

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

### Запит:

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

### Відповідь:

```
{
  _id: 'Audi',
  totalPrice: 37000,
  avgPrice: 37000,
  minPrice: 37000,
  maxPrice: 37000
}
{
  _id: 'Ford',
  totalPrice: 18000,
  avgPrice: 18000,
  minPrice: 18000,
  maxPrice: 18000
}
{
  _id: 'Volkswagen',
  totalPrice: 28000,
  avgPrice: 28000,
  minPrice: 28000,
  maxPrice: 28000
}
{
  _id: 'Tesla',
  totalPrice: 35000,
  avgPrice: 35000,
  minPrice: 35000,
  maxPrice: 35000
}
{
  _id: 'Honda',
  totalPrice: 22000,
  avgPrice: 22000,
  minPrice: 22000,
  maxPrice: 22000
}
{
  _id: 'BMW',
  totalPrice: 50000,
  avgPrice: 50000,
  minPrice: 50000,
  maxPrice: 50000
}
{
  _id: 'Chevrolet',
  totalPrice: 20000,
  avgPrice: 20000,
  minPrice: 20000,
  maxPrice: 20000
}
{
  _id: 'Toyota',
  totalPrice: 24000,
  avgPrice: 24000,
  minPrice: 24000,
  maxPrice: 24000
}
{
  _id: 'Mercedes',
  totalPrice: 42000,
  avgPrice: 42000,
  minPrice: 42000,
  maxPrice: 42000
}
{
  _id: 'Nissan',
  totalPrice: 23000,
  avgPrice: 23000,
  minPrice: 23000,
  maxPrice: 23000
}
```

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

**Запит:**

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

**Відповідь:**

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

3. Знайти власників з масивами моделей їхніх автомобілів.

Запит:

```
homework13> db.cars.aggregate([
  {
    $lookup: {
      from: "owners",
      localField: "ownerId",
      foreignField: "_id",
      as: "owner"
    }
  },
  {
    $unwind: "$owner"
  },
  {
    $group: {
      _id: "$owner",
      brands: { $push: "$brand" }
    }
  }
])
```

```
{
  _id: {
    _id: 1,
    name: 'John Doe',
    age: 45
  },
  brands: [
    'Toyota',
    'Ford',
    'Mercedes',
    'Nissan'
  ]
},
{
  _id: {
    _id: 2,
    name: 'Jane Smith',
    age: 34
  },
  brands: [
    'Honda',
    'Chevrolet',
    'Audi'
  ]
},
{
  _id: {
    _id: 3,
    name: 'Elon Musk',
    age: 49
  },
  brands: [
    'Tesla',
    'BMW',
    'Volkswagen'
  ]
}
```

Відповідь:

