

---

# Introducción al desarrollo web entorno servidor

Gabriel Rodríguez Flores

February 13, 2024

- Discriminators (Herencia)
- Hooks
- Validaciones

## Contents

<b>1</b>	<b>Teoría</b>	<b>3</b>
1.1	Hooks . . . . .	3
1.1.1	Pre . . . . .	3
1.1.2	Post . . . . .	3
1.2	Tipos de schemas . . . . .	3
<b>2</b>	<b>Ejemplos</b>	<b>4</b>
2.1	Schema types . . . . .	4
<b>3</b>	<b>Ejercicios</b>	<b>4</b>
3.1	Validaciones . . . . .	4
<b>4</b>	<b>Entregables</b>	<b>5</b>
4.1	En clase . . . . .	5
4.2	Tarea . . . . .	5

# 1 Teoría

## 1.1 Hooks

Se pueden definir hooks para ejecutar código antes o después de ciertas acciones.

### Sintaxis

```
schema.HOOK(METHOD, FUNCTION);
```

#### 1.1.1 Pre

```
schema.pre('save', function (next) {  
  // do stuff  
  next();  
});
```

#### 1.1.2 Post

```
schema.post('save', function (doc, next) {  
  // do stuff  
  next();  
});
```

También se pueden definir hooks para varios métodos específicos.

```
schema.post(['findOne', 'findOneAndUpdate'], function (next) {  
  // do stuff  
  next();  
});
```

## 1.2 Tipos de schemas

```
const schema = new Schema({  
  name: String,  
  binary: Buffer,  
  living: Boolean,  
  updated: { type: Date, default: Date.now },  
  age: { type: Number, min: 18, max: 65, required: true },  
  mixed: Schema.Types.Mixed,  
  _someId: Schema.Types.ObjectId,  
  decimal: Schema.Types.Decimal128,  
  array: [],
```

```
ofString: [String],
ofNumber: [Number],
ofDates: [Date],
ofBuffer: [Buffer],
ofBoolean: [Boolean],
ofMixed: [Schema.Types.Mixed],
ofObjectId: [Schema.Types.ObjectId],
ofArrays: [[]],
ofArrayOfNumbers: [[Number]],
nested: {
  stuff: { type: String, lowercase: true, trim: true },
},
map: Map,
mapOfString: {
  type: Map,
  of: String,
},
});
```

## 2 Ejemplos

### 2.1 Schema types

```
const schema = new Schema({
  name: { type: String, required: true },
  age: Number,
  stories: [{ type: Schema.Types.ObjectId, ref: 'Story' }],
  nickname: { type: String, required: true, unique: true },
  date: { type: Date, default: Date.now },
  meta: {
    votes: Number,
    favs: Number,
  },
});
```

## 3 Ejercicios

### 3.1 Validaciones

- Crear una validación en el modelo de datos para comprobar que el valor de un campo pertenece a la lista de otro campo.

Ejemplo válido

```
{  
  "selectedColor": "blue",  
  "colorList": ["red", "yellow", "blue", "green"],  
}
```

Ejemplo NO válido

```
{  
  "selectedColor": "gray",  
  "colorList": ["red", "yellow", "blue", "green"],  
}
```

## 4 Entregables

### 4.1 En clase

### 4.2 Tarea