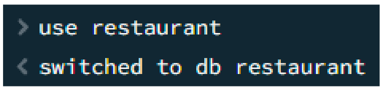
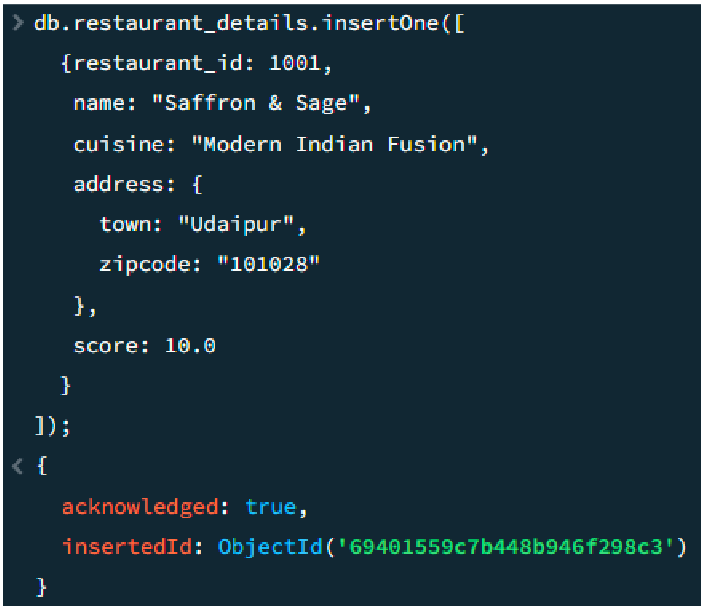
Write NoSQL Queries on “Restaurant” collection.

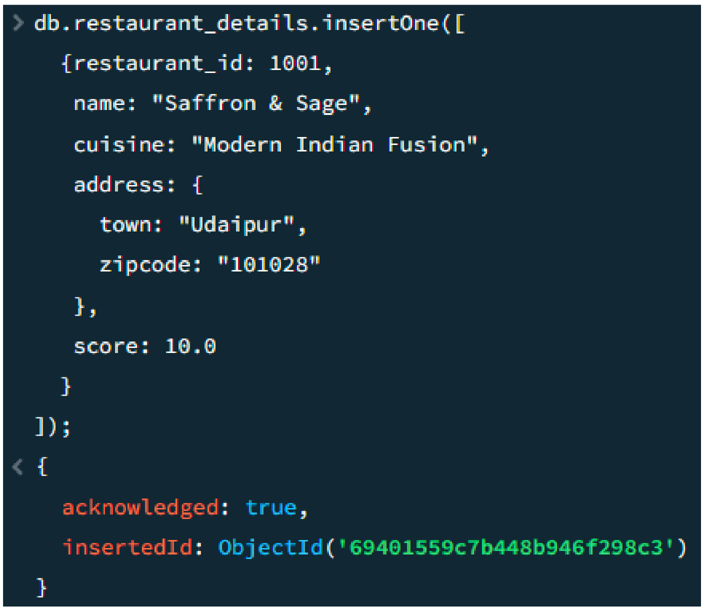


```
> use restaurant
< switched to db restaurant
```

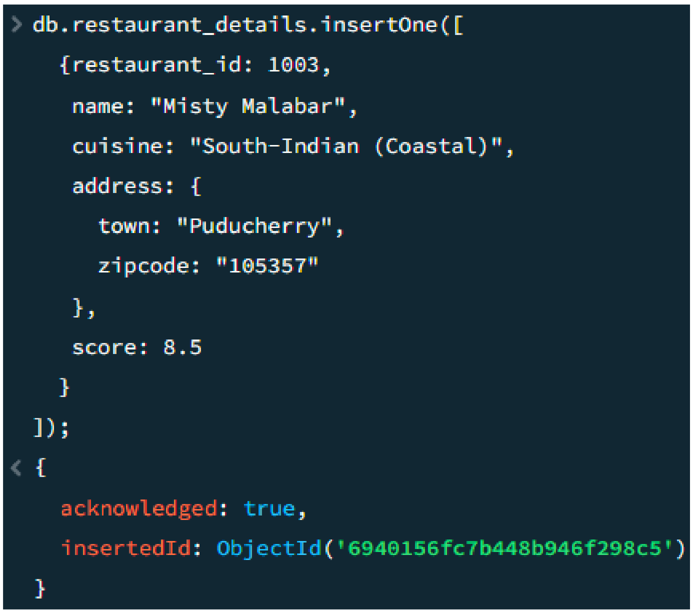
Insert at least 4 documents each with Restaurant_id, Name, Cuisine, Address (Town, Zipcode), Score attributes