#### 4. Сортувати автомобілі за віком власника.

##### Запит:

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

##### Відповідь:

```
{
  "_id": 2,
  "brand": 'Honda',
  "model": 'Accord',
  "year": 2019,
  "price": 22000,
  "ownerId": 2,
  "mileage": 15000,
  "insurance": 700,
  "owner": {
    "_id": 2,
    "name": 'Jane Smith',
    "age": 34
  }
},
{
  "_id": 5,
  "brand": 'Chevrolet',
  "model": 'Malibu',
  "year": 2019,
  "price": 20000,
  "ownerId": 2,
  "mileage": 40000,
  "insurance": 800,
  "owner": {
    "_id": 2,
    "name": 'Jane Smith',
    "age": 34
  }
},
{
  "_id": 8,
  "brand": 'Audi',
  "model": 'A4',
  "year": 2018,
  "price": 37000,
  "ownerId": 2,
  "mileage": 25000,
  "insurance": 1100,
  "owner": {
    "_id": 2,
    "name": 'Jane Smith',
    "age": 34
  }
},
{
  "_id": 1,
  "brand": 'Toyota',
  "model": 'Camry',
  "year": 2018,
  "price": 24000,
  "ownerId": 1,
  "mileage": 30000,
  "insurance": 600,
  "owner": {
    "_id": 1,
    "name": 'John Doe',
    "age": 45
  }
},
{
  "_id": 4,
  "brand": 'Ford',
  "model": 'Focus',
  "year": 2017,
  "price": 18000,
  "ownerId": 1,
  "mileage": 50000,
  "insurance": 450,
  "owner": {
    "_id": 1,
    "name": 'John Doe',
    "age": 45
  }
},
{
  "_id": 7,
  "brand": 'Mercedes',
  "model": 'C-Class',
  "year": 2020,
  "price": 42000,
  "ownerId": 1,
  "mileage": 12000,
  "insurance": 1300,
  "owner": {
    "_id": 1,
    "name": 'John Doe',
    "age": 45
  }
},
{
  "_id": 10,
  "brand": 'Nissan',
  "model": 'Altima',
  "year": 2017,
  "price": 23000,
  "ownerId": 1,
  "mileage": 55000,
  "insurance": 500,
  "owner": {
    "_id": 1,
    "name": 'John Doe',
    "age": 45
  }
},
{
  "_id": 3,
  "brand": 'Tesla',
  "model": 'Model 3',
  "year": 2020,
  "price": 35000,
  "ownerId": 3,
  "mileage": 20000,
  "insurance": 1200,
  "owner": {
    "_id": 3,
    "name": 'Elon Musk',
    "age": 49
  }
},
{
  "_id": 6,
  "brand": 'BMW',
  "model": 'X5',
  "year": 2021,
  "price": 50000,
  "ownerId": 3,
  "mileage": 10000,
  "insurance": 1500,
  "owner": {
    "_id": 3,
    "name": 'Elon Musk',
    "age": 49
  }
},
{
  "_id": 9,
  "brand": 'Volkswagen',
  "model": 'Passat',
  "year": 2019,
  "price": 28000,
  "ownerId": 3,
  "mileage": 30000,
  "insurance": 900,
  "owner": {
    "_id": 3,
    "name": 'Elon Musk',
    "age": 49
  }
}
```

5. Знайти загальну вартість автомобілів кожного власника.

**Запит:**

```
db.cars.aggregate([
  {
    $lookup: {
      from: "owners",
      localField: "ownerId",
      foreignField: '_id',
      as: "owner"
    }
  },
  {
    $unwind: "$owner"
  },
  {
    $group: {
      _id: "$owner",
      totalCost: { $sum: "$price" }
    }
  }
])
```

**Відповідь:**

```
{
  _id: {
    _id: 3,
    name: 'Elon Musk',
    age: 49
  },
  totalCost: 113000
}
{
  _id: {
    _id: 1,
    name: 'John Doe',
    age: 45
  },
  totalCost: 107000
}
{
  _id: {
    _id: 2,
    name: 'Jane Smith',
    age: 34
  },
  totalCost: 79000
}
```

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

### Запит:

```
db.cars.aggregate([
  {
    $addFields: {
      age: { $subtract: [2024, "$year"] }
    }
  },
  {
    $group: {
      _id: "$brand",
      averageAge: { $avg: "$age" }
    }
  }
])
```

### Відповідь:

```
{
  _id: 'BMW',
  averageAge: 3
}
{
  _id: 'Tesla',
  averageAge: 4
}
{
  _id: 'Ford',
  averageAge: 7
}
{
  _id: 'Honda',
  averageAge: 5
}
{
  _id: 'Nissan',
  averageAge: 7
}
{
  _id: 'Chevrolet',
  averageAge: 5
}
{
  _id: 'Audi',
  averageAge: 6
}
{
  _id: 'Volkswagen',
  averageAge: 5
}
{
  _id: 'Toyota',
  averageAge: 6
}
{
  _id: 'Mercedes',
  averageAge: 4
}
```

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

### Запит:

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

### Відповідь:

```
{
  _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
}
{
  _id: 4,
  brand: 'Ford',
  model: 'Focus',
  year: 2017,
  price: 18000,
  ownerId: 1,
  mileage: 50000,
  insurance: 450,
  expensive: false
}
{
  _id: 5,
  brand: 'Chevrolet',
  model: 'Malibu',
  year: 2019,
  price: 20000,
  ownerId: 2,
  mileage: 40000,
  insurance: 800,
  expensive: false
}
{
  _id: 6,
  brand: 'BMW',
  model: 'X5',
  year: 2021,
  price: 50000,
  ownerId: 3,
  mileage: 10000,
  insurance: 1500,
  expensive: true
}
...
```

