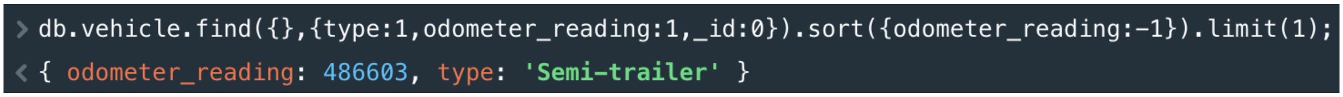
Query 1:

```
db.vehicle.find({},{type:1,odometer_reading:1,_id:0}).sort({odometer_reading:-1}).limit(1)
;
```

Output:



```
> db.vehicle.find({},{type:1,odometer_reading:1,_id:0}).sort({odometer_reading:-1}).limit(1);
< { odometer_reading: 486603, type: 'Semi-trailer' }
```

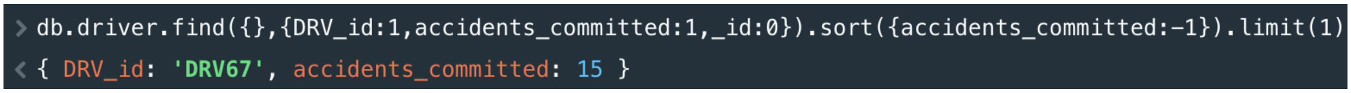
Analytic Purpose:

From this query, it is observed that Semi-trailer trucks are being preferred the most by clients. This implies that the demand is high for these trucks relative to other types of trucks. Therefore, this can be considered while purchasing new trucks in the future.

Query 2:

```
db.driver.find({},{DRV_id:1,accidents_committed:1,_id:0}).sort({accidents_committed:-1}
).limit(1);
```

Output:



```
> db.driver.find({},{DRV_id:1,accidents_committed:1,_id:0}).sort({accidents_committed:-1}).limit(1)
< { DRV_id: 'DRV67', accidents_committed: 15 }
```

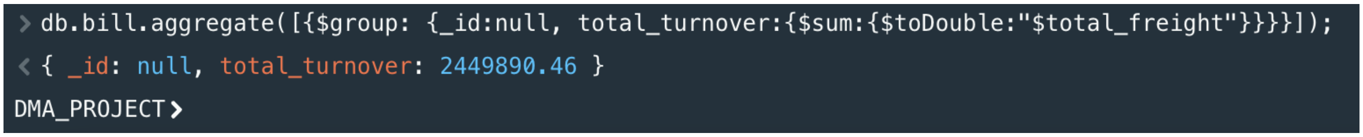
Analytic Purpose:

This query returns those drivers who have committed the most number of accidents till date. It can be used to provide a warning to these drivers, and they can be put under notice for a period of time, until they clear their record (by assigning them shorter trips or reduction in their salary).

Query 3:

```
db.bill.aggregate([
                {
                        $group:
                        {_id:null, total_turnover: {$sum:{$toDouble:"$total_freight"}}}
                }
]);
```

Output:



```
> db.bill.aggregate([{$group: {_id:null, total_turnover:{$sum:{$toDouble:"$total_freight"}}}}]);
< { _id: null, total_turnover: 2449890.46 }
DMA_PROJECT>
```

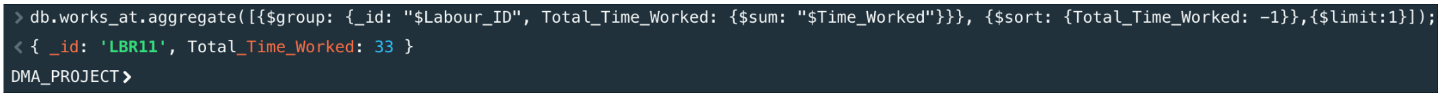
Analytic Purpose:

This query gives the company’s total turnover till date. It can be used to compute average yearly income for the company, and can be compared with the actual yearly income, to compute during which years the company performed above average or below average. It also represents the company’s status in the market.

Query 4:

```
db.works_at.aggregate([
                        {
                                $group: {_id: "$Labour_ID",
                                          Total_Time_Worked: {$sum: "$Time_Worked"}}
                        },
                        {
                                $sort: {Total_Time_Worked: -1}
                        },
                        {
                                $limit:1
                        }
]);
```

Output:



```
> db.works_at.aggregate([{$group: {_id: "$Labour_ID", Total_Time_Worked: {$sum: "$Time_Worked"}}}, {$sort: {Total_Time_Worked: -1}},{$limit:1}]);
< { _id: 'LBR11', Total_Time_Worked: 33 }
DMA_PROJECT>
```

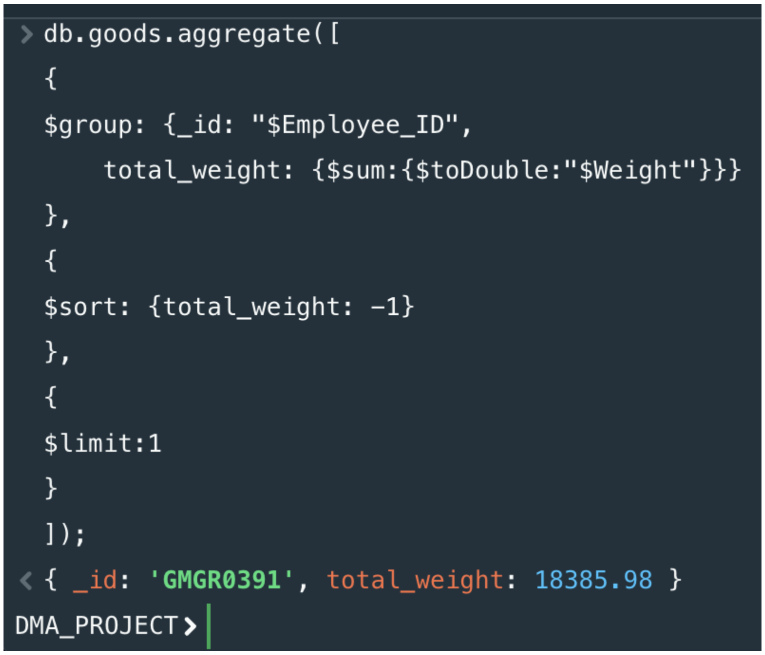
Analytic Purpose:

This query is used to find the laborer who has worked for the most number of hours till date. He can be given the liberty to choose his own shifts, or he could be given an extra leave. If this particular laborer continues to work diligently even in the future, his hourly wage can be increased; he can be rewarded with a loyalty bonus.

Query 5:

```
db.goods.aggregate([
		{
			$group: {_id: "$Employee_ID",
					total_weight: {$sum:{$toDouble:"$Weight"}}}
		},
		{
			$sort: {total_weight: -1}
		},
		{
			$limit:1
		}
]);
```

Output:



```
> db.goods.aggregate([
  {
  $group: {_id: "$Employee_ID",
      total_weight: {$sum:{$toDouble:"$Weight"}}}
  },
  {
  $sort: {total_weight: -1}
  },
  {
  $limit:1
  }
  ]);
< { _id: 'GMGR0391', total_weight: 18385.98 }
DMA_PROJECT>
```

Analytic Purpose:

This query returns the name of the goods manager who has prioritized our company for transportation over the rest of the competition. He can be offered extra commission to recommend our services to his client/customer list.