How to debug the game engine with emacs/gdb

- these are very brief notes on how you can debug your `dhewm3` engine with `emacs` and `gdb`
  - this game engine will also have had python bots capability added
- firstly it is a good idea to ensure that you have no python bots running on your machine
- ```
  $ ps aux | grep python
  ```
- use the Unix command line `kill` command to kill any process running python and doom

Build a test map including python bot

- ```
  $ cd Sandpit/chisel/python
  $ ./developer-txt2map ../maps/python.txt
  ```


Build a test map including python bot

- now run `gdb` from within `emacs`
  - start `emacs` and press F5, which will compile dhewm3 and start `gdb`
- now exit from dhewm3
- in the `.gdbinit` emacs buffer add the line `break idPlayer::RegisterPython` before the line `run`
  - we are informing `gdb` that we wish to add a breakpoint in the `RegisterPython` method
- now press F5 again
  - this will run `gdb` and set the breakpoint as requested
- from a command line terminal run the python bot
- ```
  $ Sandpit/git-doom3/pybot-dhewm3/python-bot
  $ python python_doommarine_1.py
  ```


Build a test map including python bot

- `dhewm3` should start, now tell `dhewm3` to load in `tiny.map`
  - it should now connect to your python doom marine
  - and then it should stop at the break point in RegisterPython
- in `emacs` press `M-x gdb-many-windows`
  - ie `<ALT> x gdb-many-windows`
- you can now single step and inspect the game engine using the gui/mouse interface of emacs or the command line in the `gdb` window
  ```