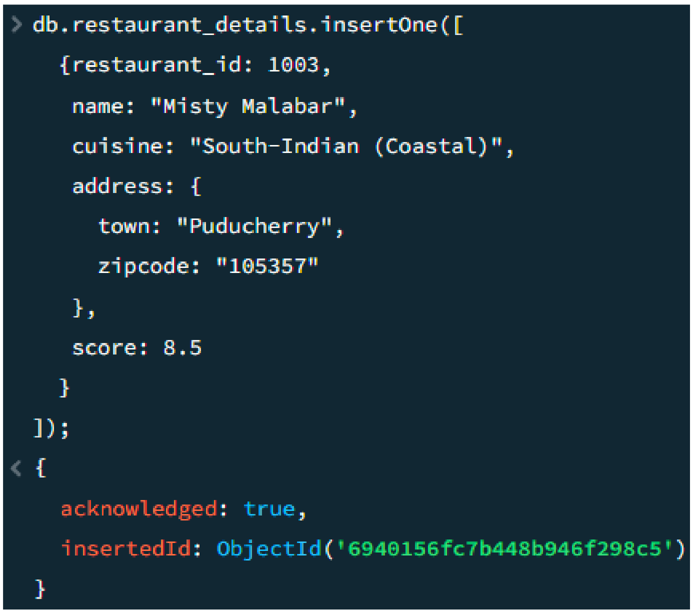
```
> db.restaurant_details.insertOne([
    {restaurant_id: 1001,
     name: "Saffron & Sage",
     cuisine: "Modern Indian Fusion",
     address: {
       town: "Udaipur",
       zipcode: "101028"
     },
     score: 10.0
    }
  ]);
< {
    acknowledged: true,
    insertedId: ObjectId('69401559c7b448b946f298c3')
  }
```



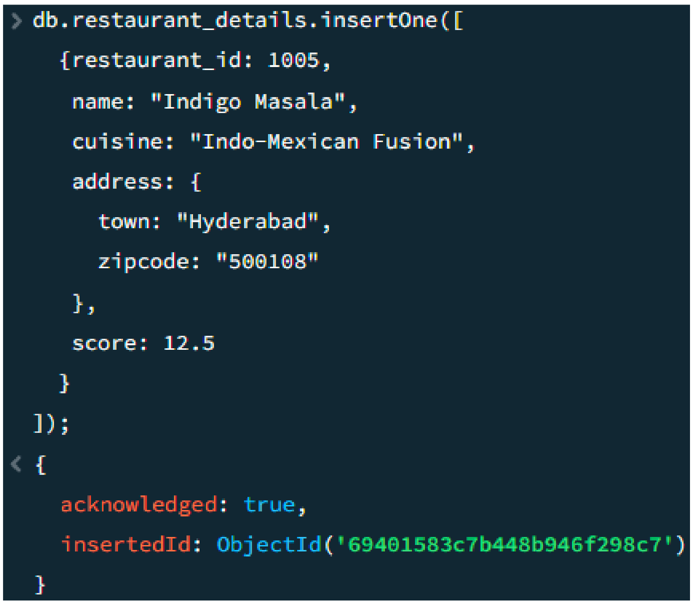
```
> db.restaurant_details.insertOne([
    {restaurant_id: 1001,
     name: "Saffron & Sage",
     cuisine: "Modern Indian Fusion",
     address: {
       town: "Udaipur",
       zipcode: "101028"
     },
     score: 10.0
    }
  ]);
< {
    acknowledged: true,
    insertedId: ObjectId('69401559c7b448b946f298c3')
  }
```



```
> db.restaurant_details.insertOne([
    {restaurant_id: 1003,
     name: "Misty Malabar",
     cuisine: "South-Indian (Coastal)",
     address: {
       town: "Puducherry",
       zipcode: "105357"
     },
     score: 8.5
    }
  ]);
< {
    acknowledged: true,
    insertedId: ObjectId('6940156fc7b448b946f298c5')
  }
```

