Step By Step Tutorial

A Tutorial for New York Noir

by Daniel Turgeon

v4.0 - 10/13/26

SUMMARY

A step-by-step tutorial that will gently teach you how to create a case for New York Noir.

- **Author**: Daniel Turgeon
- **Status**: Complete
- **Additional credits**: Special thanks to Jesse Reichler for helping write, contributing, and reviewing this tutorial; and Debbie Levy for editing this document.
- **Game system**: nyNoir
- **Case date**: 6/20/1915
- **Difficulty**: 1 out of 5
- **Playtime**: 2 hours
- **Compiled**: Tue, Oct 13, 2026 at 07:44 PM CDT / Casebook v6.97 (10/10/26) / LeadDb v6
- **Typesetting**: 10pt letter twosided (xelatex)
- **Stats**: 11 Leads / 3 Docs / 1 Marker / 19 Images / 1 Day / 12.5k Words (1165 avg.) / 107.7 KB

This page intentionally left blank.

Table of Contents

Front Matter 1
- Cover Page . . . 1
- Table of Contents . . . 3

How To Use This Document 5
- Prerequisites . . . 6
- Writing Workflow . . . 7
- Casebook General Overview . . . 8

Getting Started 9
- Regenerate the PDF . . . 12

Document Structure 13

Configuring Your Game 15

Creating The Front Matter (Cover Page, Instructions, Etc.) 21

Creating Your First Day 23

Formatting Text 25
- Markdown . . . 26
- White space (tabs, spaces and blank lines) . . . 28

Functions 29

Creating Leads 31
- Leads from the Directories . . . 32
- Setting Lead Time (Duration) . . . 33
- Autogenerated Lead Numbers . . . 34
- Inline Leads . . . 34
- Finding Leads For Your Case . . . 36

Using Tags 37

Gaining Documents 39

Working with Events 41

Day One 43
- Introduction . . . 43
- Evening of Day 1 . . . 45

LEADS 47
- 1 . . . 48
  - 1-9896 . . . 48
- 3 . . . 49
  - 3-6077 . . . 49
- 4 . . . 50
  - 4-3335 . . . 50
- 5 . . . 51
  - 5-5104 . . . 51
  - 5-6715 . . . 51
  - 5-6952 . . . 51
- 6 . . . 52
  - 6-3677 . . . 52
  - 6-7138 . . . 52
- 7 . . . 53
  - 7-3807 . . . 53
  - 7-5997 . . . 53
- 8 . . . 54
  - 8-4077 . . . 54

DOCUMENTS 55
- Document 1 . . . 57
- Document 2 . . . 59
- Document 3 . . . 61

END 63
- Conclusion . . . 63
- Questions . . . 64
- Answers . . . 65
- Epilogue - My first week in NYPD . . . 66
- Postscript: Behind the Scenes . . . 67
- Full Walkthrough . . . 68

HINTS 69
- Hint for Marker A1 . . . 71

What's Next 73
- Publish Your Game . . . 74
- Beyond the Basics . . . 75

This page intentionally left blank.

How To Use This Document

You've decided to write a case for New York Noir. Congratulations!

In this tutorial, I will try to gently guide you through your journey.

> **NOTE**: This tutorial will not teach you how to tell a **good** story, or even how to make an **interesting** game. This tutorial just focuses on the technical side of creating a case.

This document is meant to be read primarily from start to finish. Only the **LEADS** section should not be read in a consecutive manner.

> **IMPORTANT**: I will frequently refer to a given section or ask you to look at a specific page. In this case you should go to the indicated section, and then return to wherever you were previously.

You will be taught how to build a very simple **New York Noir** investigation case, and you will learn the basic rules and tools to achieve this goal.

At the end, you will have a story case and a PDF document to play along with.

I will teach you how to start from a blank template and build the document as you are learning.

> **NOTE**: You can refer to the
>
> https://nynoir.org/case/simple-tutorial-case/
>
> This demo case includes a link to its accompanying Casebook source code. At the end of this tutorial, you will have written the same text. Feel free to have a look and refer to it if needed.

> **HINT**: I suggest enabling the PDF's *Table of Contents* sidebar. You can see the document structure and easily jump back when needed.

Prerequisites

To use this guide, you should be somewhat familiar with the nature of a New York Noir game or have a general knowledge of the **Sherlock Holmes Consulting Detective** style of game.

I suggest you play the introductory game **The Past is a Crooked Game**.

Have a look at these documents to familiarize yourself with the game's rules:

- New York Noir: Quick Start Rules
- New York Noir: Navigation Guidebook

All the documents above are available here:

- https://nynoir.org/downloads
- https://nynoir.org/cases

IMPORTANT: To follow this tutorial and create your first case, you will need to create an *Author* account on the New York Noir website. Start by signing up at nynoir.org to create your account, then visit the "Authoring" tab and submit a request to have your account upgraded. There's no charge for this (or any other feature on nynoir.org) and your request should be granted within 24 hours. Should you need any assistance with any of the steps in this tutorial, you can stop by our Discord at `https://discord.gg/GhBmg5JFPc` or email jessereichler@gmail.com.

Writing Workflow

First, let me explain the basic process of writing a case for New York Noir:

Cases are written in plain text, using a language named *Casebook*. It helps handle all the little details and lets you do the most important thing: **Writing your story**.

In fact, the Casebook language and nynoir.org tools were actually used to create the pdf document you are reading right now!

> **NOTE**: You won't actually be writing your case *on* the nynoir.org website. Rather, you will use the nynoir.org website to "compile" your text into a pdf.
>
> To actually write the text of your case, you can use something like **Google Docs**, which makes it easy to write text and also keep track of revisions, collaboratively edit documents, etc. Alternatively you can use a text editor of your own choice, but the nynoir.org website knows how to automatically fetch your latest text from Google Docs, which is very convenient. If you use a different text editor, you'll have to copy and paste your latest changes manually each time you want to rebuild your pdf.

For the remainder of this tutorial, to make a distinction between the **generated** text and the tutorial guide, I will put the guidance in a colored box for the walkthrough:

> Here's an info text.

```
Here's Casebook language and text.
```

> Here's the generated text.

> **NOTE**: Here's an advanced hint text.

The tutorial walkthrough will be plain text. No box, no color.

Casebook General Overview

In the following pages, I will walk you through writing a simple case while teaching you the *Casebook* language.

With Casebook you can generate a lot of text and graphical images with a simple line.

For example, I can insert the date into a text by doing this:

The *Casebook* syntax:

```
The current date is $date("6/9/1915", year=false).
```

The generated text:

> The current date is Wednesday, June 9th.

Simple indeed!

See how I present the Casebook code and the generated text in boxes. I will reuse this approach in the following pages.

Getting Started

This section will give guidance on how to extract the blank template and continue through to build your first PDF. I will also provide a reminder section on how to rebuild your PDF when you have made some changes.

Let's start your new case by making a copy of a "minimal" starting template for writing new cases.

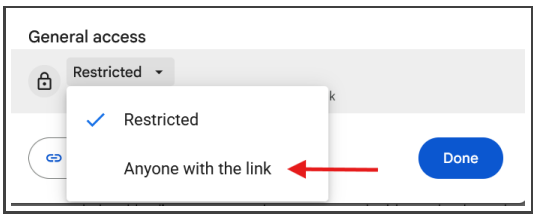
> **NOTE**: The "minimal" starting document that you are making a copy of is is **not** blank. It contains a significant amount of boiler-plate text that I will describe in subsequent topics. This boiler-plate text helps you get started more quickly.

Start by opening the Google Doc at

> https://docs.google.com/document/d/1am1HNjGZhSgjJKIvRkTmadlxXIiuXZxycIfxuhEV-lo/edit?usp=sharing

and make your own copy by accessing the menu through **File** > **Make a Copy**. Name it whatever you want, and save it in your **Google Drive folder** (you'll need a Google Drive account for this tutorial).

