { $first: "$ParentCompanyId" },
28     ParentCompanyName: { $first: "$ParentCompanyName" },
29     TotalAir: { $sum: "$Air" },
30     TotalWater: { $sum: "$Water" },
31     TotalLand: { $sum: "$Land" }
32   }},
33
34
35   { $group: {
36     _id: "$ParentCompanyId",
37     CompanyName: { $first: "$ParentCompanyName" },
38     TotalAir: { $sum: "$TotalAir" },
39     TotalWater: { $sum: "$TotalWater" },
40     TotalLand: { $sum: "$TotalLand" }
41   }},
42
43   // sumiranje ukupno za sve kompanije
44   { $group: {
45     _id: null,
46     companies: { $push: "$$ROOT" },
```

```
47     TotalAirAll: { $sum: "$TotalAir" },
48     TotalWaterAll: { $sum: "$TotalWater" },
49     TotalLandAll: { $sum: "$TotalLand" }
50   }},
51
52
53   { $project: {
54     TotalAirAll: 1,
55     TotalWaterAll: 1,
56     TotalLandAll: 1,
57     AirTop3: {
58       $map: {
59         input: { $slice: [ { $sortBy: { TotalAir: -1 } } ], 3 ] },
60         as: "c",
61         in: {
62           CompanyName: "$$c.CompanyName",
63           TotalAir: "$$c.TotalAir",
64           Percent: { $multiply: [ { $divide: ["$$c.TotalAir", "$TotalAirAll"] }, 100 ] }
65         }
66       }
67     },
68     WaterTop3: {
69       $map: {
70         input: { $slice: [ { $sortBy: { TotalWater: -1 } } ], 3 ] },
71         as: "c",
72         in: {
73           CompanyName: "$$c.CompanyName",
74           TotalWater: "$$c.TotalWater",
75           Percent: { $multiply: [ { $divide: ["$$c.TotalWater", "$TotalWaterAll"] }, 100 ] }
76         }
77       }
78     },
79     LandTop3: {
80       $map: {
81         input: { $slice: [ { $sortBy: { TotalLand: -1 } } ], 3 ] },
82         as: "c",
83         in: {
84           CompanyName: "$$c.CompanyName",
85           TotalLand: "$$c.TotalLand",
86           Percent: { $multiply: [ { $divide: ["$$c.TotalLand", "$TotalLandAll"] }, 100 ] }
87         }
88       }
89     }
90   }}
91 ]);
92
93
94
```

```
1
2
3  const YEAR = 2020;
4
5  db.OffSiteTransferAggregate.aggregate([
6
7    { $match: { year: YEAR } },
8
9
10   { $unwind: "$allTransfers" },
11   { $unwind: "$allTransfers.chemical.transfersForChemical" },
12
13
14   {
15     $project: {
16       IndustrySector: "$facility.IndustrySector",
17       Recycle: "$allTransfers.chemical.transfersForChemical.OffSiteRecycledTotal",
18       Energy: "$allTransfers.chemical.transfersForChemical.OffSiteEnergyRecoveryTotal",
19       Treatment: { $add: ["$allTransfers.chemical.transfersForChemical.OffSiteTreatedTotal", "$allTransfers.chemical.transfersForChemical.POTWTotalTransfers"] },
20       Disposal: "$allTransfers.chemical.transfersForChemical.OffSiteReleaseTotal"
21     }
22   },
23
24
25   {
26     $group: {
27       _id: "$IndustrySector",
28       totalRecycle: { $sum: "$Recycle" },
29       totalEnergy: { $sum: "$Energy" },
30       totalTreatment: { $sum: "$Treatment" },
31       totalDisposal: { $sum: "$Disposal" }
32     }
33   },
34
35   // ukupne vrednosti i priprema top3
36   {
37     $group: {
38       _id: null,
39       sectors: { $push: "$$ROOT" },
40       totalRecycle: { $sum: "$totalRecycle" },
41       totalEnergy: { $sum: "$totalEnergy" },
42       totalTreatment: { $sum: "$totalTreatment" },
43       totalDisposal: { $sum: "$totalDisposal" }
44     }
45   },
46
```