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

**Запит:**

```
db.cars.aggregate([
  {
    $addFields: {
      age: { $subtract: [2024, "$year"] }
    }
  },
  {
    $addFields: {
      ageGroup: {
        $switch: {
          branches: [
            { case: { $gt: ["$age", 6] }, then: "Старий" },
            { case: { $and: [{ $gte: ["$age", 3] }, { $lte: ["$age", 6] } ] }, then: "Середній" },
          ],
          default: "Новий"
        }
      }
    }
  },
  {
    $project: {
      age: 0
    }
  }
])
```

**Відповідь:**



```

{
  _id: 1,
  brand: 'Toyota',
  model: 'Camry',
  year: 2018,
  price: 24000,
  ownerId: 1,
  mileage: 30000,
  insurance: 600,
  ageGroup: 'Середній'
}
{
  _id: 2,
  brand: 'Honda',
  model: 'Accord',
  year: 2019,
  price: 22000,
  ownerId: 2,
  mileage: 15000,
  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'
}
{
  _id: 6,
  brand: 'BMW',
  model: 'X5',
  year: 2021,
  price: 50000,
  ownerId: 3,
  mileage: 10000,
  insurance: 1500,
  ageGroup: 'Середній'
}
{
  _id: 7,
  brand: 'Mercedes',
  model: 'C-Class',
  year: 2020,
  price: 42000,
  ownerId: 1,
  mileage: 12000,
  insurance: 1300,
  ageGroup: 'Середній'
}
{
  _id: 8,
  brand: 'Audi',
  model: 'A4',
  year: 2018,
  price: 37000,
  ownerId: 2,
  mileage: 25000,
  insurance: 1100,
  ageGroup: 'Середній'
}
{
  _id: 9,
  brand: 'Volkswagen',
  model: 'Passat',
  year: 2019,
  price: 28000,
  ownerId: 3,
  mileage: 30000,
  insurance: 900,
  ageGroup: 'Середній'
}
{
  _id: 10,
  brand: 'Nissan',
  model: 'Altima',
  year: 2017,
  price: 23000,
  ownerId: 1,
  mileage: 55000,
  insurance: 500,
  ageGroup: 'Старий'
}

```

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

### Запит:

```
db.cars.aggregate([
  {
    $addFields: {
      priceRange: {
        $switch: {
          branches: [
            { case: { $and: [{ $gte: ['$price', 0] }, { $lt: ['$price', 20000] } ] }, then: 'Дешевий' },
            { case: { $and: [{ $gte: ['$price', 20000] }, { $lte: ['$price', 30000] } ] }, then: 'Середній' },
            { case: { $gte: ['$price', 30000] }, then: 'Дорогий' }
          ],
          default: 'Помилка'
        }
      }
    }
  }
])
```

### Відповідь:

{ _id: 1, brand: 'Toyota', model: 'Camry', year: 2018, price: 24000, ownerId: 1, mileage: 30000, insurance: 600, priceRange: 'Середній' }	{ _id: 4, brand: 'Ford', model: 'Focus', year: 2017, price: 18000, ownerId: 1, mileage: 50000, insurance: 450, priceRange: 'Дешевий' }	{ _id: 7, brand: 'Mercedes', model: 'C-Class', year: 2020, price: 42000, ownerId: 1, mileage: 12000, insurance: 1300, priceRange: 'Дорогий' }
{ _id: 2, brand: 'Honda', model: 'Accord', year: 2019, price: 22000, ownerId: 2, mileage: 15000, insurance: 700, priceRange: 'Середній' }	{ _id: 5, brand: 'Chevrolet', model: 'Malibu', year: 2019, price: 20000, ownerId: 2, mileage: 40000, insurance: 800, priceRange: 'Середній' }	{ _id: 8, brand: 'Audi', model: 'A4', year: 2018, price: 37000, ownerId: 2, mileage: 25000, insurance: 1100, priceRange: 'Дорогий' }
{ _id: 3, brand: 'Tesla', model: 'Model 3', year: 2020, price: 35000, ownerId: 3, mileage: 20000, insurance: 1200, priceRange: 'Дорогий' }	{ _id: 6, brand: 'BMW', model: 'X5', year: 2021, price: 50000, ownerId: 3, mileage: 10000, insurance: 1500, priceRange: 'Дорогий' }	{ _id: 9, brand: 'Volkswagen', model: 'Passat', year: 2019, price: 28000, ownerId: 3, mileage: 30000, insurance: 900, priceRange: 'Середній' }
		{ _id: 10, brand: 'Nissan', model: 'Altima', year: 2017, price: 23000, ownerId: 1, mileage: 55000, insurance: 500, priceRange: 'Середній' }

