

Building a Model



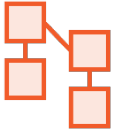
Mark Scott

AUTHOR

@tripletdad99



Introduction



Build a model from the mongoose schema



Relationship between Mongoose and MongoDB documents



Construct and save documents to MongoDB



Our First Model



Customer Model Example

```
const mongoose = require('mongoose');

const customerSchema = new mongoose.Schema({
  name:      String,
  address:   String,
  city:      String,
  state:     String,
  country:   String,
  zipCode:   Number,
  createdAt: Date,
  isActive:  Boolean
});

// Build a model from the customer schema
let Customer = mongoose.model('Customer', customerSchema);

// Add a custom property to the schema
customerSchema.add({ discountCode: String});

const DiscountedCust = mongoose.model('DiscountedCust', customerSchema);
```



Standup Model

```
// standup.js model file
const mongoose = require('mongoose');

const standupSchema = new mongoose.Schema({
  teamMemberId:    mongoose.Schema.Types.ObjectId
  teamMember:      String,
  project:         String,
  workYesterday:   String,
  workToday:       String,
  impediment:      String,
  createdOn:       Date
});

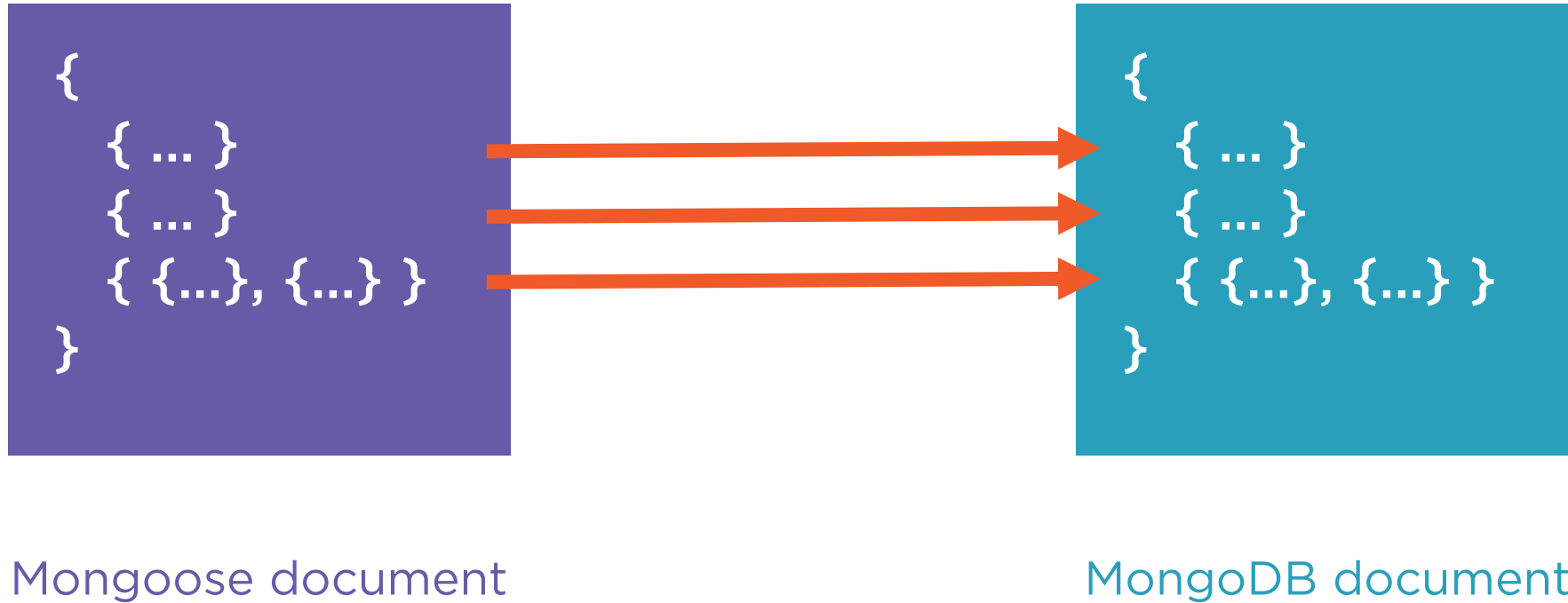
// Expose (export) the model now
module.exports = mongoose.model('Standup', standupSchema )
```



