

1. Знайти загальну вартість, середню вартість, максимальну та мінімальну вартість автомобілів за маркою.

```
task> db["cars"].aggregate([{
  $group: { _id: "$brand",
    totalSum: { $sum: "$price" }, |
    averagePrice: { $avg: "$price" },
    maxPrice: { $max: "$price" },
    minPrice: { $min: "$price" } } }])
```

```
< {
  _id: 'Ford',
  totalSum: 18000,
  averagePrice: 18000,
  maxPrice: 18000,
  minPrice: 18000
}
{
  _id: 'Chevrolet',
  totalSum: 20000,
  averagePrice: 20000,
  maxPrice: 20000,
  minPrice: 20000
}
{
  _id: 'Volkswagen',
  totalSum: 28000,
  averagePrice: 28000,
  maxPrice: 28000,
  minPrice: 28000
}
{
  _id: 'Tesla',
  totalSum: 35000,
  averagePrice: 35000,
  maxPrice: 35000,
  minPrice: 35000
}
{
  _id: 'Mercedes',
  totalSum: 42000,
  averagePrice: 42000,
  maxPrice: 42000,
  minPrice: 42000
}
{
  _id: 'Audi',
  totalSum: 37000,
  averagePrice: 37000,
  maxPrice: 37000,
  minPrice: 37000
}
```

2. Знайти кількість автомобілів за маркою, випущених після 2018 року.

```
task> db["cars"].aggregate([
  {
    $match: {
      year: { $gt: 2018 }
    }
  },
  {
    $group: {
      _id: "$brand",
      count: { $sum: 1 },
    }
  })
```

```
< {
  _id: 'Chevrolet',
  count: 1
}
{
  _id: 'BMW',
  count: 1
}
{
  _id: 'Honda',
  count: 1
}
{
  _id: 'Volkswagen',
  count: 1
}
{
  _id: 'Mercedes',
  count: 1
}
{
  _id: 'Tesla',
  count: 1
}
```

3. Знайти власників з масивами моделей їхніх автомобілів.

```
task> db["owners"].aggregate([
  { $lookup: {
    from: 'cars',
    localField: '_id',
    foreignField: 'ownerId',
    as: 'carsList',
  }
}]
```

```
< {
  _id: 1,
  name: 'John Doe',
  age: 45,
  carsList: [
    {
      _id: 1,
      brand: 'Toyota',
      model: 'Camry',
      year: 2018,
      price: 24000,
      ownerId: 1,
      mileage: 30000,
      insurance: 600
    },
    {
      _id: 4,
      brand: 'Ford',
      model: 'Focus',
      year: 2017,
      price: 18000,
      ownerId: 1,
      mileage: 50000,
      insurance: 450
    },
    {
```

4. Сортувати автомобілі за віком власника.

```
task> db["cars"].aggregate([
  { $lookup: {
    from: 'owners',
    localField: 'ownerId',
    foreignField: '_id',
    as: 'ownerInfo',
  },
  { $unwind: '$ownerInfo' },
  { $sort: { 'ownerInfo.age': 1 } }
])
```

```
< {
  _id: 2,
  brand: 'Honda',
  model: 'Accord',
  year: 2019,
  price: 22000,
  ownerId: 2,
  mileage: 15000,
  insurance: 700,
  ownerInfo: {
    _id: 2,
    name: 'Jane Smith',
    age: 34
  }
}
{
  _id: 5,
  brand: 'Chevrolet',
  model: 'Malibu',
  year: 2019,
  price: 20000,
  ownerId: 2,
  mileage: 40000,
  insurance: 800,
  ownerInfo: {
    _id: 2,
    name: 'Jane Smith',
    age: 34
  }
}
{
  _id: 8,
  brand: 'Audi',
  model: 'A4',
  year: 2018
```

5. Знайти загальну вартість автомобілів кожного власника.

```
task> db["owners"].aggregate([
  { $lookup: {
    from: 'cars',
    localField: '_id',
    foreignField: 'ownerId',
    as: 'carsList',
  }
},
{ $project: {
  name: 1,
  totalCarsSum: { $reduce: {
    input: "$carsList",
    initialValue: 0,
    in: { $add: ['$value', '$$this.price'] }
  } }
}
})
```

```
< {
  _id: 1,
  name: 'John Doe',
  totalCarsSum: 107000
}
{
  _id: 2,
  name: 'Jane Smith',
  totalCarsSum: 79000
}
{
  _id: 3,
  name: 'Elon Musk',
  totalCarsSum: 113000
}
task> |
```