Share the document by clicking on the **Share** button in the upper right of the screen. Then change the *General access* from **Restricted** to **Anyone with the link** (this just means that if you share the url of the document with someone, they will be able to read it):



Click on the **Copy link** button so the **link path** is now stored in the clipboard. That **link path** will be useful later.

This minimal template that you've copied will act as a starting point for your case, and you should be able to create a pdf from it without making any changes.

Let's try to generate a pdf from your new copy now. Go to

https://nynoir.org/cases/new/

CONTINUED ON NEXT PAGE

REMEMBER: If you can't access that page, it means you don't have an Author account yet. You can request one and it's free. Go to Prerequisites (p.6) for all details.

First, fill the **short name** in the field with whatever you want.

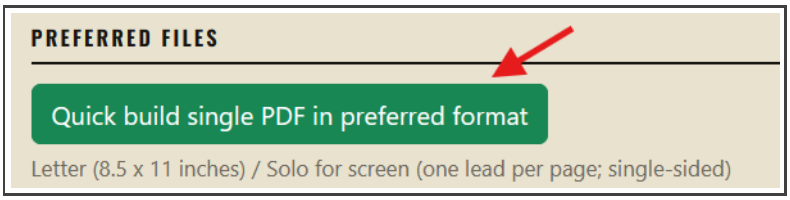
Then go down and look for the **Fetch and Update** button. Paste the **link** to your Google Doc in the box beside it, and click the button to automatically fetch your Google Doc (alternatively you could just paste your game text into the text box below).

NOTE: You can always come back here to retrieve you document's url.

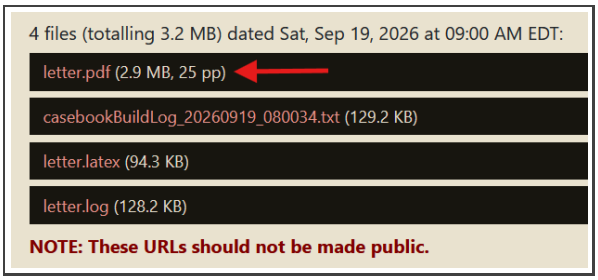
Then click **Save** and your first case is saved.

Now, you will have to **generate your first pdf**. Just click on the *Generated PDFs* in the upper right corner:

Then push the green **build** button and wait.



After a short while, you will have a generated pdf.



You can ignore the other three files that were generated – these are only useful for diagnosing problems.

CONTINUED ON NEXT PAGE

NOTE: The instructions above tell you how to generate a *preferred* build of your case, which just means that you are building a single pdf in your preferred format (single or double columns) for yourself. That's what you'll be generating most of the time – it's a pdf just for you. Your case will not be made public until you eventually "publish" it publicly. I will instruct you on how to do this in one of the last sections: "Publish Your Game".

Regenerate the PDF

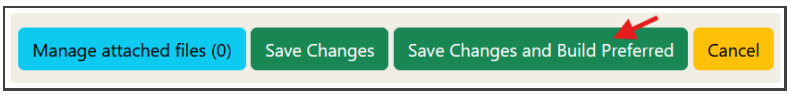
You will often edit your document and make a lot of changes while writing the case. Assuming you are ready to get a new version of the pdf, just follow the steps below.

Regenerate

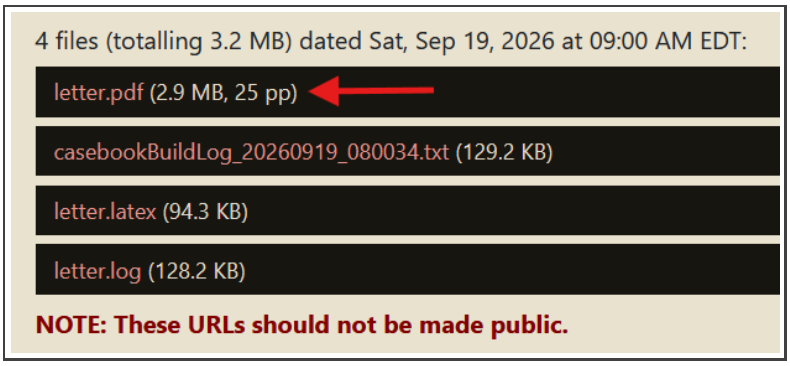
After editing your Google Doc, go back to your game on nynoir.org (you can find your cases in your profile or on the Cases page). Go to your game page and click **edit**:

Then click the **Fetch and Update** button to pull in your latest changes from Google Docs.

Followed by the **Save Changes and Build Preferred** button:

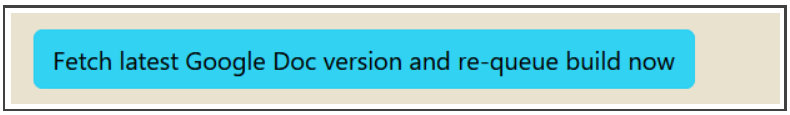


After a short while, you will have a new generated pdf. *Voilà!*



In one click

You can do all the previous steps in one click from the **Generated PDFs** page of a game by clicking on the button **Fetch latest Google Doc version and re-queue build now**:



Document Structure

Before we dive into writing the text of our case, we need to understand a bit more about the structure of a Casebook document. Let's start with an overview of the document:

Lines that start with one or more # characters are **section headers**. Multiple # characters allow us to nest subsections within parent sections. In this way, your entire document is organized into a hierarchy of sections and subsections.

If we just look at the lines starting with a # character, we can get an idea of the overall structure of the document:

```
# OPTIONS
…
# SETUP
…
# FRONT "Front Matter"
…
# REPORT "Author Report"
…
# D1 "Day One"
…
# LEADS
…
# DOCUMENTS
…
# END
…
# HINTS
…
```

I've removed all the text inside each section so you can get a feel for the overall structure of your case document. Within each of those main sections, there are child subsections (starting with two ##) and sometimes even sub-subsections (###).

Most of the top-level sections (starting with a single #) have special uses which I will talk about later in this document.

For now, if you glance through the case document that you have just started writing, you will see a lot of plain text that is just part of your story – but also many things you don't understand yet, most of which are function calls starting with a "$" character. Let's ignore all that for now.

CONTINUED ON NEXT PAGE

NOTE: Section headers that start with # characters always have an ID (the first bit of text that follows the # characters).

Think of the ID as the internal name of the section. It can't have any spaces in it. The ID is how you refer to it if you want to send the player to that section. Often you will want to specify an alternate, more friendly **label** for the section that is displayed on the page in place of ID (for example, if you want to display a header with multiple words).

To specify an alternate label for the section, just put it in double quotes after the ID. If no label is specified, the ID is displayed. If you *only* specify a label in double quotes, it will use that as both ID and label.

TIP: Everything in Casebook, including section ID names, tags, and function names, are all case sensitive!

In the next topic we will look at the first two special sections, # **OPTIONS** and # **SETUP**.

Configuring Your Game

The configuration of your game details at the start of your document can seem overwhelming. There are a **lot** of scary-looking lines that start out your document.

The good news is that most of those lines can be left alone, and most cases have the same basic initial configuration. In any case, you won't ever be writing these lines from scratch – you'll basically start with a minimal blank starting case and just tweak some values to configure certain properties of your game (such as the title of the case, etc.).

So, starting from the minimal Google Doc that is your case, let's take a look at the intimidating # **OPTIONS** and # **SETUP** sections in a little more detail, and try to demystify them a little bit.

In the # **OPTIONS** section, you'll want to make some changes to match the details of the game you are creating.

In the # **SETUP** section, you'll want to add and customize your case. You will be editing this section while writing your case.

 # OPTIONS

Think of the # **OPTIONS** section as the place where you specify information about your game that is shown on the nynoir.org website and on the cover page of your game, as well as some key features of your case.