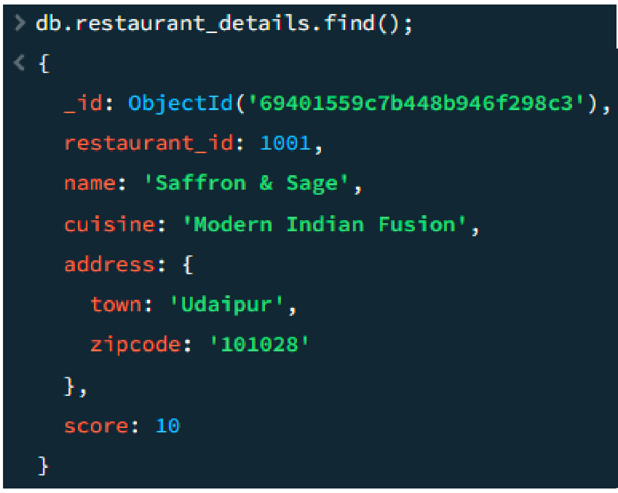


```
> db.restaurant_details.insertOne([
    {restaurant_id: 1003,
     name: "Misty Malabar",
     cuisine: "South-Indian (Coastal)",
     address: {
       town: "Puducherry",
       zipcode: "105357"
     },
     score: 8.5
    }
  ]);
< {
    acknowledged: true,
    insertedId: ObjectId('6940156fc7b448b946f298c5')
  }
```

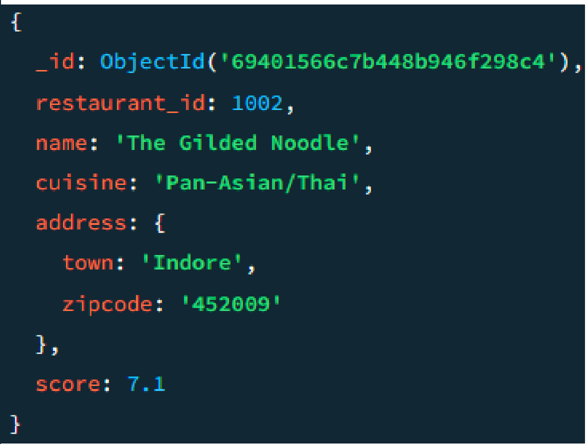


```
> db.restaurant_details.insertOne([
    {restaurant_id: 1005,
     name: "Indigo Masala",
     cuisine: "Indo-Mexican Fusion",
     address: {
       town: "Hyderabad",
       zipcode: "500108"
     },
     score: 12.5
    }
  ]);
< {
    acknowledged: true,
    insertedId: ObjectId('69401583c7b448b946f298c7')
  }
```

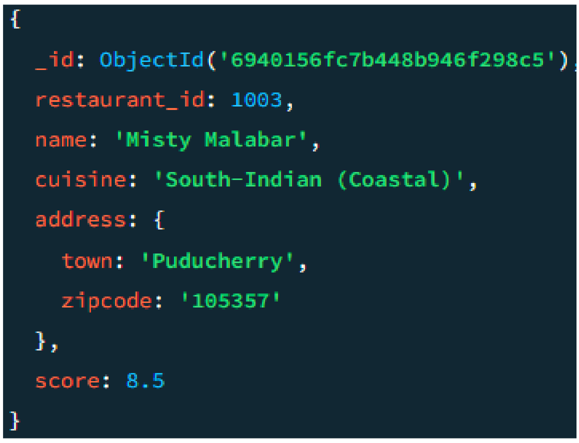
Write a MongoDB query to display all the documents in the collection restaurants.



