

This program is very adaptable to any given game setting.

If you want to edit an existing class or feat, or change the ratings given to feats and class subfeatures (the name we gave to stuff like Sorcerer's bloodlines and Barbarian's instinct, etc.), simply open the corresponding workbook in Microsoft Excel and change the values.

Feat_Database.xlsx and subfeatures.xlsx, respectively.

For ratings, the values can range from 0 to 100, higher meaning the ability is more likely to be picked by a character specializing in that playstyle. 50 is the neutral value.

If you wanted to add a class to the game, that is also fairly straightforward. There are four steps involved:

1. Open ClassAttributes.xlsx in Excel and add a row for the class. This row needs a class name (keep the name the same between steps), a primary ability score, and an optional secondary ability score. These ability scores tell the program what to focus on in terms of level up ability score boosts and feats.
2. If you want to add more feats, either specifically for the class or to existing classes, open Feat_Database.xlsx. If you want to add a new class list worth of feats, duplicate or create a copy of the template, and name that new sheet the exact same that you named the class in the previous step.

For starters, to have multiple requirements in one cell, separate them with a semicolon, this will require all prerequisites in the list.

An "or" list is a little more complicated, you'll have to make multiple rows of the same feat entry with the same name and ratings, but with the different requirements. For instance, if a feat required you to be a Master in Perception and either Expert in Occult or Trained in Arcane, make one entry with a skill prerequisite "Perception:Expert;Occult:Master" and a second with "Perception:Expert;Arcane:Trained".

A dash ("-") indicates there is no requirement, and such a dash should fill any empty cell. Fill it out as follows:

- Name: Name of the feat
- Level: Minimum level required to take the feat
- Atk, Support, and Defence ratings: As explained above, these values can be zero to 100. Values above 50 mean that the feat is more likely to be picked by a character specializing in that playstyle, below 50 lowers the chance.
- WeaponType: If the feat requires a specific type of weapon, specify it here. Accepted values are 1H, 2H, Melee, Dual, Bow, Crossbow, Ranged, Thrown, and Any (also accepts a "-")
- WeaponReq: If the feat requires a specific weapon property, specify it here. Accepted values are FH (Free Hand), agile, volley, shield, and reload_0 (fastest reload speed).
- Feat Prerequisite: If the feat requires other feat(s), put them here.
- Skill Prerequisite: If the feat requires a certain skill to be Trained, Expert, Master, or Legendary rank, put Skill:Required Rank in the cell.

- Other Prerequisite: Can be several things, ranging from what spell list you cast from to what class features or subfeature you have (only checks by name). Too many to list here, look over the existing sheets for exact examples.
The rest of the columns are not yet implemented.
3. Open `Class_levelup_charts.xlsx`. Either duplicate the template sheet, or make a new sheet in this workbook and copy the contents of the template sheet into it. The template data isn't 100% necessary, it just sets everything up as per the game's default formulaic progression. The name of the sheet also has to match the sheet name in the prior steps. As for what to enter in the cells, the first thing you want to do is replace the Class in "Feat:Class" with the name of your class (the name of the sheet containing your class' feats, to be precise). Next:
 - The first column is level, this should count from 1 to 20.
 - Hit Points: The number of Hit Points gained at the level.
 - Skills: Due to how the program works, the first entry in each non "-" cell has to be the number of skill proficiency increases gained at that level. After that, you can assign a maximum rank with Maximum:Expert, Master, or Legendary. At level 1, you can assign any skills to start off as Trained by simply entering the name of the skill.
 - Class Features: This is where you tell the program when to pick certain subfeatures. Other than that, all of these are just names kept in a list for prerequisite purposes. In other words, put anything you want here, but only entries like "Subfeature:Bloodline" are actually parsed. You can have multiple subfeatures.
 - Feats: This is where you specify when the class gains feats, at what level, and *from what classes*. Yeah, you are not limited to feats from your class list if you do not want to be. For instance, if you were developing a Magus as a third-party class, you could pull from Feat:Wizard and Feat:Fighter, or you could make a feat list for Magus, *or you could do both*. You can also assign bonus feats with "Bonus:feat name", these are given to the class while ignoring prerequisites.
 - Proficiencies: This one looks complicated, but is actually very formulaic. There are four proficiency ranks: Trained, Expert, Master, and Legendary. These can be applied to anything, from skills to saving throws to weapon and armor. Anything not specified starts off as Untrained. Note that "Fortitude" is shortened to "Fort". For all possibilities, look at existing entries for existing classes.
 4. If you want to add a new class subfeature, open `subfeatures.xlsx` and copy the template sheet to a new sheet. This one does not have to be named the same as the class, instead name it whatever the class feature is called (Muse, for instance). Give it a few entries, each row should have a name and three ratings (one for offensive, support, and defensive, you get the drill).

And there you have it, Homebrew content is easy to add to this character optimizer.
Good luck and happy gaming!