Let's start with the `$configureGameInfo` function call:

```
$configureGameInfo(name="ShortName", title="Long title", subtitle="A Case for New York
Noir", authors="Author Name <author email>", version="3.1", versionDate="6/23/25",
status="Draft", difficulty=1, duration=1)
```

Go ahead and change the **name**, **title** and **authors** string values in your Google Doc to replace them with your own.

CONTINUED ON NEXT PAGE

When you are ready, you might change a few more values in the `configureGameInfo` function:

- **duration** specifies the expected playtime of your case, in hours (a short case would typically be 1 hour, but longer cases can easily reach 5 or more hours).
- **difficulty** typically ranges from 1 (easiest) to 5 (hardest).
- **status** lets players know whether the game is playable yet. Leave this as "draft" for now, and then transition to "beta" when you are beta testing, or "playable" when the game is playable by everyone.
- **version** and **versionDate** should be used by you to help track changes to the case. This isn't so important in the early days, but will become more important as your case gets further revisions. It's also a good sanity check to help you be sure you are viewing the latest version of your case. I strongly encourage you to keep these values updated, and furthermore to make use of a Version History section in the # **Author Report** section of the document, which will be described later.
- For a full list of options see https://nynoir.org/authoring/casebookReference#configureGameInfo.

Now let's look at the `$configureFeatures` function call:

```
$configureFeatures(documents=true, newspapers=true, days=1)
```

The main purpose of the `$configureFeatures` function is to tell the website what features your game supports, so that the https://nynoir.org/cases/ website shows them as icons next to your game and lets players filter by them. The other thing that you specify is the number of days in your case, which is also listed on the website. For a full list of features you can configure, you can see https://nynoir.org/authoring/casebookReference#configureFeatures.

The `$configureGamePage` function call is mostly used to define the **thumbnail** image appearing on the Case listing web page https://nynoir.org/cases. It can also be used to set additional information and download instructions that will appear on the case's website.

```
$configureGamePage(info="Case build using the Step By Step Tutorial.",
thumbnail="portraits/20240324_174230.jpg")
```

The `$configureGameSummary` function call is quite simple: you can define a longer description for your game. This will be shown on the website, but also on the cover page of your game. Think of it as an advertisement for your case.

CONTINUED ON NEXT PAGE

```
$configureGameSummary("This is a longer description of my case.")
```

The `$configureGameInfoExtra` function is useful to set the date of your game and other specific info. It will be shown on the cover page and on the game's webpage. Go ahead and change **gameDate**, which should match the date of your first day in the section # **D1**.

```
$configureGameInfoExtra(cautions="", url="", extraCredits="", gameDate="6/20/1920",
gameSystem="nyNoir", copyright="")
```

Other interesting parameters that you might like to change at some point in time:

- **gameDate** is the date in a string in the mm/dd/yyyy format for the day you are configuring.
- **gameSystem** is the type of game, which is nyNoir by default. You can create other styles of game, but that's outside the scope of this tutorial, so we will stick to nyNoir.
- **extraCredits** is extra information that will be shown on the cover page and in the game listing. If you want to add thanks or collaborator names, this is where it should be done.
- **cautions** is extra information that will be shown on the cover page. If you want to warn players that the case has adult language or add other content warnings, this is the place.
- **url** is empty by default, but you can include a link to a website url for players to learn more. This will be shown on the cover page.
- **copyright** is empty by default, but you can add a Copyright line.

Now there are a couple more important options to configure.

Let's look at `$configureClock(clockMode=true)`. It's an on/off switch that determines if duration times are added to leads by default. In other words, this switch tells Casebook whether your case is one in which players keep track of what time it is, and how time passes.

The classic Sherlock Holmes Consulting Detective cases did not involve tracking time, but most New York Noir cases do, so we will leave the **clockMode** setting at true for our tutorial case. I will revisit adjusting lead times in the section on "**Setting Lead Time (Duration)**".

Next you will see a few more advanced configuration calls in your document; these can all be left as is, and won't need to be modified for the purposes of this tutorial. But to satisfy your curiosity, let's take a look at some of them.

CONTINUED ON NEXT PAGE

```
// version of the New York Noir database your case uses; normally use the latest (v6)
$configureLeadDb(version="v6", versionPrevious="v6")

// Options that control how marker tags are named.
$configureTags(alwaysNumber=true, mode="sequential")

//Various options for controlling formatting; autoStyleQuotes=true renders nicer dialog
quotes but reports errors if you have any imbalances.
$configureRenderer(autoStyleQuotes=false, timeStyle="header", defaultTimeStyle="red",
zeroTimeStyle="hide")
```

Lastly you will see some commands that configure various **sections** of your case, such as the **Front** section (where your cover page, table of contents, and instructions go), as well as the main **Leads** section (where the bulk of your text will go), the **DOCUMENTS** section, and the **HINTS** section, among others.

Most of the section configuration functions are just specifying sensible values for how entries in that section should be laid out. Most specify that text should be laid out in a single column, and fields like **sectionBreak**, and **leadBreak** control when page breaks are injected into the pdf. For example, a common entry in spare sections is "soloFacing", which means page breaks ensure each subsection begins on an odd-numbered page.

The `$configureLeads` function is a bit more involved, because that's where most of your story will be told, and where you sometimes need to handle things differently:

```
$configureLeads(sectionBreak="soloFacing", leadBreak="solo", leadColumns=taskLeadColumns,
childPlugins="nynoir", time=60, address="auto")
```

Above you will see that the Lead section column count (**leadColumns**) is handled differently, since the website will control this to produce different versions of your case pdf depending on whether it is intended to be printed or read onscreen. You'll also see a setting to control the default duration of leads, and an option that tells the Casebook PDF builder that the case is for New York Noir (childPlugins="nynoir"), and that it should automatically add the address info from the database for entries in the Leads section. For more information on the options available for the `$configureLeads` and other section configuration functions, see https://nynoir.org/authoring/casebookReference#configureLeads

We can leave all of this as is, since we won't need to modify it for our tutorial.

CONTINUED ON NEXT PAGE

 # SETUP

The setup section is where you will define stuff specific to your case. Here you will define each day of your case, the documents used in your case, and any special tags used in your case.

We will be adding to the **# SETUP** section over the course of this tutorial.

> As always, you can refer to the online document https://nynoir.org/authoring/casebookReference for help regarding any functions. You can also ask questions on the Discord channel.

This page intentionally left blank.

Creating The Front Matter (Cover Page, Instructions, Etc.)

Even before you get to the introduction to your case, you'll want to edit the # **FRONT** section of your case. This is where you control the look of the cover page, add a table of contents, and provide instructions to the player.

If we look at the # **FRONT** section we will see 3 child sections (COVER, CONTENTS, and Instructions). Let's look at these in turn:

The **COVER** section generates the cover page and is made up of just two function calls. The first specifies a background full page image, and the second just generates the default built-in cover page text which renders the title and other game information for you automatically.

```
## COVER "Cover Page"
$pageBackground(path="portraits/20231207_153848.jpg", opacity=0.33)
$blurbCoverPage(): { }
```

Next up we have the **CONTENTS** child section, which is just the table of contents. A single function call `$toc` is all we need here. You'll notice the `$(..)` syntax for configuring this section, which will become familiar to you. This is our way of overriding default settings for this section, to ensure that the table of contents is by itself an odd page (soloFacing), so that it's not on the back of the cover page.

```
## CONTENTS "Table of Contents" $(leadBreak="soloFacing", dividers=false)
$toc()
```

CONTINUED ON NEXT PAGE

Lastly we have the **Instructions** child section. The single call to `$instructionsGame` generates multiple pages of instructions. There are too many available options to go into in this tutorial, but you can read the documentation for the function here: https://nynoir.org/authoring/casebookReference#instructionsGame. This function can generate instructions specific to the features used in your game, and can even insert copies of various extra pdf sheets for the player to write on.