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

### Запит:

```
db.cars.aggregate([
  {
    $lookup: {
      from: "owners",
      localField: "ownerId",
      foreignField: "_id",
      as: "owner"
    }
  },
  { $unwind: "$owner" },
  {
    $addFields: {
      ownerAgeCategory: {
        $switch: {
          branches: [
            { case: { $lt: ["$owner.age", 35] }, then: "Молодий" },
            { case: { $and: [{ $gte: ["$owner.age", 35] }, { $lte: ["$owner.age", 50] } ] }, then: "Дорослий" },
            { case: { $gt: ["$owner.age", 50] }, then: "Похилий" }
          ],
          default: false
        }
      }
    }
  }
])
```

### Відповідь:

```
{
  "_id": 1,
  "brand": 'Toyota',
  "model": 'Camry',
  "year": 2018,
  "price": 24000,
  "ownerId": 1,
  "mileage": 30000,
  "insurance": 600,
  "owner": {
    "_id": 1,
    "name": 'John Doe',
    "age": 45
  },
  "ownerAgeCategory": 'Дорослий'
},
{
  "_id": 2,
  "brand": 'Honda',
  "model": 'Accord',
  "year": 2019,
  "price": 22000,
  "ownerId": 2,
  "mileage": 15000,
  "insurance": 700,
  "owner": {
    "_id": 2,
    "name": 'Jane Smith',
    "age": 34
  },
  "ownerAgeCategory": 'Молодий'
},
{
  "_id": 3,
  "brand": 'Tesla',
  "model": 'Model 3',
  "year": 2020,
  "price": 35000,
  "ownerId": 3,
  "mileage": 20000,
  "insurance": 1200,
  "owner": {
    "_id": 3,
    "name": 'Elon Musk',
    "age": 49
  },
  "ownerAgeCategory": 'Дорослий'
},
...
```

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

### Запит:

```
db.cars.aggregate([
  {
    $addFields: {
      serviceNeeded: {
        $switch: {
          branches: [
            { case: { $gt: ["$mileage", 50000] }, then: "Потрібне" },
            { case: { $lte: ["$mileage", 50000] }, then: "Не потрібне" }
          ],
          default: false
        }
      }
    }
  }
])
```

### Відповідь:

```
{
  _id: 1,
  brand: 'Toyota',
  model: 'Camry',
  year: 2018,
  price: 24000,
  ownerId: 1,
  mileage: 30000,
  insurance: 600,
  serviceNeeded: 'Не потрібне'
}
{
  _id: 2,
  brand: 'Honda',
  model: 'Accord',
  year: 2019,
  price: 22000,
  ownerId: 2,
  mileage: 15000,
  insurance: 700,
  serviceNeeded: 'Не потрібне'
}
{
  _id: 4,
  brand: 'Ford',
  model: 'Focus',
  year: 2017,
  price: 18000,
  ownerId: 1,
  mileage: 50000,
  insurance: 450,
  serviceNeeded: 'Не потрібне'
}
{
  _id: 10,
  brand: 'Nissan',
  model: 'Altima',
  year: 2017,
  price: 23000,
  ownerId: 1,
  mileage: 55000,
  insurance: 500,
  serviceNeeded: 'Потрібне'
}
...
```

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

### Запит:

```
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" }
          ],
          default: false
        }
      }
    }
  },
])
```

### Відповідь:

<pre>{   _id: 1,   brand: 'Toyota',   model: 'Camry',   year: 2018,   price: 24000,   ownerId: 1,   mileage: 30000,   insurance: 600,   insuranceCost: 'Середня' }</pre>	<pre>{   _id: 4,   brand: 'Ford',   model: 'Focus',   year: 2017,   price: 18000,   ownerId: 1,   mileage: 50000,   insurance: 450,   insuranceCost: 'Низька' }</pre>	<pre>{   _id: 6,   brand: 'BMW',   model: 'X5',   year: 2021,   price: 50000,   ownerId: 3,   mileage: 10000,   insurance: 1500,   insuranceCost: 'Середня' }</pre>
<pre>{   _id: 10,   brand: 'Nissan',   model: 'Altima',   year: 2017,   price: 23000,   ownerId: 1,   mileage: 55000,   insurance: 500,   insuranceCost: 'Середня' }</pre>	...	