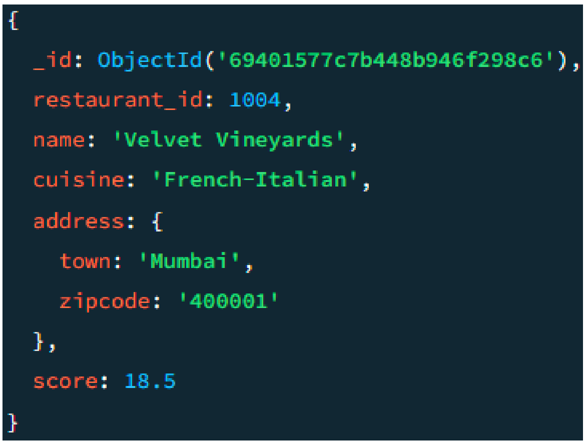
```
> db.restaurant_details.find();
< {
    _id: ObjectId('69401559c7b448b946f298c3'),
    restaurant_id: 1001,
    name: 'Saffron & Sage',
    cuisine: 'Modern Indian Fusion',
    address: {
      town: 'Udaipur',
      zipcode: '101028'
    },
    score: 10
  }
```



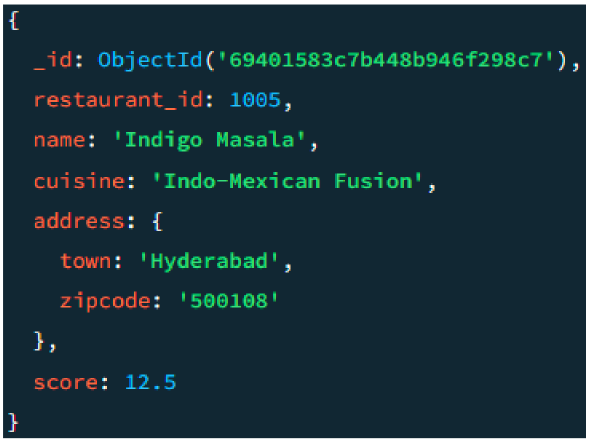
```
{
  _id: ObjectId('69401566c7b448b946f298c4'),
  restaurant_id: 1002,
  name: 'The Gilded Noodle',
  cuisine: 'Pan-Asian/Thai',
  address: {
    town: 'Indore',
    zipcode: '452009'
  },
  score: 7.1
}
```



```
{
  _id: ObjectId('6940156fc7b448b946f298c5'),
  restaurant_id: 1003,
  name: 'Misty Malabar',
  cuisine: 'South-Indian (Coastal)',
  address: {
    town: 'Puducherry',
    zipcode: '105357'
  },
  score: 8.5
}
```

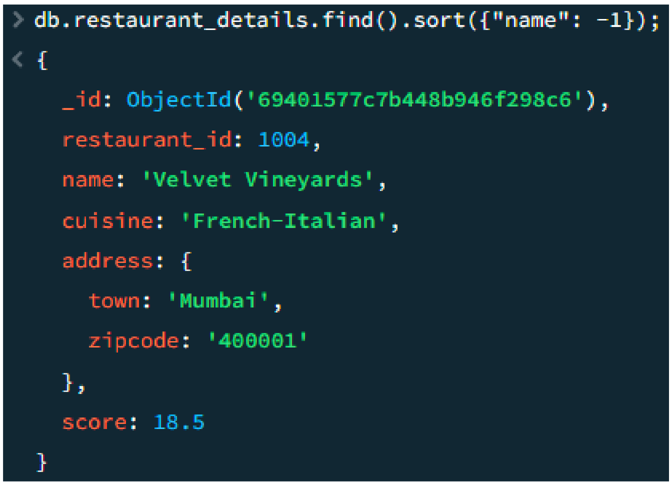


```
{
  _id: ObjectId('69401577c7b448b946f298c6'),
  restaurant_id: 1004,
  name: 'Velvet Vineyards',
  cuisine: 'French-Italian',
  address: {
    town: 'Mumbai',
    zipcode: '400001'
  },
  score: 18.5
}
```