6. Обчислити середній вік автомобілів кожної марки, для чого краще використовувати додаткове поле age, додане до кожного документу.

```
task> db["cars"].aggregate([
  { $addFields: {
    age: { $subtract: [ { $year: '$$NOW' }, '$year' ] }
  } },
  { $group: {
    _id: '$brand',
    averageAge: { $avg: '$age' }
  } }
])
```

```
< {
  _id: 'Mercedes',
  averageAge: 4
}
{
  _id: 'Toyota',
  averageAge: 6
}
{
  _id: 'Nissan',
  averageAge: 7
}
{
  _id: 'Tesla',
  averageAge: 4
}
{
  _id: 'Ford',
  averageAge: 7
}
{
  _id: 'Chevrolet',
  averageAge: 5
}
{
  _id: 'Volkswagen',
  averageAge: 5
}
```

7. Додати поле "expensive", яке приймає значення "True" для машин дорожче \$30000 та "False" для інших.

```
task> db["cars"].aggregate([
  { $addFields: {
    expensive: {
      $cond: {
        if: { $gt: [ '$price', 30000 ]
        then: true,
        else: false,
      }
    }
  } },
])
```

```
< {
  _id: 1,
  brand: 'Toyota',
  model: 'Camry',
  year: 2018,
  price: 24000,
  ownerId: 1,
  mileage: 30000,
  insurance: 600,
  expensive: false
}
{
  _id: 2,
  brand: 'Honda',
  model: 'Accord',
  year: 2019,
  price: 22000,
  ownerId: 2,
  mileage: 15000,
  insurance: 700,
  expensive: false
}
{
  _id: 3,
  brand: 'Tesla',
  model: 'Model 3',
  year: 2020,
  price: 35000,
  ownerId: 3,
  mileage: 20000,
  insurance: 1200,
  expensive: true
}
```

8. Додати поле "ageGroup", яке визначає вік автомобіля у групи: "Новий" (< 3 років), "Середній" (3-6 років) та "Старий" (> 6 років).

```
task> db["cars"].aggregate([
  { $addFields: {
    ageGroup: {
      $switch: {
        branches: [
          {
            case: { $lt: [ { $subtract: [ { $year: '$$NOW' }, '$year' ] ], 3 ] },
            then: 'Новий'
          },
          {
            case: { $lte: [ { $subtract: [ { $year: '$$NOW' }, '$year' ] ], 6 ] },
            then: 'Середній'
          },
          {
            case: { $gt: [ { $subtract: [ { $year: '$$NOW' }, '$year' ] ], 6 ] },
            then: 'Старий'
          }
        ]
      }
    }
  }
}
])
```

>_MONGOSH

```
{
  insurance: 700,
  ageGroup: 'Середній'
}
{
  _id: 3,
  brand: 'Tesla',
  model: 'Model 3',
  year: 2020,
  price: 35000,
  ownerId: 3,
  mileage: 20000,
  insurance: 1200,
  ageGroup: 'Середній'
}
{
  _id: 4,
  brand: 'Ford',
  model: 'Focus',
  year: 2017,
  price: 18000,
  ownerId: 1,
  mileage: 50000,
  insurance: 450,
  ageGroup: 'Старий'
}
{
  _id: 5,
  brand: 'Chevrolet',
  model: 'Malibu',
  year: 2019,
  price: 20000,
  ownerId: 2,
  mileage: 40000,
  insurance: 800,
  ageGroup: 'Середній'
}
{
```


9. Додати поле "priceRange", яке визначає діапазон ціни автомобіля: "Дешевий" (< \$20000), "Середній" (\$20000-\$30000) та "Дорогий" (> \$30000).

```
task> db["cars"].aggregate([
  { $addFields: {
    priceRange: {
      $switch: {
        branches: [
          {
            case: { $lt: [ '$price', 20000 ] },
            then: 'Дешевий'
          },
          {
            case: { $lte: [ '$price', 30000 ] },
            then: 'Середній'
          },
          {
            case: { $gt: [ '$price', 30000 ] },
            then: 'Дорогий'
          }
        ]
      }
    }
  }
],
{ $project: { priceRange: 1 } }
])
```

```
< {
  _id: 1,
  brand: 'Toyota',
  model: 'Camry',
  year: 2018,
  price: 24000,
  ownerId: 1,
  mileage: 30000,
  insurance: 600,
  priceRange: 'Середній'
}
{
  _id: 2,
  brand: 'Honda',
  model: 'Accord',
  year: 2019,
  price: 22000,
  ownerId: 2,
  mileage: 15000,
  insurance: 700,
  priceRange: 'Середній'
}
{
  _id: 3,
  brand: 'Tesla',
  model: 'Model 3',
  year: 2020,
  price: 35000,
  ownerId: 3,
  mileage: 20000,
  insurance: 1200,
  priceRange: 'Дорогий'
}
{
```