Documents



Documents - Relationship to MongoDB

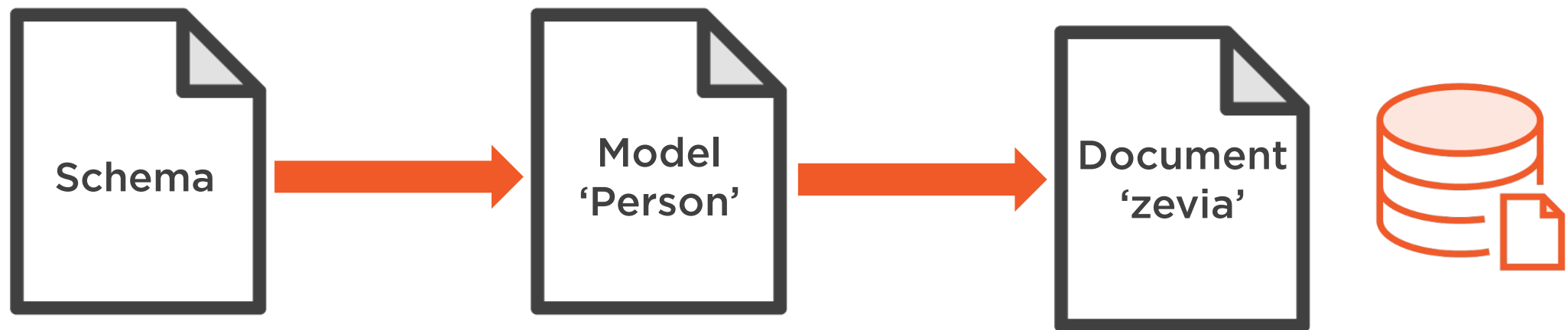


ODM

Object Document Modeling (Mapper)




```
const personSchema = new mongoose.Schema({  
  firstName: String,  
  lastName: String  
});  
  
const Person = mongoose.model('Person', personSchema);  
  
const zevia = new Person({  
  firstName: 'Zevia',  
  lastName = 'Cola'  
});
```



Sub-documents



```
const mongoose = require('mongoose');

// Pre-define sub-documents...
const subCategory = {
  name: String,
  description: String,
  isActive: Boolean };

const subAnswers = {
  answerText: String,
  isCorrect: Boolean,
  displayOrder: Number };

const subQuestions = {
  type: { type: String },
  text: String,
  answers: [ subAnswers ] };

// Define main document schema...
const quizSchema = new mongoose.Schema({
  name: String,
  description: String,
  isActive: Boolean,
  categories: [ subCategory ],
  questions: [ subQuestions ]
});
```

```
name
description
isActive
categories [ ... ]
questions [
  questionType
  questionText
  answers [
    answerText
    isCorrect
    displayOrder
  ]
]
```



```
// Build Quiz model from schema
const Quiz = mongoose.model('Quiz',
quizSchema);

// Document instance of model
let quiz1 = new Quiz({
  name: 'Favorite Things Quiz',
  description: 'Demo Quiz',
  categories: [
    {
      name: 'Favorites',
      description: 'Favorite things
quiz category',
      isActive: true
    },
    { // other categories... }
  ]
});

// Save the document now
quiz1.save(callback);
```

```
name
description
isActive
categories [ ... ]
questions [
  questionType
  questionText
  answers [
    answerText
    isCorrect
    displayOrder
  ]
]
```



```
let categories = [];
// Fill in the categories array
const cat1 = { name: 'Test1', description: 'Test 1 category', isActive: true };
const cat2 = { name: 'Test2', description: 'Test 2 category', isActive: true };

categories.push(cat1, cat2);

let questions = [];
// Add in some questions and answers...
const q1 = { type: 'Multiple-Choice', text: 'What is your favorite color?',
  answers: [
    { answerText: 'Red', isCorrect: false, displayOrder: 1 },
    { answerText: 'White', isCorrect: false, displayOrder: 2 },
    { answerText: 'Blue', isCorrect: true, displayOrder: 3 }
  ] };
const q2 = { type: 'Multiple-Choice', text: 'What is your favorite animal?',
  answers: [
    { answerText: 'Dog', isCorrect: true, displayOrder: 1 },
    { answerText: 'Cat', isCorrect: false, displayOrder: 2 },
    { answerText: 'Squirrel', isCorrect: false, displayOrder: 3 }
  ] };
questions.push(q1, q2);

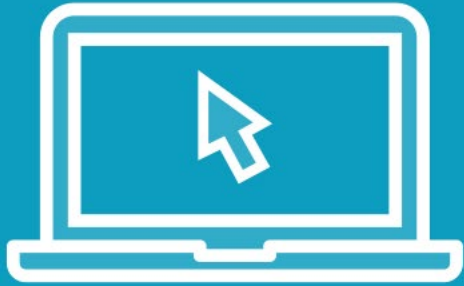
// Create the parent quiz document - supply the categories and questions now...
const quiz2 = new Quiz({
  name: 'Example Quiz',
  description: 'Example Quiz long description...',
  categories: categories,
  questions: questions });
```



placeholder



Demo



Building a Model from a Schema

Documents and Sub-documents

Build the Models

- Standup
- Project
- TeamMember

Save Mongoose documents to MongoDB

- API routes



placeholder