If we wanted to add custom instructions, we could do so here just by writing additional text after the `$instructionsGame` call.

```
## Instructions
$instructionsGame(events=true, flexTime=true, wrapUp=true)
```

Creating Your First Day

OK, it's time to start our case properly now.

In New York Noir, almost all cases are organized into one or more **days**. Having two or more days is one of the things that sets New York Noir cases apart – allowing you, the author, to create events that happen between days in order to make the world feel more alive.

Each day will have its own top-level section. Within that section, you might have an introduction that the player reads when starting the day, and then perhaps an end-of-day event that will tell them what happens when they end that day.

> Note: The way the ending of days is handled in New York Noir can be one of the most intimidating aspects of the Casebook language for new authors, because it is such an unusual concept involving keeping track of time, and concepts such as "events" and "required tags" (concepts which you will learn about later). For now, let's just focus on the basics and leave events for later.

Let's start by just creating a single day.

The very first thing we will need to do, before writing our Day 1 text, is add a line to the # **SETUP** section to tell Casebook about day 1. If you don't do this, you'll get an error when you try to build your pdf.

Let's go to the # **SETUP** section and add a line configuring day 1:

```
$configureDay(1, "normal", date="6/20/1915", start=10, end=15, leadColumns=1,
sectionBreak="beforeFacing", leadBreak="none")
```

> **NOTE**: You can ignore most of the parameters of this function for now, and just focus on the basics. The first argument is 1 for day 1. The start and end values tell Casebook that your day starts at 10 AM and ends at 3 PM, and the date specifies that the date of the case is June 20th, 1915. For a full list of configurable features you can see https://nynoir.org/authoring/casebookReference#configureDay.

> If you add multiple days to your case, you will have to add a `$configureDay` line for each day of your case.
>
> This pattern, where we need to define something in the # **SETUP** section before we can add the actual content of the item later on, is a common one in the Casebook language, and it's a common one in most programming languages. It helps the system to flag unintended errors, and means that the setup section of your document holds a concise and compact description of many of the important details of your case.

CONTINUED ON NEXT PAGE

Now edit your document so that the top-level # **D1** section (the first day of our case) looks like this:

```
# D1 "Day One"
## "Introduction"
$blurbDay(1,"dayTimeStart")
This is my first day at work as an NYPD detective.
```

It should look like this in the pdf:

Day One

Introduction

10 AM - Sunday, June 20th, 1915

This is my first day at work as an NYPD detective.

Note that the "D1" part does not show in the rendered page. That's because "D1" is the **ID** of the section, which is for you to use internally. The **label** is what's shown to the user in the rendered pdf ("Day One").

The function call `$blurbDay(1,"dayTimeStart")` generates text about the first day's date, day, and time, based on how we defined the first day in the # **SETUP** section.

CONTINUED ON NEXT PAGE

Formatting Text

Great, we have some text. Now let's see how to make it look fancier.

Markdown

Casebook uses the standard "Markdown" syntax for changing the look of text, like making text bold or italic. Let's see some examples:

The *Casebook* syntax:

```
This text is **bold**.
```

The generated text:

This text is **bold**.

Let's have a look at some more examples, presented in a bulleted list style:

The *Casebook* syntax:

```
* bold **Markdown**.
* italic *Markdown*.
* Strikethrough ~~Markdown~~.
* bold and italic ***Markdown***.
* underlined __Markdown__.
* bold and underlined &&Markdown&&.
```

The generated text:

- bold **Markdown**.
- italic *Markdown*.
- Strikethrough ~~Markdown~~.
- bold and italic ***Markdown***.
- underlined Markdown.
- bold and underlined **Markdown**.

Above, you see that **bullets** were added too. To do so, just add a * at the start of the line. To indent even more, add 2 more spaces before the * like this:

The *Casebook* syntax:

```
* item 1
* item 2
  * nested item 1
  * nested item 2
    * even more nested
```

The generated text:

- item 1
- item 2
  - nested item 1
  - nested item 2
    - even more nested

Google Docs Specifics

It's important to understand that any graphical formatting you do in your text editor (Google Docs, or whatever), will be **ignored** when generating a pdf using our tools! That's because Casebook defines its own "markup" language which controls such things. This language is much better suited for precise and consistent control over the look of cases, and it allows the tools to do some fancy stuff like properly formatting your case in a variety of formats (single column, two columns, etc.). Philosophically, you should focus on the CONTENT of your writing, not the graphical layout – let the tools handle that part.

There are a couple more "Gotchas" that you have to watch out for when using Google Docs.

- One is that if you're editing your case using Google Docs, and you put an asterisk at the start of a line, Google Docs may automatically turn that into an indented bullet point. This is counter-productive because Casebook will ignore any such formatting in your document. To avoid this, I recommend you get into the habit of using a period (.) instead of a * at the far left of a line for indenting bullet points. They work identically, but Google Docs won't mess with them!
- Another is to make sure you go into the menu Tools -> Preferences and uncheck the option "Use smart quotes". If you don't, you can run into weird errors if Google Docs gets confused about the use of unicode double-quote characters in your document.

White space (tabs, spaces and blank lines)

One more thing to mention is that extra spaces and blank lines in the text will often be automatically trimmed by the Casebook language. If you want to explicitly force them, use the ~ on a single line for an empty line and a ~ in a sentence for a space.

Here's an example of text that could be added to our intro. For simplicity, I won't add it but will just show you here:

The *Casebook* syntax:

```
This is my first day at work as an NYPD detective.
I don't realize it.
~
It's my **first** day as a __detective__ … ~~~ It has always been my dream and it's
now a reality!
~
$box():{
Why?
 * I have always been fascinated by detective life.
   * My father is a detective.
    * I have such good memories of all the detective stories I read.
* We've been robbed once, and ever since that time I've wanted to remove those bad
guys from the streets.
}
```

The generated text:

This is my first day at work as an NYPD detective.

I don't realize it.

It's my **first** day as a detective … It has always been my dream and it's now a reality!

Why?

- I have always been fascinated by detective life.
  - My father is a detective.
  - I have such good memories of all the detective stories I read.
- We've been robbed once, and ever since that time I've wanted to remove those bad guys from the streets.

Functions

Casebook includes some functions to help you format text in more interesting ways. One of the most common things you will want to do is put text inside a nice box. Let's write "This is a boxed text." in a box using the `$box` function.

The *Casebook* syntax:

```
$box(): { This is a boxed text. }
```

The generated text:

This is a boxed text.

As another example, a common way to start the introduction to each day is by displaying the current date, time and day of the week. We can ask Casebook to generate a standard line of text that displays the start of a given day, which generates the text below. Other helper functions can be used to insert text referring to the day of the week for a given date, etc.

The *Casebook* syntax:

```
$blurbDay(1,"dayTimeStart")
```

The generated text:

10 AM - Sunday, June 20th, 1915

You've now had your first taste of using **functions!** We'll see many more before we are done with the tutorial.

Functions in Casebook let you perform more advanced operations, and there are a lot of functions available. They always start with a `$` character, followed by the function name, parameters inside the parentheses, and sometimes a block of text in between curly braces, which can span multiple lines.

For comprehensive documentation on the available functions, see: https://nynoir.org/authoring/casebookReference.

This page intentionally left blank.

Creating Leads

Most of your case story will be told as the player visits people and places they find in the directories. So most of what you will be doing is writing the text of scenes that are organized into existing ”leads” corresponding to specific people or places. Let’s see how this is done.

Leads from the Directories

In the white and yellow pages, you can see leads that come from the directories. They are defined with a unique identifier: "X-YYYY" :

- *X* is a number from 1 to 9
- *YYYY* a 4-digit number.

Let's use this one:

– 13th Precinct 240 E. 21st St ..... **GD** 022 **8-4077**

So you understood that the lead is **8-4077**.