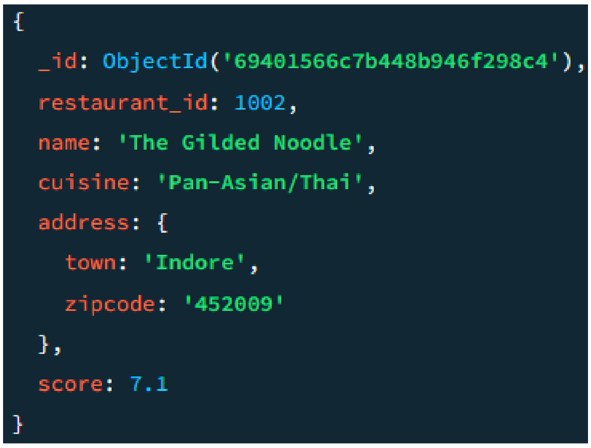


```
{
  _id: ObjectId('69401583c7b448b946f298c7'),
  restaurant_id: 1005,
  name: 'Indigo Masala',
  cuisine: 'Indo-Mexican Fusion',
  address: {
    town: 'Hyderabad',
    zipcode: '500108'
  },
  score: 12.5
}
```

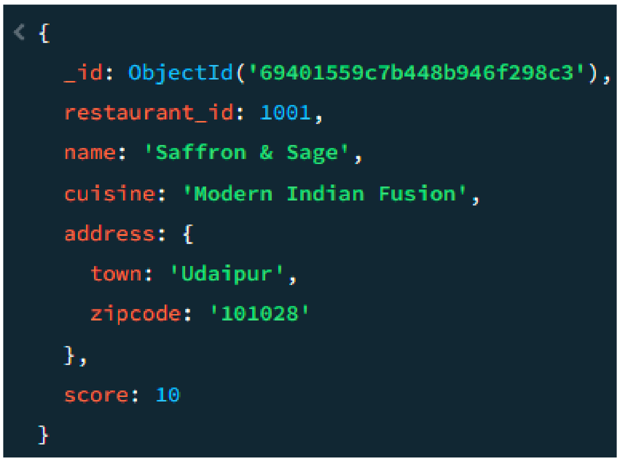
Write a MongoDB query to arrange the name of the restaurants in descending along with all the columns.



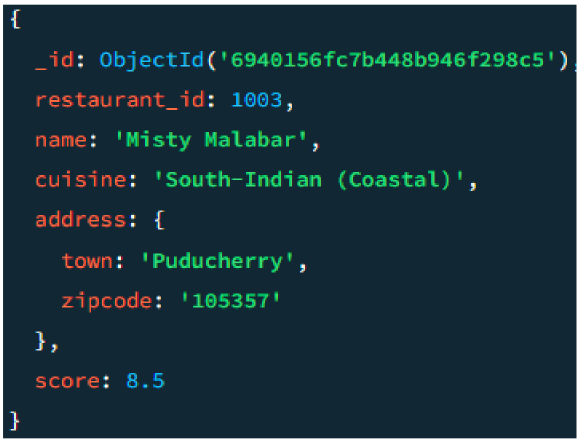
```
> db.restaurant_details.find().sort({"name": -1});
< {
    _id: ObjectId('69401577c7b448b946f298c6'),
    restaurant_id: 1004,
    name: 'Velvet Vineyards',
    cuisine: 'French-Italian',
    address: {
      town: 'Mumbai',
      zipcode: '400001'
    },
    score: 18.5
  }
```



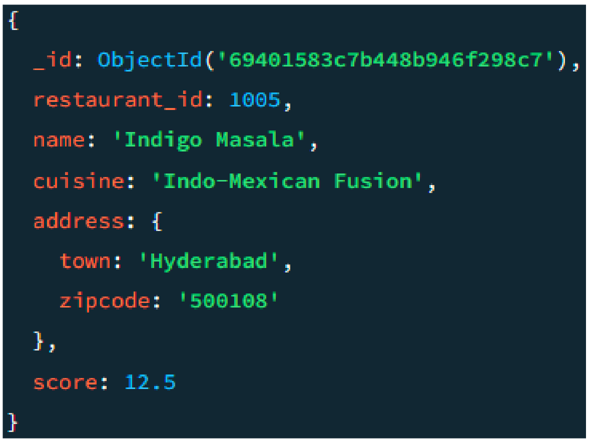
```
{
  _id: ObjectId('69401566c7b448b946f298c4'),
  restaurant_id: 1002,
  name: 'The Gilded Noodle',
  cuisine: 'Pan-Asian/Thai',
  address: {
    town: 'Indore',
    zipcode: '452009'
  },
  score: 7.1
}
```



```
< {
    _id: ObjectId('69401559c7b448b946f298c3'),
    restaurant_id: 1001,
    name: 'Saffron & Sage',
    cuisine: 'Modern Indian Fusion',
    address: {
      town: 'Udaipur',
      zipcode: '101028'
    },
    score: 10
  }
```

