



# Discriminators

## The `model.discriminator()` function

Discriminators are a schema inheritance mechanism. They enable you to have multiple models with overlapping schemas on top of the same underlying MongoDB collection.

Suppose you wanted to track different types of events in a single collection. Every event will have a timestamp, but events that represent clicked links should have a URL. You can achieve this using the `model.discriminator()` function. This function takes 3 parameters, a model name, a discriminator schema and an optional key (defaults to the model name). It returns a model whose schema is the union of the base schema and the discriminator schema.

```
const options = { discriminatorKey: 'kind' };

const eventSchema = new mongoose.Schema({ time: Date }, options);
const Event = mongoose.model('Event', eventSchema);

// ClickedLinkEvent is a special type of Event that has
// a URL.
const ClickedLinkEvent = Event.discriminator('ClickedLink',
  new mongoose.Schema({ url: String }, options));

// When you create a generic event, it can't have a URL field...
const genericEvent = new Event({ time: Date.now(), url: 'google.com' });
assert.ok(!genericEvent.url);

// But a ClickedLinkEvent can
const clickedEvent = new ClickedLinkEvent({ time: Date.now(), url: 'google.com' });
assert.ok(clickedEvent.url);
```

## Discriminators save to the Event model's collection

Suppose you created another discriminator to track events where a new user registered. These `SignedUpEvent` instances will be stored in the same collection as generic events and `ClickedLinkEvent` instances.

```
const event1 = new Event({ time: Date.now() });
const event2 = new ClickedLinkEvent({ time: Date.now(), url: 'google.com' });
const event3 = new SignedUpEvent({ time: Date.now(), user: 'testuser' });

await Promise.all([event1.save(), event2.save(), event3.save()]);
const count = await Event.countDocuments();
assert.equal(count, 3);
```

## Discriminator keys

The way mongoose tells the difference between the different discriminator models is by the 'discriminator key', which is `__t` by default. Mongoose adds a String path called `__t` to your schemas that it uses to track which discriminator this document is an instance of.

```
const event1 = new Event({ time: Date.now() });
const event2 = new ClickedLinkEvent({ time: Date.now(), url: 'google.com' });
const event3 = new SignedUpEvent({ time: Date.now(), user: 'testuser' });

assert.ok(!event1.__t);
assert.equal(event2.__t, 'ClickedLink');
assert.equal(event3.__t, 'SignedUp');
```