In the # **LEADS**, I refer to it as :

```
## 8-4077 "13th Precinct"
You visit the 13th Precinct. Other detectives welcome you.
```

If you look at the Table of Contents near the start of the pdf, you will see that Casebook has automatically added an entry for the "8-4077" lead to the Table of Contents. Using Casebook ensures that your pdfs always have a complete Table of Contents.

Finally, since this is a lead found in the directories, the player usually finds their own way there. This is the fun part of the investigation. But if you need to, you can direct the player there explicitly by writing:

The *Casebook* syntax:

```
$goLead("8-4077")
```

The generated text:

- Go to 8-4077 (p.54)

IMPORTANT!: A large case may have hundreds of leads, but you don't have to worry about the order of your leads in the # **LEADS** section. When your pdf is built, Casebook will automatically sort your leads into alphanumeric order and list them in the Table of Contents. Feel free to organize them in your document in whatever order you see fit – and don't forget you can use blank lines, comments, and visual page breaks in Google Docs to help you break your leads into groups for your own benefit.

Setting Lead Time (Duration)

While not required, most New York Noir cases use a **clock**. This means that the player starts each day at a certain time, and most leads take a certain amount of time to visit (typically 30 or 60 minutes).

First you need to ensure that time tracking is enabled.

Check that in the # **OPTIONS** section, the following *clockMode* line is set to true:

```
$configureClock(clockMode=true)
```

You can configure the default time each lead takes in the # **SETUP** section when you call the `$configureLeads` function:

```
$configureLeads(sectionBreak="soloFacing", leadBreak="solo", leadColumns=1,
childPlugins="nynoir", time=60, address="auto")
```

The default time per lead is set above to 60 minutes.

To customize the time that a lead takes to visit, you specify options for the lead using the following syntax:

Default time lead:

```
# 8-4077 "13th Precinct"
```

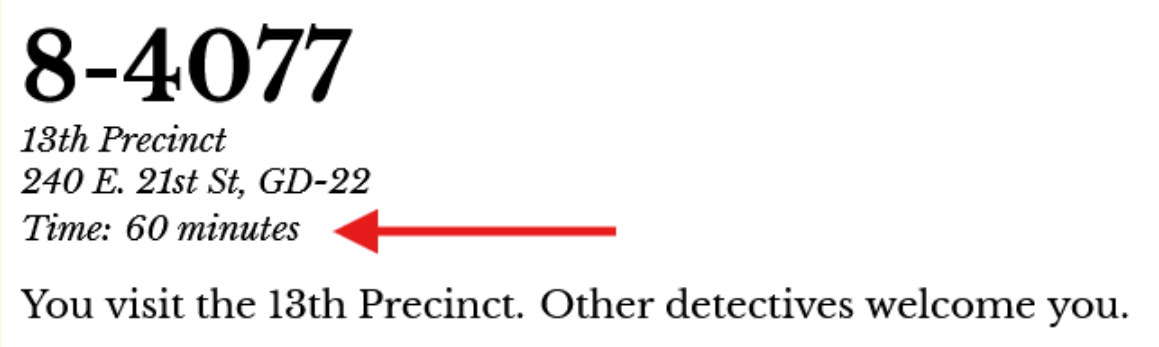
120 minute lead:

```
# 8-4077 "13th Precinct" $(time=120)
```

A lead that takes no time:

```
# 8-4077 "13th Precinct" $(time=0)
```

When a lead has a non-zero time duration, the duration will be shown under the lead header:



Autogenerated Lead Numbers

In addition to creating leads for people and places in the directories, you can also create custom leads that are not associated with people or places. You might use such custom leads to hold story sections that are triggered by events, or by branching choices the player makes, etc. You create such custom leads by giving them a custom internal ”id” name, and asking Casebook to assign them a unique lead # when the pdf is built.

As with tags, you refer to the custom leads by the internal id name you assigned, but the player sees a randomized value that disguises the significance of the lead. By specifying ”autoLead=true” you are telling Casebook to take care of assigning the lead # and sorting it into the correct place in the pdf.

To do so, add a dedicated entry in the section # **LEADS** like this:

```
## BreakRoom $(autoLead=true)
You encounter your new partner Bill.
```

And in the current section

The *Casebook* syntax:

```
If you want to go to the break room $goLead("BreakRoom")
```

The generated text:

If you want to go to the break room go to 7-3807 (p.53)

Inline Leads

You can generate leads directly when you are writing your story, using the `$inline()` function. This is a powerful function that makes it easy to write branching and special entries. When casebook encounters an **inline** block, it will automatically assign it a random and unique lead # and place the text inside that lead (properly sorted into the lead section), and then direct the player to read that lead. In this way you can write out branching or conditional paragraphs in one place, and let Casebook do the work of hiding the text under lead entries that won’t be read by accident. Let’s write one.

The *Casebook* syntax:

```
If you want to attend the staff meeting $inline():{ The staff meeting occurs. There
was a missing persons report. Since we must wait 24 hours before investigating, you
will have to check on that tomorrow if this is still the case. }
```

The generated text:

If you want to attend the staff meeting go to 3-6077 (p.49).

Finding Leads For Your Case

Note: Let's take a break from learning how to write your case, and discuss how to find leads from the directory that you want to use in your case. You could of course use the printed directories, but I'm going to give you some tips for how to use the online tools.

Let's add a lead for the home police station of our detective:

- Go to https://nynoir.org/map/, the online map & search directories.
- Click on the Yellow Pages, the **Y** on the right section.
- Select the **Police (NYPD) - Neighborhood Precinct**.
- Click on the 13th Precinct. That's the police station in the Gashouse District Neighborhood. You'll see it highlighted on the map.
- Click on the **star** button in the selected zone to make it a bookmark so we can find it more easily in the future.

Once selected, you should see that the 13th Precinct is:

- Lead # 8-4077
- On block GD-22 in the Gashouse District

You could just write down the lead # (8-4077) but you could also click the clipboard button to copy the lead # to the clipboard, or click on the Log, the **L** on the right section. You will see the same info which you can copy and paste if needed.

Using the online map & search tools, you can do all sorts of searching and filtering to help you find places that you might want to use in your case. As an author you can use the Advanced search tab to do all kinds of wildcard searches to help you find related leads.

Using Tags

It's time now to talk about **tags**.

There are different types of tags, but in general tags are used to keep track of when a player gains access to certain things, or when we want to keep track of places a player has gone, or decisions they have made.

A **tag** is defined using the `$defineTag()` function. Each tag has a unique "id" which consists of two parts separated by a period (dot). The first part is a prefix that defines the type of tag, and the second part is a unique name of your choice.

The two most common types of tags are "cond." tags and "doc." tags. The former is used to keep track of when a player has visited a certain lead or made a certain choice. The latter is used for keeping track of access to documents.

We'll learn how to make *document* tags later. For now let's learn how to create a conditional ("**cond.**") tag that will keep track of when the player visits a specific lead to fill their car up with gas.

The first thing we have to do is define the tag in the # **SETUP** section:

```
$defineTag("cond.gas", label = "Fill the gas tank", deadline=1)
```

Notice that we set "deadline=1" for the tag, which tells Casebook that the player MUST find this tag before the end of day 1. You'll see how this is enforced when we look at "Events" later.

Once that is done we can check if the player has the tag whenever we want, and assign the tag whenever we want.

Let's create the gas station lead where we tell the player to mark that they have the tag, using the `$gainTag` function:

The *Casebook* syntax:

```
$gainTag("cond.gas")
```

The generated text:

Circle **Marker A1** on your case tracking sheet.

Notice how Casebook has added instructions for the player, and has assigned the tag a letter and number so that the player is not told exactly what the tag refers to.