```

47
48 {
49   $project: {
50     _id: 0,
51     TotalAmounts: {
52       Recycle: "$totalRecycle",
53       Energy: "$totalEnergy",
54       Treatment: "$totalTreatment",
55       Disposal: "$totalDisposal"
56     },
57     RecycleTop3: {
58       $map: {
59         input: { $slice: [{ $sortBy: { input: "$sectors", sortBy: { totalRecycle: -1 } } }, 3] },
60         as: "s",
61         in: {
62           Sector: "$$_id",
63           Amount: "$$_totalRecycle",
64           Percent: { $cond: [{ $eq: ["$totalRecycle", 0] }, 0, { $multiply: [{ $divide: ["$$s.totalRecycle", "$totalRecycle"] }, 100] }] }
65         }
66       }
67     },
68     EnergyTop3: {
69       $map: {
70         input: { $slice: [{ $sortBy: { input: "$sectors", sortBy: { totalEnergy: -1 } } }, 3] },
71         as: "s",
72         in: {
73           Sector: "$$_id",
74           Amount: "$$_totalEnergy",
75           Percent: { $cond: [{ $eq: ["$totalEnergy", 0] }, 0, { $multiply: [{ $divide: ["$$s.totalEnergy", "$totalEnergy"] }, 100] }] }
76         }
77       }
78     },
79     TreatmentTop3: {
80       $map: {
81         input: { $slice: [{ $sortBy: { input: "$sectors", sortBy: { totalTreatment: -1 } } }, 3] },
82         as: "s",
83         in: {
84           Sector: "$$_id",
85           Amount: "$$_totalTreatment",
86           Percent: { $cond: [{ $eq: ["$totalTreatment", 0] }, 0, { $multiply: [{ $divide: ["$$s.totalTreatment", "$totalTreatment"] }, 100] }] }
87         }
88       }
89     },
90     DisposalTop3: {
91       $map: {
92         input: { $slice: [{ $sortBy: { input: "$sectors", sortBy: { totalDisposal: -1 } } }, 3] },
93         as: "s",
94         in: {
95           Sector: "$$_id",

```

```

96         Amount: "$$s.totalDisposal",
97         Percent: { $cond: [{ $eq: ["$totalDisposal", 0] }, 0, { $multiply: [{ $divide: ["$$s.totalDisposal", "$totalDisposal"] }, 100] }] }
98     }
99 }
100 }
101 }
102 }
103 ]);
104
105
106
107
```

# UPIT 4

```

1
2  const YEAR = 2023;
3  const minLat = 30.0;
4  const maxLat = 50.0;
5  const minLng = -100.0;
6  const maxLng = -80.0;
7
8  db.OnSiteReleaseAggregate.aggregate([
9    { $match: {
10      year: YEAR,
11      "facility.Latitude": { $gte: minLat, $lte: maxLat },
12      "facility.Longitude": { $gte: minLng, $lte: maxLng }
13    }},
14    { $unwind: "$allReleases" },
15    { $unwind: "$allReleases.chemical.releasesForChemical" },
16    { $project: {
17      ChemicalId: "$allReleases.chemical.ChemicalId",
18      ChemicalName: "$allReleases.chemical.ChemicalName",
19      Air: { $ifNull: ["$allReleases.chemical.releasesForChemical.Air", 0] },
20      Water: { $ifNull: ["$allReleases.chemical.releasesForChemical.Water", 0] },
21      Land: { $add: [
22        { $ifNull: ["$allReleases.chemical.releasesForChemical.Landfills", 0] },
23        { $ifNull: ["$allReleases.chemical.releasesForChemical.Underground", 0] }
24      ]}
25    }},
26    { $group: {
27      _id: "$ChemicalId",
28      ChemicalName: { $first: "$ChemicalName" },
29      totalAir: { $sum: "$Air" },
30      totalWater: { $sum: "$Water" },
31      totalLand: { $sum: "$Land" }
```

