1. Preface: im gonna nerd ot about dungeons and dragons and say a lot of lingo so if you’re confused about what I’m saying so I’ve put a lingo link in the chat
2. Ok so story time – 2 year long campaign I was dming was coming to an end
   1. Tough week at work and had totally forgotten about an upcoming session
   2. I had 30 mins to prep before go time, I was stressed out I didn’t want my players to have a bad session or cancel last minute!
   3. Coincidentally I had this halfbaked untested combat generator that hadn’t seen the light of day in a while
   4. I took the gamble, generated the combat for that session in 20 mins and now here I am to share how that generator changed the way I DM
3. So hi I’m atty (he/him), associate game designer @ improbable, I’m still figuring out my procgen class but lets go with bard/druid for now, and GM for 3 years – and this is a talk about what it looks like when my procgen obsessions start leaking into my ttrpg life
4. K lets generate a monster together using this super basic python script that I wrote over the course of a weekend
5. Ok that was mostly for fun and so that y’all see how it works but we will use this stat block that I generated literally 30 mins ago for the rest of the talk, you’ll see why in a moment
6. So at first look this is weird right, a lot of doesn’t look like a proper fully fledged stat block from MM, here’s an MM stat block for reference
   1. That’s exactly what I was going for bc I wanted to make a lightweight, flexible, custom tool to go with my improv heavy dm style
7. So lets talk about the design of the generator briefly because that might offer insight as to why I made it the way I made it
   1. 5e already gives me a set of rules for what a player and a monster can don on their turns in combat, so it was easy to model the output of the generator based on statblocks and actions in the books
   2. However I realized that what I really needed to define was the kind of output that I wanted to see at the end – what kinds of output lead to interesting combat encounters?
   3. Matt colville made a video about action oriented combat – it highlights a design philosophy about how we can make combat in 5e engaging by telling an evolving story through the combat. He says to pick a monster, think about what it would do under certain situations and design mechanics around those situations. – what does the goblin boss do when its cornered? Yell at his subordinates to attack all at once! Get rid of the threat!
   4. I ended up borrowing from colville’s AOC design philosophy because I’ve been using it for years and it works great for me – but I decided to invert that philosophy
   5. I would generate the numbers and mechanics first and then use those as constraints to create the monster, that way I don’t have to the big think, I just gotta justify why things are the way they are, which to me is a lot easier and faster. This eventually led to my new DMing motto, here it is.
   6. The generator deals with the numbers stuff that I find tedious to think about during prep and lets me focus on storytelling by offering a level of abstraction in the descriptions for the monster’s mechanics in the output. It does this by describing mechanics in adjective-verb pairs
      1. Eg. Special action? Drain? Resource?
      2. Could mean anything in a limited mechanical subspace but its purpose is to inspire my creativity by offering constraints for me to work within
   7. So lets go back to this statblock -> after polishing it for 10 mins this is what it looked like
   8. This is princess Anjani, she was the subject to terrible torment in the castle of the city above, having gone through terrible things at the hand of the sultanate. It’s said that her restless and veangeful spirit haunts these sewers to this day
      1. That drain ability became a bonus action that allows her to drain PCs spell slots, and use them to fuel her own ethereal transformation
8. If my players stumble across her then there’s a good chance they might get a bit curious about this sultanate… oh whats this? A story thread?
9. As a DM I can pepper in these generated combats and weave the outcome or the story of the combat into the overarching narrative later, because I’m lazy
10. So with that I want to share some thoughts and questions I’ve reflected on as I started making more generators to save me prep time for my sessions
    1. what parts of GMing do you want to put more focus on than others, this may reveal what parts of DMIng feel tedious to you and what things you might want to automate, so that you can spend more time on a different facet of DMing that you want to work on
    2. Session prep can get lonely – at the table we’re improv-ing with our friends as NPCs, but while we’re prepping there’s often no one there to yes-and you, a generator could be that improv partner to help make session prep more creative!
    3. I’ve learned to embrace the dark side of random generation and lean into it, It’s made my ttrpg sessions more spontaneous and fun. Nothing feels better than telling your players that even you don’t know what a combat will look like and then revelling in the fear in their eyes. Less time big preppin more time big chillin
    4. What else can we automate in our TTRPGs? All this to ask “can we automate npcs dialogue? Faction wars? BBEG’s path to victory? Entire complex thematic dungeons? Skill challenges? Simulate the entire world between sessions?
11. I encourage y’all to try making a generator for your ttrpg games if you haven’t already, whatever that might look like for you. its quite fun in my experience,. Share your results with me on twitter!