CONTINUED ON NEXT PAGE

Now let's write a lead that checks whether the player has this tag:

The *Casebook* syntax:

```
$missingTag("cond.gas"), $inline(time=30): {
You realize that you would not have enough gas to go there, so you go back to the
precinct. }
$hasTag("cond.gas"), $inline(time=60): {
You drive to the shop and receive your new gun.
}
```

The generated text:

If you have **not** circled **Marker A1** on your case tracking sheet, go to 4-3335 (p.50).

If you have circled **Marker A1** on your case tracking sheet, go to 5-6952 (p.51).

Again, as the author you will always refer to the tag by its ID ("cond.gas") while Casebook will always refer to it by a letter-number pair, so that the player does not know what the marker refers to.

Gaining Documents

Documents in New York Noir are special because they are normally placed near the end of the pdf in the # **DOCUMENTS** section, and players are told when to record that they have gained access to each document.

I'll put off teaching you how to create documents until we get to the # **DOCUMENTS** section itself, but let's have a quick look at how you inform the player from within a lead when they have gained access to a document.

> **TIP**: Players are never allowed to just browse the Documents section. You'lll need to explicitly tell the player when they gain access to each document, even if it's just in the Introduction.

To tell the player they have gained access to a document, you use the same mechanism you use to inform them when they gain a conditional tag, using the `$gainTag` function.

Let's assume that we have already defined a document tag in the # **SETUP** section as follows:

```
$defineTag("doc.bossNote", label="Memo Letter from Boss")
```

Now let's see what gets rendered in the pdf when we give the player the document tag:

The *Casebook* syntax:

```
You find the note from your boss:
$gainTag("doc.bossNote")
```

The generated text:

> You find the note from your boss:
>
> Circle **Document 1** on your case tracking sheet. You have gained access to **Document 1** (Memo Letter from Boss), which can be found at the back of this case book on page 57.

Notice that although we refer to the document by its tag id ("doc.bossNote"), the instructions to the player are much nicer. The player is not only told the document number, label, and where to find it, but is helpfully reminded to make a note of this fact on their Case Tracking Sheet.

CONTINUED ON NEXT PAGE

TIP: In the # **DOCUMENTS** section near the end of this tutorial, you'll see how we actually create this Memo Letter document, but you can jump there now if you like by following the document link above or going to the specified page number.

Working with Events

In writing a case for New York Noir, it is common to break up the story into multiple days, and to use ”events” that trigger the end of each day. In addition, cases may have other events that are scheduled and triggered at specific times.

An **event** is a way of forcing a player to read a certain section after a certain number of turns. It makes a case feel more dynamic and tense. It can create a sense of tension by limiting the amount of turns a player has before some consequence is triggered.

> Because events can be somewhat unintuitive to players, the Casebook language has a number of functions that help you create events and instruct the user on how to handle them, as well as functions to handle the common mechanic of creating special end-of-day events.

Let’s modify our first day to create an **End of Day Event** that we want to force the player to visit at the end of the first day. Since we defined the end day of day 1 in the # **CONFIG** section, the system knows the time when day 1 ends (3 PM).

When the end of the day comes around, we want to direct the player to a special lead entry that we will write.

The first thing we’ll do is create a lead that we are going to direct the player to jump to after the event triggers. If this were some random event like getting hit by a car we might put it in the # **LEADS** section with our other leads, but for end of day events it’s common to make a special end of day event within the top level day section (e.g. # **D1**). Here’s what it might look like:

```
## "Evening of Day 1"
It's the evening of day 1.  What a day for your first assigned case! Go to the conclusion
of the game and answer the questions to solve the case.
```

Now, back at the end of the **Introduction** to Day 1, we are going to create the event which will instruct the player on what to do. We do this using the `$event()` function. This will generate text telling the player what to do and when. We’ll also add some instructions for the player to start their day.

Here’s what the *new* Day 1 section looks like:

CONTINUED ON NEXT PAGE

```
# D1 "Day One"

 ## "Introduction"
$blurbDay(1,"dayTimeStart")

This is my first day at work as an NYPD detective.

$event(day=1, when="evening"): {$instructionsEvening(1, next="Evening of Day 1")}
$instructionsStartDay(1)

 ## "Evening of Day 1"
What a day for your first assigned case! Go to the conclusion of the game and answer the
questions to solve the case.
```

To see what this looks like in the rendered PDF just peek at the next page.

There is something very important to take note of: When the event triggers, the player is not directed directly to the "Evening of Day 1" entry. That's because by default the `$event` function creates an automatically generated lead, and then fills this lead with the text generated by the contents of the event block (`$instructionsEvening`).

Only at the end of *that* lead will it make sure the player has found all the required tags for the day, and only **then** will it send them to the specified section ("**Evening of Day 1**"). It's weird, I know. But visit the Day One section and you'll see how it works.

Notice also that when you go to the event lead, you will see text telling the player that they must have found Marker A1 and Document 2. That's because when we defined those 2 tags we set their **deadline** to day 1.

Day One

Introduction

10 AM - Sunday, June 20th, 1915

This is my first day at work as an NYPD detective.

Record the following **mandatory** event in your schedule:

- **When**: Today, day #**1** (Sunday, June 20th, 1915).
- **Time**: 3 pm
- **Where to go**: 6-7138 (p.52)
- **Mandatory**: YES.

When you meet the conditions described above, visit the lead specified.

You are now ready to start your day. On a blank case log sheet record that it is Day **1** (**Sun, Jun 20th**) and that the current time is **10 am**. Then close this case book and begin searching for leads in the directories. Don't forget to keep an eye on your schedule for the **3 pm** event above that will trigger and lead to the conclusion of your work day.

STOP!

Stop reading this case book now, and begin searching for leads in the directories.

Do not turn the page. You should have set an event which will trigger at the end of **day 1**, which will instruct you on what to do when the day ends.

Evening of Day 1

What a day for your first assigned case! Go to the conclusion of the game and answer the questions to solve the case.

This page intentionally left blank.

LEADS

STOP!

WARNING! Do **not** read through the rest of this document like a book from beginning to end. Lead entries are meant to be read individually only when you look up a lead by its number.

Close this book now and follow rulebook instructions for looking up leads.

1

1-9896

Gun Shop
638 Lexington Ave, TB-25
Time: 60 minutes

If you have **not** circled **Marker A1** on your case tracking sheet, go to 6-3677 (p.52).

If you have circled **Marker A1** on your case tracking sheet, go to 7-5997 (p.53).

3

3-6077

Inline Leads (p.34) contd.

The staff meeting occurs. There was a missing persons report. Since we must wait 24 hours before investigating, you will have to check on that tomorrow if this is still the case.

4

4-3335

Using Tags (p.37) contd.
Time: ***30 minutes***

You realize that you would not have enough gas to go there, so you go back to the precinct.

5

5-5104

Gas station
210 E. 21st St, GD-22
Time: ***15 minutes***

You filled up the car with gas.

Circle **Marker A1** on your case tracking sheet.

5-6715

Hint for Marker A1 (p.71) contd.

Hint: It's in the same building lot. Go to the Esso Gas (Gashouse District) (5-5104).

If this hadn't already occurred to you, mark **2** demerit boxes on your case tracking sheet.

5-6952

Using Tags (p.37) contd.
Time: 60 minutes

You drive to the shop and receive your new gun.

6

6-3677

Gun Shop (1-9896 on p.48) contd.
Time: ***30 minutes***

You realize that you would not have enough gas to go there, so you go back to the precinct.

6-7138

Introduction (p.43) - Scheduled event for day 1

It's **3 pm** on **Sunday, June 20th, 1915**, and day **#1** is ending.

> The following **2** items must be found before you may end the current day:
> - **Marker A1**
> - **Document 2**

Record +1 reputation on your case tracking sheet for each of these items that you have already found.

If you have not yet found **both** items, resume searching for leads now until you have (using hints if needed), and consider yourself in **"overtime"** for the rest of the day. In overtime, time does not advance past **3 pm**.

