

Assignment No.10

Name : Shailesh Pawar

PRN : 123B1B230

1. Display the salary of employees department wise

```
var map1 = function () {  
    emit(this.dname, this.salary);  
};
```

```
var reduce1 = function (key, values) {  
    return Array.sum(values);  
};
```

```
db.Teachers_230.mapReduce(map1, reduce1, { out: "dept_salary_total_230" });
```

```
customer_230> var map1 = function () {  
...   emit(this.dname, this.salary);  
... };  
...  
... var reduce1 = function (key, values) {  
...   return Array.sum(values);  
... };  
...  
... db.Teachers_230.mapReduce(map1, reduce1, { out: "dept_salary_total_230" });  
...  
DeprecationWarning: Collection.mapReduce() is deprecated. Use an aggregation instead.  
See https://docs.mongodb.com/manual/core/map-reduce for details.  
{ result: 'dept_salary_total_230', ok: 1 }
```

2. Display the count of employees departmentwise 3. Consider the collection “Orders “ with the following types of documents

```
var map2 = function () {  
    emit(this.dname, 1);  
};
```

```
var reduce2 = function (key, values) {  
    return Array.sum(values);  
};
```

```
db.Teachers_230.mapReduce(map2, reduce2, { out: "dept_emp_count_230" });
```

```
customer_230> var map2 = function () {  
...   emit(this.dname, 1);  
... };  
...  
... var reduce2 = function (key, values) {  
...   return Array.sum(values);  
... };  
...  
... db.Teachers_230.mapReduce(map2, reduce2, { out: "dept_emp_count_230" });  
...  
{ result: 'dept_emp_count_230', ok: 1 }  
customer_230> █
```

3. Consider the collection “Orders “ with the following types of documents.

```
var map3 = function () {  
  emit(this.cust_id, this.price);  
};
```

```
var reduce3 = function (key, values) {  
  return Array.sum(values);  
};
```

```
db.Orders_230.mapReduce(map3, reduce3, { out: "total_price_per_customer_230" });
```

```
customer_230> var map3 = function () {  
...   emit(this.cust_id, this.price);  
... };  
...  
... var reduce3 = function (key, values) {  
...   return Array.sum(values);  
... };  
...  
... db.Orders_230.mapReduce(map3, reduce3, { out: "total_price_per_customer_230" });  
...  
{ result: 'total_price_per_customer_230', ok: 1 }  
customer_230> █
```

4. Display the total salary of employees working in computer department.
var map4 = function () {

```
  if (this.dname === "COMP") {  
    emit("Computer_Department", this.salary);
```

```
}  
};
```

```
var reduce4 = function (key, values) {  
  return Array.sum(values);  
};
```

```
db.Teachers_230.mapReduce(map4, reduce4, { out: "comp_dept_salary_230" });
```

```
customer_230> var map4 = function () {  
...   if (this.dname === "COMP") {  
...     emit("Computer_Department", this.salary);  
...   }  
... };  
...  
... var reduce4 = function (key, values) {  
...   return Array.sum(values);  
... };  
...  
... db.Teachers_230.mapReduce(map4, reduce4, { out: "comp_dept_salary_230" });  
...  
{ result: 'comp_dept_salary_230', ok: 1 }  
customer_230> 
```

5. Consider each document in the zipcode collection has the following form:

```
var map5 = function () {  
  if (this.state === "NY") {  
    emit(this.city, this.pop);  
  }  
};
```

```
var reduce5 = function (key, values) {  
  return Array.sum(values);  
};
```

```
db.zipcode_230.mapReduce(map5, reduce5, { out: "ny_city_pop_230" });
```

```
customer_230> var map5 = function () {  
...   if (this.state === "NY") {  
...     emit(this.city, this.pop);  
...   }  
... };  
...  
... var reduce5 = function (key, values) {  
...   return Array.sum(values);  
... };  
...  
... db.zipcode_230.mapReduce(map5, reduce5, { out: "ny_city_pop_230" });  
...  
{ result: 'ny_city_pop_230', ok: 1 }  
customer_230> 
```