```
32     }},
33
34     // ukupne sume
35     { $group: {
36         _id: null,
37         chemicals: { $push: "$$ROOT" },
38         totalAir: { $sum: "$totalAir" },
39         totalWater: { $sum: "$totalWater" },
40         totalLand: { $sum: "$totalLand" }
41     }},
42
43
44     { $project: {
45         _id: 0,
46         totalAir: 1,
47         totalWater: 1,
48         totalLand: 1,
49         TopAir: { $slice: [
50             { $sortByArray: { input: "$chemicals", sortBy: { totalAir: -1 } } }, 10
51         ]},
52         TopWater: { $slice: [
53             { $sortByArray: { input: "$chemicals", sortBy: { totalWater: -1 } } }, 10
54         ]},
55         TopLand: { $slice: [
56             { $sortByArray: { input: "$chemicals", sortBy: { totalLand: -1 } } }, 10
57         ]}
58     }}
59 ]);
60
```

# UPIT 5

```
1
2
3  const YEAR = 2024;
4  const MEDIUM = "Water"; // "Air", "Water", "Land"
5  const TOP_CHEMICALS = 10;
6  const TOP_DISEASES = 10;
7
8  db.OnSiteReleaseAggregate.aggregate([
9      { $match: { year: YEAR } },
10
11
12      { $unwind: "$allReleases" },
13      { $unwind: "$allReleases.chemical.releasesForChemical" },
14
```

```

15
16 { $project: {
17     ChemicalId: "$allReleases.chemical.ChemicalId",
18     ChemicalName: "$allReleases.chemical.ChemicalName",
19     Diseases: "$allReleases.chemical.diseases",
20     Amount: {
21         $cond: [
22             { $eq: [MEDIUM, "Land"] },
23             { $add: [
24                 "$allReleases.chemical.releasesForChemical.Landfills",
25                 "$allReleases.chemical.releasesForChemical.Underground"
26             ]},
27             `"$allReleases.chemical.releasesForChemical.${MEDIUM}`
28         ]
29     }
30 }},
31
32
33 { $group: {
34     _id: "$ChemicalId",
35     ChemicalName: { $first: "$ChemicalName" },
36     Diseases: { $first: "$Diseases" },
37     TotalRelease: { $sum: "$Amount" }
38 }},
39
40
41 { $sort: { TotalRelease: -1 } },
42 { $limit: TOP_CHEMICALS },
43
44
45 { $facet: {
46     TopChemicals: [
47         { $project: { _id: 0, ChemicalId: "$_id", ChemicalName: 1, TotalRelease: 1 } }
48     ],
49     TopDiseases: [
50         { $unwind: "$Diseases" },
51         { $lookup: {
52             from: "Diseases",
53             localField: "Diseases",
54             foreignField: "DiseaseId",
55             as: "diseaseInfo"
56         }},
57         { $unwind: "$diseaseInfo" },
58         { $group: { _id: "$diseaseInfo.DiseaseName", Count: { $sum: 1 } } },
59         { $sort: { Count: -1 } },
60         { $limit: TOP_DISEASES },
61         { $project: { _id: 0, DiseaseName: "$_id", Count: 1 } }
62     ]
63 }},

```



```
64
65     { $addField: { Year: YEAR, Medium: MEDIUM } }
66   ]);
67
```

## UPIT 6

---