Furthermore, while in overtime, for every **2 (time advancing) leads** you visit that do not result in you gaining one of these items, you must read a hint to help you find one (taking demerits for hints as normal).

When you have found **both** required items, proceed to: Evening of Day 1 (p.45).

7

7-3807

Contd. from Autogenerated Lead Numbers on p.34
Time: 60 minutes

You encounter your new partner Bill.

7-5997

Gun Shop (1-9896 on p.48) contd.
Time: 60 minutes

You drive to the shop and receive your new gun.

Circle **Document 2** on your case tracking sheet. You have gained access to **Document 2** (Your gun), which can be found at the back of this case book on page 59.

8

8-4077

13th Precinct
240 E. 21st St, GD-22
Time: 60 minutes

You visit the 13th Precinct. Other detectives welcome you.

DOCUMENTS

Now let's take a look at how to make various documents.

Documents in New York Noir are special because they are normally placed at the end of the pdf, and players are told when to record that they have gained access to each document. The Casebook language includes functions to help you inform the player where to find documents and how to keep track of which ones they gain access to.

However, while all documents are stored here in the # **DOCUMENTS** section near the end of your file, you will still need to **define** each document ahead of time, in the # **SETUP** section, using document **tags**.

> Note that document tags need to be defined in the same order that the documents are listed in the # **DOCUMENTS** section.

Let's start by going back to the # **SETUP** section near the top of our case and add lines to define 3 documents that we will create:

```
$defineTag("doc.bossNote", label="Memo Letter from Boss")
$defineTag("doc.gun", label = "Your gun", deadline=1)
$defineTag("doc.newspaper", label = "Newspaper front page")
```

With these 3 documents *defined* in the # **SETUP** section, we can go back to the # **DOCUMENTS** section and actually create the documents themselves.

> **IMPORTANT**: While every document will eventually be assigned a number, you as the author should never refer to documents by their number, and should instead refer to them by their tag id – in the same way that when you create conditional tags they will be assigned letter-number pairs (e.g. A1), but you should always refer to them by their internal tag id. This makes it possible for you to remove, add, or rearrange documents and tags at any time without ever having to worry about the numbers changing. It's one more way that using Casebook makes it easier for you to maintain your case.

This page intentionally left blank.

Document 1

Memo Letter from Boss, from Gaining Documents (p.39)

Each document in the special Documents section starts with a header that specifies the document **tag** id, like this:

```
## doc.bossNote
```

When Casebook sees this, it will render the document page using the **label** that you used when you defined the document tag – so if you look at the top of this page you'll see two lines in a large bold font, specifying first the document number (**"Document 1"**) and then the document label (**"Memo Letter from Boss"**).

The contents of a document are written in the exact same way as we write leads, using markdown syntax etc. But often documents will make use of Casebook functions that are designed to let you do some visually interesting things. For this document, we want to display some typewritten text on a yellow sheet.

Let's see how to do that:

CONTINUED ON NEXT PAGE

The *Casebook* syntax:

```
$format(style="tornYellow"): {
$format(style="typewriter"): {
$date("6/9/1915", year=false)
~
Dear John,
Welcome to the 13th Precinct. We have received your new gun. Please go to the Home
Defense & Gun Shop to obtain it. Also, don't miss your first briefing today at 2 PM
in the local 211.
~
P.S.: Your patrol car is almost out of gas -- don't forget to refill it.
} // end typewriter
} // end tornYellow
```

The generated text:

```
Wednesday, June 9th

Dear John,
Welcome to the 13th Precinct. We have received your new
gun. Please go to the Home Defense & Gun Shop to obtain
it. Also, don't miss your first briefing today at 2 PM in
the local 211.

P.S.: Your patrol car is almost out of gas -- don't forget
to refill it.
```

Document 2

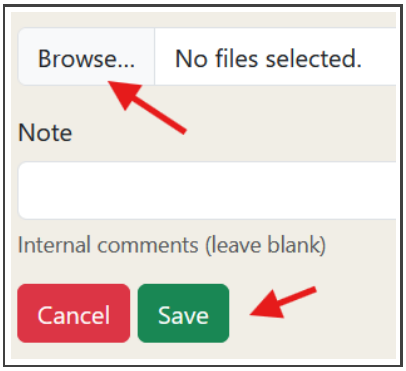
Your gun, from 7-5997 (p.53)

To show that the player gained a gun, we want to add a picture of it. So I will show you how to do it.

You must have a local image file on your computer to do so. Go to the Case page of your game and click **Uploaded files**.

Click **Upload new files for case**:

Click **Browse** and select your file, or drag and drop the image(s) into the file input, and click **Save**.



From now on, I will assume that you've uploaded the *gun.png*.

Then you can refer to the uploaded file by writing this:

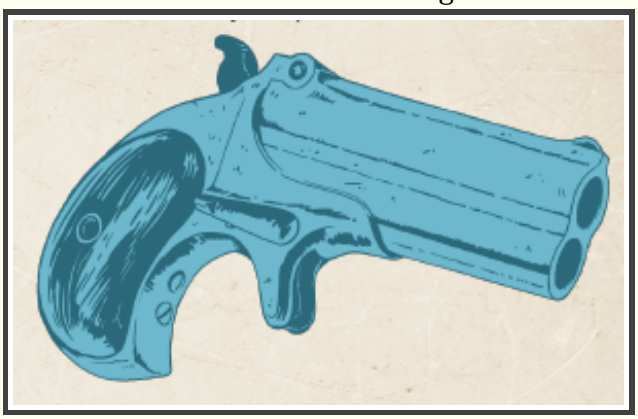
CONTINUED ON NEXT PAGE

The *Casebook* syntax:

```
$image("gun.png", height="2in", caption="Your new Winchester gun", captionPos="top",
borderWidth=2, padding=2)
```

The generated text:

Your new Winchester gun



Document 3

newspaper front page

A newspaper is a common kind of document. Casebook includes some functions to make newspapers with different common banners for the different newspapers, and the ability to write headlines, use images, and change the font and style.

> The Casebook function calls to create a newspaper can quickly get complicated because of the nested details and all of the various options. I would suggest that you start simple and test as you add each new article or make each change. That way, if you encounter an error building your pdf, you will have an easier time figuring out where the problem is.

Let's look at some code to generate a newspaper front page:

```
$newsPaper(style="nyTimes", day=1, fontSize="14", issueString="",
headlineStyle="case=upper|bold=true", columns=2): {
  $newsArticle(headline="COAL INDUSTRY DOOMED! SCIENTIST EXTRACTS BOUNDLESS FORCE FROM
FROZEN ATOMS", byline="By Edwin E. James"): {
Dr. Alistair Finch threw the Academy into an uproar yesterday by demonstrating a
miraculous apparatus that yields a continuous galvanic current from a frozen chamber. By
causing the disintegration of atomic bodies within a pressurized receptacle of liquefied
air and radium salts, this new "Transmutation Process" creates limitless energy without
combustion or smoke. Prominent capitalists are already hailing it as the absolute death of
the coal trade.
   $image(path="portraits/20231207_155323.jpg", style="frame", caption="Dr. Finch",
captionPos="bottom")
  }
 }
```

> Note that we use one of the shared portrait images in our newspaper ("portraits/20231207_155323.jpg"), so we don't have to upload it specifically for this case.

Now let's see how that looks rendered in the pdf:

CONTINUED ON NEXT PAGE

"All the News That's Fit to Print."

The New York Times

SUNDAY, JUNE 20, 1915 2 cents

COAL INDUSTRY DOOMED! SCIENTIST EXTRACTS BOUNDLESS FORCE FROM FROZEN ATOMS

By Edwin E. James

Dr. Alistair Finch threw the Academy into an uproar yesterday by demonstrating a miraculous apparatus that yields a continuous galvanic current from a frozen chamber. By causing the disintegration of atomic bodies within a pressurized receptacle of liquefied air and radium salts, this new "Transmutation Process" creates limitless energy without combustion or smoke. Prominent capitalists are already hailing it as the absolute death of the coal trade.

