The ultimate goal is to show that code is fun. **To get people excited.** To show them that programming is not scary and that it is for everyone. To show them how powerful programming skills can be.

* Admit that you don't know everything
* Tell them it's ok to make lots of mistakes
* Tell them it's ok to get frustrated

**Encourage them to make their own changes and not to follow the tutorial too precisely** -

Models:

A model is the single, definitive source of information about your data. It contains the essential fields and behaviors of the data you’re storing. Generally, each model maps to a single database table.

* Each model is a Python class that subclasses **[django.db.models.Model](https://docs.djangoproject.com/en/2.1/ref/models/instances/" \l "django.db.models.Model" \o "django.db.models.Model)**.
* Each attribute of the model represents a database field.
* With all of this, Django gives you an automatically-generated database-access API; see [Making queries](https://docs.djangoproject.com/en/2.1/topics/db/queries/).

A model class represents a database table, and an instance of that class represents a particular record in the database table.

To create an object, instantiate it using keyword arguments to the model class, then call [**save()**](https://docs.djangoproject.com/en/2.1/ref/models/instances/#django.db.models.Model.save) to save it to the database.

This performs an **INSERT** SQL statement behind the scenes. Django doesn’t hit the database until you explicitly call [**save()**](https://docs.djangoproject.com/en/2.1/ref/models/instances/#django.db.models.Model.save).

Migrations:

Migrations are Django’s way of propagating changes you make to your models (adding a field, deleting a model, etc.) into your database schema.

You should think of migrations as a version control system for your database schema. **makemigrations** is responsible for packaging up your model changes into individual migration files - analogous to commits - and **migrate** is responsible for applying those to your database.