```
1
2
3 db.OnSiteReleaseAggregate.aggregate([
4   {
5     $unwind: "$allReleases"
6   },
7   {
8     $match: {
9       $or: [
10        { "allReleases.chemical.PFAS": true },
11        { "allReleases.chemical.PBT": true },
12        { "allReleases.chemical.Carcinogen": true }
13      ]
14    }
15  },
16  {
17    $project: {
18      _id: 0,
19      FacilityId: "$facility.FacilityId",
20      City: "$facility.City",
21      State: "$facility.State",
22      IndustrySector: "$facility.IndustrySector",
23      ParentCompany: "$facility.ParentCompany.CompanyName",
24      amount_g: {
25        $ifNull: ["$allReleases.chemical.OnSiteReleaseTotalForChemical", 0]
26      }
27    }
28  },
29  {
30    $unionWith: {
31      coll: "OffSiteTransferAggregate",
32      pipeline: [
33        { $unwind: "$allTransfers" },
34        {
35          $match: {
36            $or: [
37              { "allTransfers.chemical.PFAS": true },
38              { "allTransfers.chemical.PBT": true },
39              { "allTransfers.chemical.Carcinogen": true }

```

```
40         ]
41     }
42 },
43 {
44     $project: {
45         _id: 0,
46         FacilityId: "$facility.FacilityId",
47         City: "$facility.City",
48         State: "$facility.State",
49         IndustrySector: "$facility.IndustrySector",
50         ParentCompany: "$facility.ParentCompany.CompanyName",
51         amount_g: {
52             $ifNull: ["$allTransfers.chemical.OffSiteReleaseTotalForChemical", 0]
53         }
54     }
55 }
56 ]
57 }
58 },
59 {
60     $group: {
61         _id: "$FacilityId",
62         totalReleasesGrams: { $sum: "$amount_g" },
63         City: { $first: "$City" },
64         State: { $first: "$State" },
65         IndustrySector: { $first: "$IndustrySector" },
66         ParentCompany: { $first: "$ParentCompany" }
67     }
68 },
69 {
70     $project: {
71         _id: 0,
72         FacilityId: "$_id",
73         City: 1,
74         State: 1,
75         IndustrySector: 1,
76         ParentCompany: 1,
77         totalReleasesGrams: { $round: ["$totalReleasesGrams", 3] }
78     }
79 },
80 { $sort: { totalReleasesGrams: -1 } },
81 { $limit: 10 }
82 ]));
83
84
85
```

```
1
2 // #7 Federal vs Non-Federal: Total Releases po sektoru + sažeci po grupi
3 // - koristi novu šemu: OnSiteReleaseAggregate / OffSiteTransferAggregate
4 // - broji jedinstvene FacilityId po sektoru i po federal grupi (nema dupliranja)
5
6 db.OnSiteReleaseAggregate.aggregate([
7   {
8     $project: {
9       _id: 0,
10      FacilityId: "$facility.FacilityId",
11      sector: { $ifNull: ["$facility.IndustrySector", "Unknown"] },
12      federal: { $toBool: { $ifNull: ["$facility.Federal", false] } },
13      releases: { $ifNull: ["$OnSiteReleaseTotalForAllChemicals", 0] }
14    }
15  },
16  {
17    $unionWith: {
18      coll: "OffSiteTransferAggregate",
19      pipeline: [
20        {
21          $project: {
22            _id: 0,
23            FacilityId: "$facility.FacilityId",
24            sector: { $ifNull: ["$facility.IndustrySector", "Unknown"] },
25            federal: { $toBool: { $ifNull: ["$facility.Federal", false] } },
26            releases: { $ifNull: ["$OffSiteReleaseTotalForAllChemicals", 0] }
27          }
28        }
29      ]
30    }
31  },
32  {
33    $group: {
34      _id: "$FacilityId",
35      totalReleases: { $sum: "$releases" },
36      federal: { $first: "$federal" },
37      sector: { $first: "$sector" }
38    }
39  },
40  {
41    $group: {
42      _id: { federal: "$federal", sector: "$sector" },
43      totalReleases: { $sum: "$totalReleases" },
44      facilityIds: { $addToSet: "$_id" }
45    }
46  },
```

