

# HTML EXAMPLES DOC

Hello All and welcome to the world of HTML! To follow along on this experimental path, you will need to download the [d20pro-example-html.zip](#) archive and decompress the contained 'html' folder to the root of your D20PRO installation.

Please note, the instructions contained herein are intended for use with the [3.7.3.1 Beta build of D20PRO](#).

Any resemblance to past or future builds is purely coincidental and is not intended to imply functions beyond the scope of those builds.

## URL BASED TOOLS AND INTERFACES

Okay, so you've been warned and you're still here. Great! Lets get you set up and started with the web content integration(ing?!)

Now that we've got the zip archive and 3.7.3.1 installed, lets unzip the archive into the root of your D20PRO directory. When you're done, you should have something along the lines of...

### DIRECTORY STRUCTURE

Please place the **html** directory in the **ROOT** of your D20PRO install directory.

- /html/CustomSheet - Contains a sample 5e character sheet.
- /html/deck-of-cards - Extensions example allowing for basic manipulations of a deck of cards.
- /html/dice - 3d Dice roller for using with the Custom Roller options.
- /html/terminal - Extensions tool for debugging and investigating the Javascript<->Java bridge.

The archive contains functional samples for a **Custom Character Sheet** and a **3D Dice Roller**. The archive also contains samples for a **Deck of Cards** as well as a **Terminal Tool** for help in debugging and building additional web tools.

## CUSTOM SHEETS & ROLLER

To use a **Custom Sheet** in D20PRO 3.7.3.1, you'll want to launch your campaign then open up the Options panel. On the Rules tab, you'll see two options; **Custom Sheet URL** and **Custom Roller URL**.

Using these options you can inject URLs which will function as the calls for the creature view editor (Custom Sheets) and the Dice Box.

## CUSTOM SHEETS DETAILS

To simplify the Custom Sheets example, we've stashed the javascript logic in the, aptly named, *logic.js* file. It can be found in **html/CustomSheet/js/logic.js**.

Creature Template access is available in javascript using the notation: **window.template**. You can access all class members of the Creature Template class through this interface. For example: **window.template.getName()** will result in the creature name being returned to javascript.

The Creature Template documentation is available [here](#).

## CUSTOM ROLLER DETAILS

The Custom Roller utilizes THREE.js and a few other cool bits and bobs. It's based on the work found at [teall.info](#). You'll find that the D20PRO-centric changes are specific to pushing roll results to the gamelog using **window.gamelog**. I'll continue updating the app to do other useful dice related functions, but for now it's relatively simplistic in it's functions.

Multi-user and sheet-attached rollers will come into the mix as well as development work continues!

## EXTENSIONS DETAILS

The Extensions system has access to the following classes using the following javascript calls:

- **window.template** - [Creature Template Class Docs](#)
- **window.gamelog** - Not yet available as a javadoc.
- [Additional Classes Details](#)

