

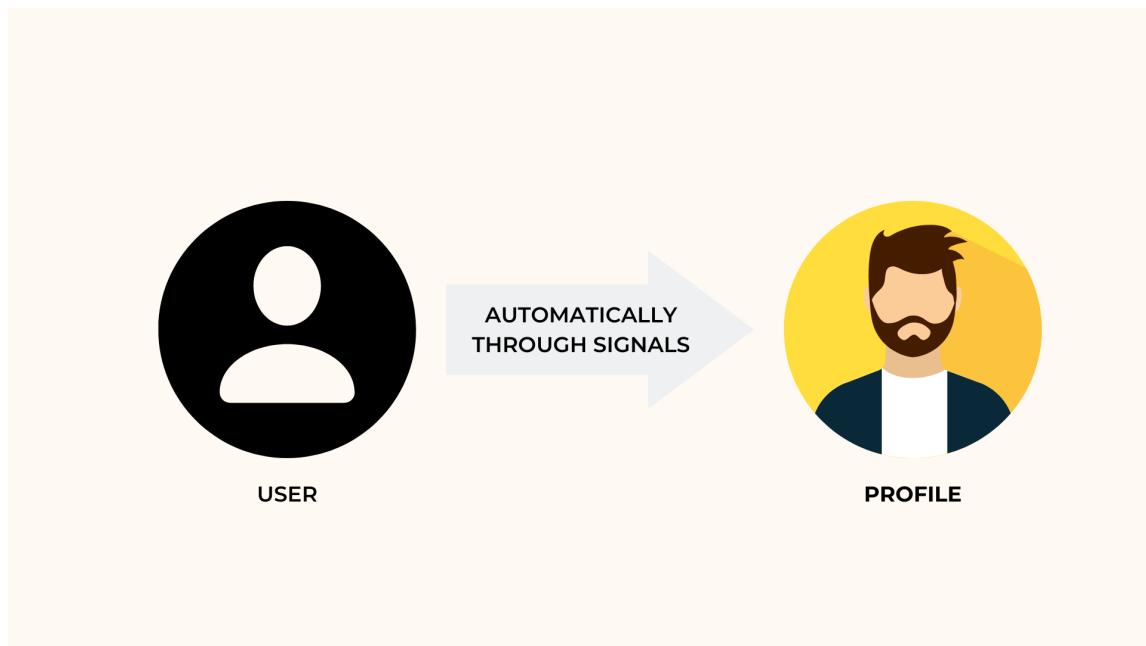


Follower Mechanism for Social Media Website

Answers to my why's?

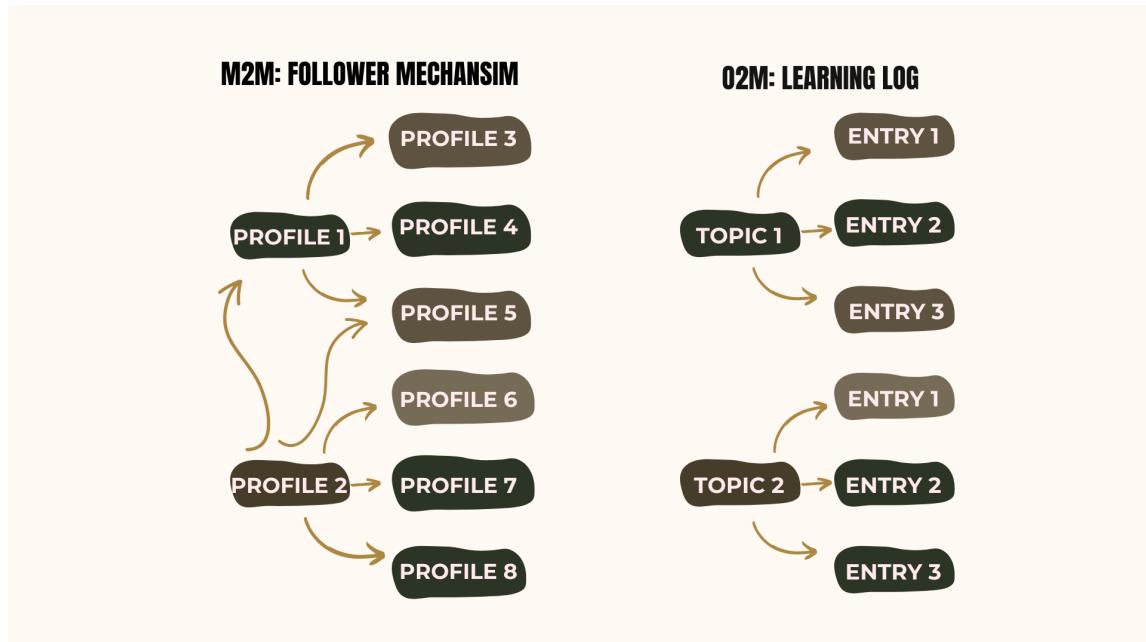
- Why do we use signals ?

To automatically create a user's profile as soon as a new user is signed into the database. Without using signals, we've to manually create a user's profile in the Django admin panel.



- Why is M2M field used to create the follower's and following's list, not O2M?

Because a user profile can follow multiple profiles and can be followed by multiple user profiles simultaneously.



- What is self as an argument in the following instance while defining M2M relationship?

It is used to define M2M relationship with self (Profile model).

Basic guide to signal integration

- Django **Signals** allow certain senders to notify a set of receivers that some action has taken place.
- A **receiver** can be any Python function or method to receive signals.
- register a signal

```
from django.db.models.signals import post_save

def signal_name(sender, instance, created, **kwargs):
    # my code

post_save.connect(signal_name, sender=Mymodel)

or using @receiver decorator

from django.db.models.signals import post_save
```

```
from django.dispatch import receiver

@receiver(post_save, sender=Mymodel)
def signal_name(sender, instance, created, **kwargs):
    # my code
```

- Where to write a signal?

```
my_app/signals.py

def my_signal_name(sender, instance, created, **kwargs):
    # my code
```

- Wrapping up steps

```
my_app/apps.py

from django.apps import AppConfig

class AppConfig(AppConfig):
    default_auto_field = 'django.db.models.BigAutoField'
    name = 'my_app'

    def ready(self):
        import my_app.signals
```

```
my_app/__init__.py

django_app_config = 'my_app.apps.MyAppConfig'
```