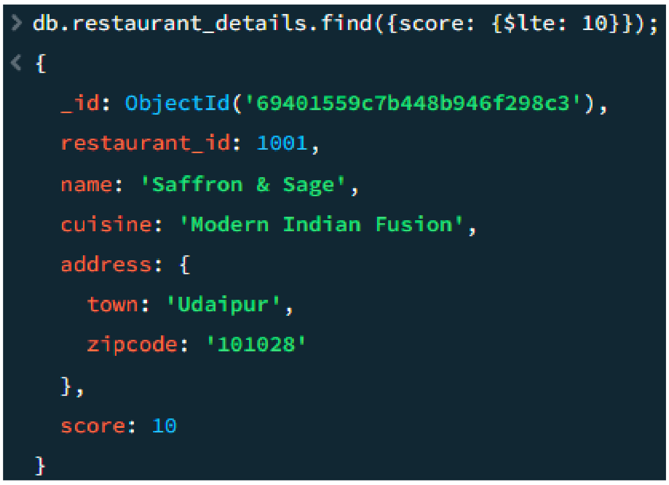


```
{
  _id: ObjectId('6940156fc7b448b946f298c5')
  restaurant_id: 1003,
  name: 'Misty Malabar',
  cuisine: 'South-Indian (Coastal)',
  address: {
    town: 'Puducherry',
    zipcode: '105357'
  },
  score: 8.5
}
```

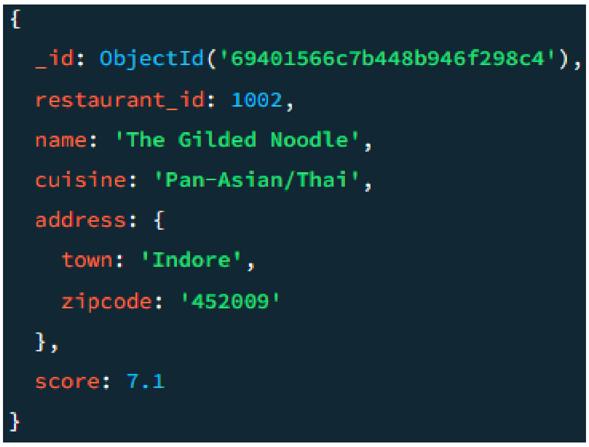


```
{
  _id: ObjectId('69401583c7b448b946f298c7'),
  restaurant_id: 1005,
  name: 'Indigo Masala',
  cuisine: 'Indo-Mexican Fusion',
  address: {
    town: 'Hyderabad',
    zipcode: '500108'
  },
  score: 12.5
}
```

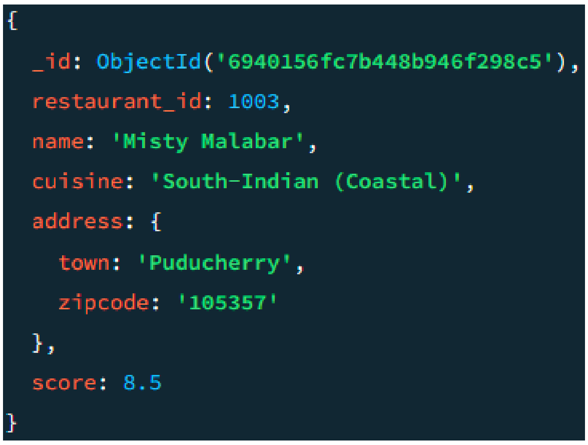
Write a MongoDB query to find the restaurant Id, name, town and cuisine for those restaurants which achieved a score which is not more than 10.



```
> db.restaurant_details.find({score: {$lte: 10}});
< {
    _id: ObjectId('69401559c7b448b946f298c3'),
    restaurant_id: 1001,
    name: 'Saffron & Sage',
    cuisine: 'Modern Indian Fusion',
    address: {
      town: 'Udaipur',
      zipcode: '101028'
    },
    score: 10
  }
```

