


## Exercise set 7 - NoSQL

- In this exercise you will create a MongoDB document collections including data similar to Ocelot training database used in SQL exercises
- After the creation of document collections data will be inserted
- Finally couple of queries will be formed for data retrieval
- Return exercise in .txt format and use formatting presented below (question always followed be the query)

```
// Select all employee documents from document collection  
db.emps.find();
```

### DEPT

Dept **varchar(4)** \*   
Manager **varchar(12)**  
Duty **varchar(12)**  
Budget **decimal(10,2)**

### EMPS

Empnum **decimal(3,0)** \*   
Dept **varchar(4)**  
Surname **varchar(12)**  
Gname **varchar(9)**  
Address **varchar(18)**  
City **varchar(10)**  
Prov **char(2)**  
Pc **varchar(6)**  
Phone **char(8)**  
Rate **decimal(6,2)**  
Taxcode **decimal(6,2)**  
Deduction **decimal(6,2)**

1. Create two document collections and use SQL table structure presented above as a reference. Add at least five rows of data to both document collections. **Important:** Mongo documents must have `_id` as primary key so use it (dept and empnum will be normal key values)!
2. Create the following queries:
  - a. Select all departments.
  - b. Select employees whose rate value is greater than 9. Order results by rate value so that smallest value is presented first.
  - c. Select employees whose rate value is between 8-12. Order results by rate value so that greatest value is presented first.
  - d. Select employees whose department is not A, B or C.
  - e. Update rate value for all employees working in department B.
  - f. Change the manager for department C so that the new manager will be David Smith.
  - g. Remove employees whose department is C.