Create Mix Project

Notes

Running Elixir

Here's a quick recap of the commands we used in the video to run an Elixir file:

1. Run the `elixir` command with the relative path of the Elixir file:

   ```
   $ elixir lib/servy.ex
   ```

   The file gets compiled into bytecode (in memory) and then run on an Erlang virtual machine.

2. Fire up an `iex` (Interactive Elixir) session and then use the `c` helper function to compile and run the file:

   ```
   $ iex
   ```

   ```
   iex> c "lib/servy.ex"
   ```

   The `c` helper function compiles the given file in memory, the module (`Servy` in this case) is loaded into the session, and any code outside of the module is interpreted.

   To exit the `iex` session, press `Ctrl+C` twice.

3. Alternatively, you can tell `iex` to interpret an Elixir file while starting by passing the relative path of the file:

   ```
   $ iex lib/servy.ex
   ```

4. When you start a standard `iex` session, it doesn't know about the paths and dependencies of a `mix` project. So to start a session in the context of a project, you need to pass the `-S mix` option:

   ```
   $ iex -S mix
   ```

5. Finally, to recompile a module while in `iex`, use the `r` helper function:

   ```
   iex> r Servy
   ```

   This recompiles and reloads the given module, which is `Servy` in this case.

Exercise: Explore Helper Functions

Start an `iex` session and then type `h` (followed by Return) to see all the helper functions at your disposal. You might also want to spend a few minutes reviewing the online documentation for the `IEx` module.

Exercise: Enable Shell History

The `iex` shell can keep a history of expressions you've run so you can use up/down arrow keys to traverse the history. We love this feature! Oddly, shell history is disabled by default. To enable shell history, add the following line to your `.zshrc`, `.bashrc` or the like:

```
export ERL_AFLAGS="-kernel shell_history enabled"
```

Once you've changed the appropriate file, make sure to either open a new command-line window or `source` your `.zshrc` or `.bashrc`. Then fire up an `iex` session and you should have shell history.

Running Elixir with Sublime

If you're using Sublime Text with the elixir-tmbundle plugin as we are in the videos, you can run an Elixir file by first going to the menu item *Tools -> Build System* and selecting *elixir*. Then hit `Cmd-B` to run the file and see the build output as shown in the video. Hit `Esc` to close the build output pane.

Code So Far

The code for this video is in the `mix-project` directory found within the `video-code` directory of the code bundle.

Go To Next Video