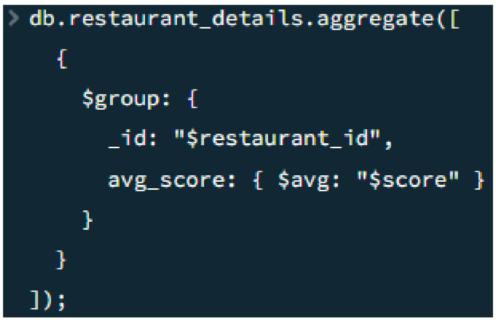


```
{
  _id: ObjectId('69401566c7b448b946f298c4'),
  restaurant_id: 1002,
  name: 'The Gilded Noodle',
  cuisine: 'Pan-Asian/Thai',
  address: {
    town: 'Indore',
    zipcode: '452009'
  },
  score: 7.1
}
```

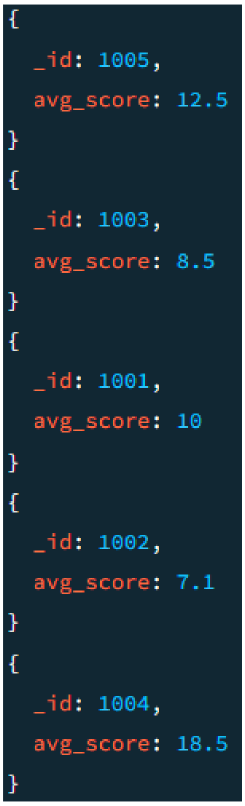


```
{
  _id: ObjectId('6940156fc7b448b946f298c5'),
  restaurant_id: 1003,
  name: 'Misty Malabar',
  cuisine: 'South-Indian (Coastal)',
  address: {
    town: 'Puducherry',
    zipcode: '105357'
  },
  score: 8.5
}
```

Write a MongoDB query to find the average score for each restaurant.

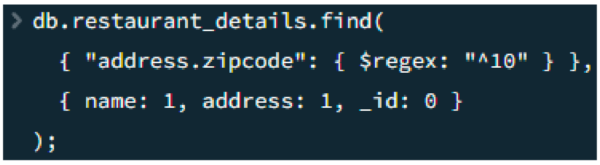


```
> db.restaurant_details.aggregate([
    {
      $group: {
        _id: "$restaurant_id",
        avg_score: { $avg: "$score" }
      }
    }
  ]);
```

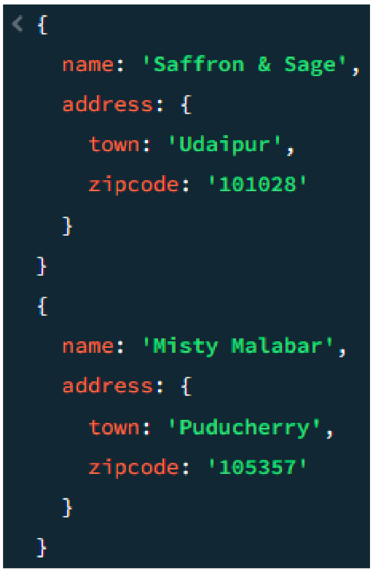


```
{
  _id: 1005,
  avg_score: 12.5
}
{
  _id: 1003,
  avg_score: 8.5
}
{
  _id: 1001,
  avg_score: 10
}
{
  _id: 1002,
  avg_score: 7.1
}
{
  _id: 1004,
  avg_score: 18.5
}
```

Write a MongoDB query to find the name and address of the restaurants that have a zip code that starts with '10'.



```
> db.restaurant_details.find(
    { "address.zipcode": { $regex: "^10" } },
    { name: 1, address: 1, _id: 0 }
  );
```



```
< {
    name: 'Saffron & Sage',
    address: {
      town: 'Udaipur',
      zipcode: '101028'
    }
  }
  {
    name: 'Misty Malabar',
    address: {
      town: 'Puducherry',
      zipcode: '105357'
    }
  }
```