Sequences

Now we can move on to sequences. Once again, the idea is this:

- We have a sequence that contains multiple e-mails.
- Each of those e-mails has some scheduling logic.

We need to handle two different time units when scheduling sequence e-mails:

- Hours
- Days

So users can create schedules such as:

- Five days after the last e-mail.
- Two hours after the previous e-mail.

They also want to specify which days the given e-mail can be sent:

- Five days after the last e-mail, but only on Fridays.
- Two hours after the last e-mail on any day.

Before modeling the tables, let's discuss how to store these schedules:

delay	unit	allowed_days
5	days	{"monday": false, "tuesday": false, "wednesday": true, "thursday": false, "friday": false, "saturday": false, "sunday": false}
2	hours	{"monday": true, "tuesday": true, "wednesday": true, "thursday": true, "friday": true, "saturday": false, "sunday": false}

The attributes are the following:

- `delay` stores the number of days/hours we need to wait after the previous e-mail.
- `unit` is either days or hours. This column can be omitted if we store every delay in hours. In this case, five becomes 120. This can work, but we don't have to convert hours from days in the code if this table has a unit column. It's easier, in my opinion.