

- [Example:](#)
- [Template: <Name> | <Type> | *<Modifiers>](#)
- [Here's a breakdown of how its used:](#)

Mongeeasy aims to simplify the process of creating Mongoose schemas using a straightforward syntax for defining the schema structure. It employs a flag-based declaration paradigm known as CSS (Carnage Schema Syntax). The syntax consists of the following elements:

Example:

```
import {model} from mongeeasy;
let Computer = model({
  Computer : [
    "name|String|!|*(3,16)",
    "price|Number|*",
    "madeIn=India|String|ref:CountryNames",
    "dateOfPurchase|Date|
  })
```

Template: <Name> | <Type> | * <Modifiers>

Name:

- the name of field, 😊😓😭 - emojis also supported

Type :

all supported, as we directly pass to Core Library

- [Array, Boolean, BigInt, Buffer, Date, Decimal, Decimal128, DocumentArray, Map, Mixed, Number, ObjectId, String, Subdocument, UUID, Oid, Object, Bool, ObjectId]

Modifiers:

"/" means "or"

- +U / ! = unique
- +R / * = required
- +I / # = index
- (0, 999) = min,max
- ['adam', 'eve'] = enum

Here's a breakdown of how its used:

- **Name** : This part represents the name of the field you want to define in your schema. It's a required element.
- **Type** : This part specifies the type of the field, and it's also a required element. You can use any of the supported types provided by Mongoose.
- **Modifiers** : These are optional and allow you to add extra attributes or flags to the field. For example, you can use **!** for unique fields, ***** for required fields, and **#** for indexed fields.

This CSS-based declaration paradigm aims to provide a simplified way of defining Mongoose schemas, making it easier to work with MongoDB data models.