Dr. Finch

END

Conclusion

Here's where we prepare the player for solving the case when ready.

You are about to read the final questions for this case. Once you read them, there is no going back into the field.

Proceed to the final questions only when you are ready to end your investigation.

STOP!

Proceed only when you are ready to answer questions.

Questions

Now you'll want to present the player with some questions, and then give them the answers on the following page. You could just write these questions and answers as plain text, but Casebook has some helper functions that let you format questions and answers in a consistent way, keeping track of potential maximum score, etc. Let's use the Casebook helper functions `$defineQuestion` to present some questions nicely formatted:

The *Casebook* syntax:

```
$defineQuestion("Q1", "What is the brand of your new rifle?", points=20, lines=1)
$defineQuestion("Q2", "What is the name of your new partner?", points=10,
lines=1)
```

The generated text:

Q1. What is the brand of your new rifle? (**max. 20 points**)

_____

Q2. What is the name of your new partner? (**max. 10 points**)

_____

Answers

And now we can show the player the correct answers and scoring guidelines on the next page. If we use the `$questionAnswer` function, Casebook will repeat the questions for us without us having to repeat ourselves, and the `$questionTotal` function automatically provides a reminder about the maximum possible score:

The *Casebook* syntax:

```
$questionAnswer("Q1"): {. ANSWER: Winchester.}
$questionAnswer("Q2"): {. ANSWER: Bill.}
~
$questionTotal()
```

The generated text:

Q1. What is the brand of your new rifle? (**max. 20 points**) ..... _____

- ANSWER: Winchester.

Q2. What is the name of your new partner? (**max. 10 points**) ..... _____

- ANSWER: Bill.

Total score from questions (max. 30): ..... _____

Epilogue - My first week in NYPD

I was proud of my first day.

Behind the Scenes: Postscript from the Author

After players finish your case, tell them a bit about the process of writing it, so they can get a look behind the curtain into the author's mind.

Full Walkthrough

It's also nice to provide a short walkthrough of the case, letting players see what a perfect playthrough might look like.

Here's one way an experienced detective might have solved the case:

Turn 1: Look at the Memo Letter (Document 1)

You get the following info:

- 13th precinct
- Gun at Home Defense & Gun Shop
- Need to fill car

Turn 2: Go to the gas station (5-5104: Esso Gas (Gashouse District))

Look up where the 13th Precinct is located in the Yellow Pages:

13th Precinct 240 E. 21st St ..... GD 022

Look for the GD district in the *Gas Stations* section of the Yellow Pages. There are two stations, but the Esso is closer.

You know the story; you could extrapolate what would be written here.

HINTS

It's very important to write hints for players that get stuck. This is most important for any "tags" that a player **must** find in order to finish the case. So while you don't have to have hints for everything that the player might encounter, you must provide hints for all required markers.

Hints are always placed in the section # **HINTS**, which is usually the last main section in your document.

You don't have to tell the player to go to "**5-5104**" (the gas station); he must find it himself. But you can help him if he cannot figure out where to go.

In general, whenever you have a tag that the player **must** find in order to progress in the game, you should add a hint to help them find this tag.

To provide a hint for a tag, we must name the hint section the same name as the tag, thus "## **cond.gas**".

Normally you will want to penalize the player for taking a hint, to create a disincentive for reading the hints. We accomplish this by instructing the player to mark **demerit** points when they read hints.

> Casebook has some helper functions to make it easier for you to write hints that cost the player demerits, and a special `$inlineHint` function that works like the `$inline` function, but is specialized for easily writing hints. There's also a special function `$autoHint` that will add a last resort hint that automatically directs the player to where they can find the required tag as a last resort.
>
> By using these functions, you can very easily write a bunch of hints in one place that require the player to jump around in the pdf so they won't accidentally read a spoiler by mistake. It's customary to have the first hint be a very gentle clue about the nature of the tag, and then have at least one more increasingly helpful hint before you end with an `$autoHint()`.

Let's write a hint for the "cond.gas" tag and give the player 1 demerit point for taking the first hint, and 2 demerits if they need a second hint. Note that before the actual hints start, we use a function to display some warning text to the player not to read hints unless they want help:

```
# HINTS
$blurbStop("hints")
## cond.gas
$mark(demerits=1)
You can only look at the neighboring buildings.
For another hint, $inlineHint(demerits=2): {Hint: It's in the same building lot. Go to
the Esso Gas (Gashouse District) (5-5104).}
$autoHint()
```

And here's how that hint entry renders in the pdf:

CONTINUED ON NEXT PAGE

STOP!

Do **not** access the hints section except when looking up a specific hint from the table of contents at the start of this case book.

Hint for Marker A1

(must be found by end of day 1)

☑ Tick **1** demerit box on your case tracking sheet.

You can look only at the neighboring buildings.

For another hint, go to *5-6715* (p.51).

If you still need help, as a last resort ☑ Tick **3** demerit boxes on your case tracking sheet, then visit one or more of the following leads where this item is obtained:

- Using Tags on p.37
- *5-5104* on p.51

This page intentionally left blank.

What's Next

If you've managed to write your first case for New York Noir, congratulations! I promise it gets easier. But your journey is only just beginning.

Your next step is to get some playtesters to try your case and give you feedback – or maybe to get a volunteer to go through your case and look for typos and issues. The good news is that the New York Noir community is friendly and helpful.

If you are ready for feedback, just join the *nynoir_caseauthors* channel in the **Co-op For Two Discord** Server https://discord.gg/GhBmg5JFPc.

There you will find users ready to playtest your case, offer constructive criticism, help you edit it, and answer any questions you might have about how to accomplish different things.

Publish Your Game

For playtesters and editors, you can leave your game unpublished and just share a link to your latest **preferred build**. But eventually you will want to **publish** your case in order to make it available to the public. Until you publish your case and make it public, it will only be visible to those who you give a direct link.

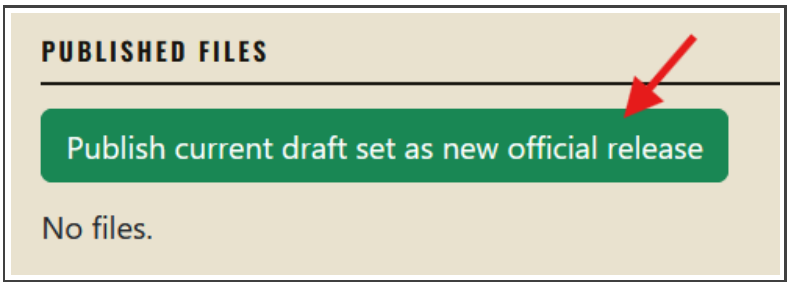
To publish your game to the public, first go to your game page and click on the *Generated PDFs* :

Then click on the *Build complete set of draft PDFs* and wait :

This will create different documents:

- soloForScreen_letter.pdf : This is a pdf where each lead is on its own page, which makes it well suited for reading on a device (pc or tablet) or online, but not suitable for printing.
- twocol_letter.pdf : This pdf lays out the case for double-sided printing in two columns to minimize printer ink and number of pages, formatted for letter-sized paper (suitable for players in the USA).
- twocol_a4.pdf : A4 version of the previous document (suitable for European players).

The **draft** build lets you examine the produced pdfs and verify that they are to your liking before you publish an official public version. If you are satisfied with the draft build, you can promote the draft version to an official published version by clicking:



And that's it! Your game is now available to the public, and will be listed on the public Cases page (https://nynoir.org/cases/).

Beyond the Basics

You can continue your journey by looking at other authoring documents and guides on this page: https://nynoir.org/authoring/.