```
47 {
48   $set: {
49     facilityCount: { $size: { $ifNull: ["$facilityIds", []] } },
50     avgPerFacility: {
51       $cond: [
52         { $gt: [{ $size: { $ifNull: ["$facilityIds", []] } }, 0] },
53         { $divide: ["$totalReleases", { $size: "$facilityIds" }] },
54         0
55       ]
56     }
57   }
58 },
59 {
60   $sort: { "_id.federal": -1, totalReleases: -1 }
61 },
62 {
63   $group: {
64     _id: "$_id.federal",
65     groupTotalReleases: { $sum: "$totalReleases" },
66     bySector: {
67       $push: {
68         sector: "$_id.sector",
69         totalReleases: "$totalReleases",
70         facilityIds: "$facilityIds",
71         facilityCount: "$facilityCount",
72         avgPerFacility: "$avgPerFacility"
73       }
74     },
75     dominantSector: {
76       $first: {
77         sector: "$_id.sector",
78         totalReleases: "$totalReleases"
79       }
80     }
81   }
82 },
83 {
84   $set: {
85     _allFacilityIds: {
86       $reduce: {
87         input: "$bySector",
88         initialValue: [],
89         in: { $setUnion: ["$$value", "$$this.facilityIds"] }
90       }
91     }
92   }
93 },
94 {
95   $set: {
```

```
96     groupFacilityCount: { $size: "$_allFacilityIds" },
97     groupAvgPerFacility: {
98       $cond: [
99         { $gt: [{ $size: "$_allFacilityIds" }, 0] },
100         { $divide: ["$groupTotalReleases", { $size: "$_allFacilityIds" }] } ],
101       0
102     ]
103   }
104 }
105 },
106 {
107   $project: {
108     _id: 0,
109     federalGroup: { $cond: ["$_id", "Federal", "Non-Federal"] },
110     totals: {
111       totalReleases: "$groupTotalReleases",
112       avgPerFacility: "$groupAvgPerFacility",
113       facilityCount: "$groupFacilityCount"
114     },
115     dominantSector: 1,
116     bySector: {
117       $map: {
118         input: "$bySector",
119         as: "s",
120         in: {
121           sector: "$$s.sector",
122           totalReleases: "$$s.totalReleases",
123           facilityCount: "$$s.facilityCount",
124           avgPerFacility: "$$s.avgPerFacility"
125         }
126       }
127     }
128   }
129 },
130 {
131   $sort: { federalGroup: 1 }
132 }
133 ]);
134
135
136
137
138
139
140
141
142
```

```
1
2
3 // Udeo reciklaže po hemikaliji – prilagođeno OffSiteTransferAggregate
4
5 const YEAR = null; // stavi null za sve godine
6 const TOP = 10;
7
8 db.OffSiteTransferAggregate.aggregate([
9
10   ...(YEAR ? [{ $match: { year: YEAR } }] : []),
11
12
13   { $unwind: "$allTransfers" },
14   { $set: { chem: "$allTransfers.chemical" } },
15
16   // sumirizuj kroz transfersForChemical[] (ako ih ima više)
17   { $addFields: {
18     _sum: {
19       $reduce: {
20         input: { $ifNull: ["$chem.transfersForChemical", []] },
21         initialValue: { rec: 0, en: 0, tr: 0, rel: 0, potw: 0 },
22         in: {
23           rec: { $add: ["$$value.rec", { $ifNull: ["$$this.OffSiteRecycledTotal", 0] }] },
24           en: { $add: ["$$value.en", { $ifNull: ["$$this.OffSiteEnergyRecoveryTotal", 0] }] },
25           tr: { $add: ["$$value.tr", { $ifNull: ["$$this.OffSiteTreatedTotal", 0] }] },
26           rel: { $add: ["$$value.rel", { $ifNull: ["$$this.OffSiteReleaseTotal", 0] }] },
27           potw: { $add: ["$$value.potw", { $ifNull: ["$$this.POTWTotalTransfers", 0] }] }
28         }
29       }
30     }
31   }},
32
33
34   { $project: {
35     ChemicalId: "$chem.ChemicalId",
36     ChemicalName: "$chem.ChemicalName",
37     FacilityId: "$facility.FacilityId",
38     City: "$facility.City",
39     IndustrySector: "$facility.IndustrySector",
40     recycled_g: "$_sum.rec",
41     totalWaste_g: { $add: ["$_sum.rec", "$_sum.en", "$_sum.tr", "$_sum.rel", "$_sum.potw"] }
42   }},
43   { $group: {
44     _id: { ChemicalId: "$ChemicalId", FacilityId: "$FacilityId" },
45     ChemicalName: { $first: "$ChemicalName" },
46     City: { $first: "$City" },
```