10. Додати поле "ownerAgeCategory", яке визначає вікову категорію власника: "Молодий" (< 35 років), "Дорослий" (35-50 років) та "Похилий" (> 50 років).

```
task> db["owners"].aggregate([
  { $addFields: {
    ownerAgeCategory: {
      $switch: {
        branches: [
          {
            case: { $lt: [ '$age', 35 ] },
            then: 'Молодий'
          },
          {
            case: { $and: [
              { $gte: [ '$age', 35 ] },
              { $lte: [ '$age', 50 ] }
            ] },
            then: 'Дорослий'
          },
          {
            case: { $gt: [ '$age', 50 ] },
            then: 'Похилий'
          }
        ]
      }
    }
  }
]
})
```

```
< {
  _id: 1,
  name: 'John Doe',
  age: 45,
  ownerAgeCategory: 'Дорослий'
}
{
  _id: 2,
  name: 'Jane Smith',
  age: 34,
  ownerAgeCategory: 'Молодий'
}
{
  _id: 3,
  name: 'Elon Musk',
  age: 49,
  ownerAgeCategory: 'Дорослий'
}
task>
```

11. Додати поле "serviceNeeded", яке вказує на необхідність обслуговування автомобіля залежно від пробігу: "Потрібне" (> 50000 км), "Не потрібне" (<= 50000 км).

```
task> db["cars"].aggregate([
  { $addFields: {
    serviceNeeded: {
      $cond: {
        if: { $gt: [ '$mileage', 50000 ]
        then: 'Потрібне',
        else: 'Не потрібне',
      }
    }
  } },
])
```

```
{
  _id: 8,
  brand: 'Audi',
  model: 'A4',
  year: 2018,
  price: 37000,
  ownerId: 2,
  mileage: 25000,
  insurance: 1100,
  serviceNeeded: 'Не потрібне'
}
{
  _id: 9,
  brand: 'Volkswagen',
  model: 'Passat',
  year: 2019,
  price: 28000,
  ownerId: 3,
  mileage: 30000,
  insurance: 900,
  serviceNeeded: 'Не потрібне'
}
{
  _id: 10,
  brand: 'Nissan',
  model: 'Altima',
  year: 2017,
  price: 23000,
  ownerId: 1,
  mileage: 55000,
  insurance: 500,
  serviceNeeded: 'Потрібне'
}
task>
```

12. Додати поле "insuranceCost", яке визначає вартість страхування автомобіля: "Низька" (< \$500), "Середня" (\$500-\$1500) та "Висока" (> \$1500).

```
task> db["cars"].aggregate([
  { $addFields: {
    insuranceCost: {
      $switch: {
        branches: [
          {
            case: { $lt: [ '$insurance', 500 ] },
            then: 'Низька'
          },
          {
            case: { $and: [
              { $gte: [ '$insurance', 500 ] },
              { $lte: [ '$insurance', 1500 ] }
            ] },
            then: 'Середня'
          },
          {
            case: { $gt: [ '$insurance', 1500 ] },
            then: 'Висока'
          }
        ]
      }
    }
  }
]
})
```

>_MONGOSH

```
mileage: 15000,
insurance: 700,
insuranceCost: 'Середня'
}
{
  _id: 3,
  brand: 'Tesla',
  model: 'Model 3',
  year: 2020,
  price: 35000,
  ownerId: 3,
  mileage: 20000,
  insurance: 1200,
  insuranceCost: 'Середня'
}
{
  _id: 4,
  brand: 'Ford',
  model: 'Focus',
  year: 2017,
  price: 18000,
  ownerId: 1,
  mileage: 50000,
  insurance: 450,
  insuranceCost: 'Низька'
}
{
  _id: 5,
  brand: 'Chevrolet',
  model: 'Malibu',
  year: 2019,
  price: 20000,
  ownerId: 2,
  mileage: 40000,
  insurance: 800,
  insuranceCost: 'Середня'
}
{
```

