* Django project URLS > app URLS > app VIEWS > render index.html
* App URLS > app VIEWS
* MODELS.PY file is used for communication with database server
  + Rum migrations after editing MODELS file
* In urlpatterns
  + “ ” is the default page
  + “about” is the name of the webpage
* Makes project dev more feasible
  + Instead of changing the everything you can change the app URLS link for ex blog-dev
* To make a page the home page of the website
  + Go to project urls and set the path to “ “
* Links to pages are in APP VIEWS
* Project folder does not contain a views folder only the URLS file
* App > Templates > App > Template.html
* Take the name of the app from the apps APPS.PY file and place it into the projects settings file
  + APPS.PY = “class BlogConfig(Appconfig)”
  + SETTINGS.PY = ‘blog.apps.BlogConfig’
* In app VIEWS
  + Return render(request, ‘blog/home.html)
* Update VIEWS to add new webpage
* Data is passed from app VIEWS, render(request, ‘blog/home.html, data)
* Inheritance
  + Grab repeating code from each file
  + A block is a section that child templates can override
  + To extend means to use the parent file as the template
* ORM
  + Object relational mapper
  + Allows us to access our database
  + Represent database structure as classes and are called Django models
* Django app migrations can be found at
  + Django\_project > APP > migrations > 0001\_initial.py
* MODELS.PY
  + Sql syntax will appear after running the **sqlmigrate** command
* Migrations
  + Allow us to update our database even if they have existing information
* Look into python double underscore methods, special methods, magic methods
* HTML/Django date formats in video #5 @ 32min
* SETTINGS.PY
  + Name of installed\_apps will be found on the apps APP.PY
* CSRF Token
  + Cross site request forgery
  + Protects from certain attacks
  + Vid 6
* Video 3
  + Pick fields from passed information from views to template
    - 13:40
* Video 5
  + Post Models
  + Database
  + Models to sql code
    - 9:50
  + Django shell
    - 13:00
  + Queries
    - 24:00
  + Creating a post with the user
    - 26:00
  + Django Dates
    - 32:00
* Video 6
  + User registration page
    - 3:00
  + Adding a new form field
    - 30:00
  + Flash messages
  + User registration and authentication
  + Messages
    - 21:00
  + Making the form
  + Saving the form
    - 30:51
  + Creating your own form
    - 36:16
  + User forms
    - 33:00
  + Crispy forms
    - 39:50
    - Pip3 install django-crispy-forms
* Video 7
  + Login page and redirect setups
  + Dynamic user display. Changing nav tab to say logout when logged
    - 18:00
  + Allowing media to work
    - 21:00
  + Making the user profile page
    - 25:00
* Video 8
  + Dealing with Django images
  + Models.OneToOneField. CASCADE
    - 2:40
  + Registering a model to the admin page
    - 7:30
  + Django shell profile and picture navigation/commands
    - 9:00
  + Settings change. Media route and media url
    - 14:00
  + Files are stored on the file system and not the database for performance reasons
  + BASE\_DIR
    - 15:30
  + Profile page setup and image display
    - 20:00
    - User.profile.image.url
  + Adding media routes to url patterns
    - 21:00
  + Profile picture
    - 25:00
  + Django signal
    - 26:50
    - Associating a user with a profile
* Video 9
  + Working with forms
  + Letting users update their information
  + From FORMS.PY to VIEWS.PY
  + Importing forms into views
    - 3:00
  + Updating pages to display forms
    - 5:00
  + Combining two forms
    - 6:00
  + Updating an image in html
    - 7:00
  + Placing base data in the forms. Populating forms. Allowing them to take affect
    - 9:00
  + Updating posts
    - 11:00
  + Updating image/file
    - 12:00
  + Checking if form is valid
    - 12:35
  + Quick message update
    - 13:20
  + Updating existing and not creating new objects
    - 15:00
  + Reformatting image size
    - 17:00
* Video 9
  + Adding the functionality to add posts
  + Ordering posts
    - 9:00
  + Using Listviews and Detailviews for posts
    - After 10:00
* Video 10
  + Views
  + Giving each post an individual view aka detailed view
    - 10:00
  + Create views
  + Ordering posts
    - 8:00
* Video 11
  + Shortening the amount of content the webpage will load
    - 4:20
  + Display only post from user who made it
    - 22:50
  + Each post author will have a dedicated page of their own posts
    - 20:00