

## Book-model:

```
var mongoose = require('mongoose');
var Schema = mongoose.Schema;

var BookSchema = new Schema({
  title: String,
  author: String,
  category: String,
  published : {
    type:Date,
    default: Date.now
  },
  Keywords: Array,
  published: Boolean,
  author:{
    type:Schema.ObjectId,
    ref: 'User'
  },
  // Embedded sub-document
  detail:{
    modelNumber: Number,
    hardcover: Boolean,
    reviews: Number,
    rank: Number
  }
});

module.exports = mongoose.model('Book', BookSchema);
```

## Book.model:

```
var mongoose = require('mongoose');
var Schema = mongoose.Schema;

var BookSchema = new Schema({
  title: String,
  author: String,
  category: String
});

module.exports = mongoose.model('Book', BookSchema);
```

### **Name.model:**

```
var mongoose = require('mongoose');
var Schema = mongoose.Schema;

var NameSchema = new Schema({
  name: String
});

module.exports = mongoose.model('Name', NameSchema);
```

### **Node First:**

```
const http = require('http');
const url = require('url');
var port = 3005;

var server = http.createServer(function(request, response){
  response.writeHead(200, {'Content-Type': 'text/plain'});
  var q = url.parse(request.url, true).query;
  // response.write(request.url);
  // response.end();
  // response.end("Node Code is working!");
  var txt = q.year + " " + q.month;
  response.end(txt);
});

server.listen(port,function(){
  console.log("Server listening to port: " + port);
});

console.log("Hello World");
// console.log(window);
```

### **AppNode:**

```
const bodyParser = require('body-parser');
var express = require('express');
var app = express();
var mongoose = require('mongoose');
var Book = require('./Book.model');
var db = 'mongodb://localhost:27017/testdb';
mongoose.connect(db);

app.use(bodyParser.json())
```

```
app.use(bodyParser.urlencoded({
  extended:true
}));
```

```
app.get('/', function(req,res){
  res.send("Happy to be here");
});
```

```
app.get('/getBooks', function(req,res){
  console.log('getting all books');
  Book.find({})
  .exec(function(err,books){
    if(err){
      res.send('error has occured');
    } else {
      console.log(books);
      res.json(books);
    }
  });
});
```

```
app.get('/books/:id', function(req,res){
  console.log('getting one book');
  Book.findOne({
    _id: req.params.id
  })
  .exec(function(err,book){
    if(err){
      res.send('error occured');
    } else {
      console.log(book);
      res.json(book);
    }
  })
});
```

```
app.post('/book', function(req,res){
  var newBook = new Book();
  newBook.title = req.body.title;
  newBook.author = req.body.author;
  newBook.category = req.body.category;
  newBook.save(function(err,book){
    if(err){
      res.send('error saving book');
    }
  });
});
```

```

        } else {
            console.log(book);
            res.send(book);
        }
    })
})

app.put('/book/:id', function(req,res){
    Book.findOneAndUpdate({
        _id:req.params.id
    },
    {$set: {title:req.body.title}},
    {upsert:true},
    function(err, newBook){
        if(err){
            console.log("Error occured");
        } else {
            console.log(newBook);
            res.send(newBook);
        }
    });
});

```

```

app.delete('/book/:id', function(req,res){
    Book.findOneAndRemove({
        _id:req.params.id
    }, function(err,book){
        if(err){
            res.send('Error Deleting');
        } else {
            console.log(book);
            res.status(204);
        }
    })
})

```

```

var port = 8080;

```

```

app.listen(port, function(){
    console.log("App listening on port " + port);
});

```