```
47     IndustrySector: { $first: "$IndustrySector" },
48     recycled_g:      { $sum: "$recycled_g" },
49     totalWaste_g:    { $sum: "$totalWaste_g" }
50   }},
51
52
53
54
55   { $facet: {
56     perChem: [
57       { $group: {
58         _id: "$_id.ChemicalId",
59         ChemicalName: { $first: "$ChemicalName" },
60         totalRecycled_g: { $sum: "$recycled_g" },
61         totalWaste_g: { $sum: "$totalWaste_g" }
62       }},
63       { $set: {
64         recycleSharePct: {
65           $cond: [
66             { $gt: ["$totalWaste_g", 0] },
67             { $multiply: [{ $divide: ["$totalRecycled_g", "$totalWaste_g"] }, 100] },
68             0
69           ]
70         }
71       }},
72       { $project: {
73         _id: 0,
74         ChemicalId: "$_id",
75         ChemicalName: 1,
76         totalWasteGrams: { $round: ["$totalWaste_g", 3] },
77         totalRecycledGrams: { $round: ["$totalRecycled_g", 3] },
78         recycleSharePct: { $round: ["$recycleSharePct", 2] }
79       }}
80     ],
81     topFac: [
82       { $sort: { "_id.ChemicalId": 1, recycled_g: -1 } },
83       { $group: {
84         _id: "$_id.ChemicalId",
85         topFacilityId: { $first: "$_id.FacilityId" },
86         topRecycled_g: { $first: "$recycled_g" },
87         City: { $first: "$City" },
88         IndustrySector: { $first: "$IndustrySector" }
89       }},
90       { $project: {
91         _id: 0,
92         ChemicalId: "$_id",
93         DominantFacility: {
94           FacilityId: "$topFacilityId",
95
```

```

96         City: "$City",
97         IndustrySectorApprox: "$IndustrySector",
98         RecycledGramsAtTopFacility: { $round: ["$stopRecycled_g", 3] }
99     }
100 }
101 ]
102 }},
103
104
105 { $project: {
106     merged: {
107         $map: {
108             input: "$perChem",
109             as: "pc",
110             in: {
111                 $mergeObjects: [
112                     "$$pc",
113                     {
114                         $ifNull: [
115                             {
116                                 $first: {
117                                     $filter: {
118                                         input: "$stopFac",
119                                         as: "tf",
120                                         cond: { $eq: ["$$tf.ChemicalId", "$$pc.ChemicalId"] }
121                                     }
122                                 }
123                             },
124                             {}
125                         ]
126                     }
127                 ]
128             }
129         }
130     }
131 }},
132
133 // 6) Top 10 po procentu reciklaže
134 { $unwind: "$merged" },
135 { $replaceRoot: { newRoot: "$merged" } } },
136
137 { $facet: {
138     top10: [
139         { $sort: { recycleSharePct: -1, totalWasteGrams: -1, ChemicalName: 1 } },
140         { $limit: TOP }
141     ]
142 }}
143 ]);

```



```
1
2
3
4  const CHEM_NAME = "Ammonia";
5  const YEAR_FROM = 2020;
6  const YEAR_TO   = 2024;
7
8
9  const onsite = [
10    { $match: { year: { $gte: YEAR_FROM, $lte: YEAR_TO } } },
11    { $unwind: "$allReleases" },
12    { $set: { chem: "$allReleases.chemical" } },
13    { $match: { "chem.ChemicalName": CHEM_NAME } },
14    { $project: {
15      _id: 0,
16      FacilityId: "$facility.FacilityId",
17      Year: "$year",
18      IndustrySector: "$facility.IndustrySector",
19      releases: { $ifNull: ["$chem.OnSiteReleaseTotalForChemical", 0] }
20    }},
21    { $group: {
22      _id: { FacilityId: "$FacilityId", Year: "$Year", Sector: "$IndustrySector" },
23      totalReleases: { $sum: "$releases" }
24    }}
25  ];
26
27
28  const offsite = [
29    { $match: { year: { $gte: YEAR_FROM, $lte: YEAR_TO } } },
30    { $unwind: "$allTransfers" },
31    { $set: { chem: "$allTransfers.chemical" } },
32    { $match: { "chem.ChemicalName": CHEM_NAME } },
33    { $project: {
34      _id: 0,
35      FacilityId: "$facility.FacilityId",
36      Year: "$year",
37      IndustrySector: "$facility.IndustrySector",
38      releases: { $ifNull: ["$chem.OffSiteReleaseTotalForChemical", 0] }
39    }},
40    { $group: {
41      _id: { FacilityId: "$FacilityId", Year: "$Year", Sector: "$IndustrySector" },
42      totalReleases: { $sum: "$releases" }
43    }}
44  ];
45
46
```

```
47 db.OnSiteReleaseAggregate.aggregate([
48   ... onsite,
49   { $unionWith: { coll: "OffSiteTransferAggregate", pipeline: offsite }},
50
51
52   { $group: {
53     _id: { Year: "$_id.Year", Sector: "$_id.Sector" },
54     totalReleases: { $sum: "$totalReleases" },
55     facilityCount: { $sum: 1 }
56   }},
57   { $project: {
58     _id: 0,
59     Year: "$_id.Year",
60     IndustrySector: "$_id.Sector",
61     totalReleasesGrams: { $round: ["$totalReleases", 3] },
62     facilityCount: 1
63   }},
64   { $sort: { Year: 1, totalReleasesGrams: -1 } }
65 ]);
66
67
68
69
```

# UPIT 10

```
1
2
3 db.OnSiteReleaseAggregate.aggregate([
4   { $match: { year: 2015 } },
5   { $project: {
6     _id: 0,
7     FacilityId: "$facility.FacilityId",
8     City: "$facility.City",
9     IndustrySector: "$facility.IndustrySector",
10    total: { $ifNull: ["$OnSiteReleaseTotalForAllChemicals", 0] }
11  }},
12   { $unionWith: {
13     coll: "OffSiteTransferAggregate",
14     pipeline: [
15       { $match: { year: 2015 } },
16       { $project: {
17         _id: 0,
18         FacilityId: "$facility.FacilityId",
19         City: "$facility.City",
20         IndustrySector: "$facility.IndustrySector",
```

```
21         total: { $ifNull: ["$OffSiteReleaseTotalForAllChemicals", 0] }
22     }}
23 ]
24 }},
25 { $group: {
26     _id: "$FacilityId",
27     City: { $first: "$City" },
28     IndustrySector: { $first: "$IndustrySector" },
29     totalReleases: { $sum: "$total" }
30 }},
31 { $sort: { City: 1, totalReleases: -1 } },
32 { $group: {
33     _id: "$City",
34     totalReleases: { $sum: "$totalReleases" },
35     facilityCount: { $sum: 1 },
36     dominantSector: { $first: "$IndustrySector" }
37 }},
38 { $set: {
39     avgPerFacility: {
40         $cond: [
41             { $gt: ["$facilityCount", 0] },
42             { $divide: ["$totalReleases", "$facilityCount"] },
43             0
44         ]
45     }
46 }},
47 { $project: {
48     _id: 0,
49     City: "$_id",
50     totalReleasesGrams: { $round: ["$totalReleases", 3] },
51     facilityCount: 1,
52     avgPerFacilityGrams: { $round: ["$avgPerFacility", 3] },
53     dominantSector: 1
54 }},
55 { $sort: { totalReleasesGrams: -1 } }
56 ]});
57
58
```