# Python Developer Technical Challenge Development Notes:

Part One:

* Began by reading the attached journal article to establish a rough blueprint of the expected outcomes of the project. Noted that the journal article made mention of comparing deep learning models as well as pulling in more data than what seems to be used in the sample code.
* Converted sample code to .py and opened in Spyder to start a line-by-line analysis. This involved deliberately breaking things to determine how they worked. For example, caseids was considerably shorter than the bounds on tqdm (probably having been copied from the literature on vitalDB) and threw an error when a case out of bounds was randomly selected.
* From there, I let the figures and print() statements guide my walk through the code.
* Made note that many of the processes used in the code could be cleaned up in a function.
* I was and still am suspicious of the accuracy ratings. Often times the model would report 90%+ accuracy, but the waveform would be almost flat compared to the true ABP. I suspect that including Bland-Altman plots as well as max/min like in the journal article would give a clearer picture.
* Overall, the sample code is a great skeleton to hang the project on.

Part Two:

* Since I can build a function, but have less experience with OOP I turned to Claude for help with refactoring the code. The project seems to naturally build one block on top of the other, so I thought builder would be the best pattern and tried to recreate that for part two. I knew that with my lack of experience here that I would be wise to lean on an assistant and chose Claude.
* I began by gathering all of the important variables in the project into a config file and began organizing essential processes (testing, segmenting, etc) into functions.
* Claude was immensely helpful at tying these pieces together and cleaning them up into a proper project where the pieces could contact each other. I was able to take my file with my piecemeal functions and Claude could split them into main and the model to complement my config file.
* For this section of the challenge, I made no major changes other than implementing OOP. The scheme is otherwise identical save for changes to avoid errors.

Part Three:

* My goal for part three was three fold: (a) implement systolic and diastolic BP as well as mean BP, (b) try bi-directional LSTM because my research indicated that might be an improvement, (c) include Bland-Altman like in the original journal article.
* My goal in all of the above was to improve accuracy. My thought being that the mean blood pressure only would erase important peaks and valleys (ie a patient holding steady at 120/80 and a patient having a crisis at 150/50 would both have the same mean); a BDLSTM would be able to capture future cardiac signals and trace them back to a blood pressure at a point in time; and Bland-Altman may tell us more about the underlying weaknesses of the model.
* I was partially successful in these endeavors. Leaning on Claude again, I was able to sketch out the functions, turn them over to Claude for some crucial help formatting, then repeat them for error elimination.
* As a side note, I have found this three-segmented cycle to be the most effective generally regardless of LLM: I clean up the code as much as possible to limit the potential for hallucinations, LLM comes in to do crucial work in areas of lack of expertise or time, then I bring the code in and troubleshoot. That seems to be the quickest and most painless way to include and LLM in my experience.
* If I had another work day to devote to this project, I would successfully implement the Bland-Altman plots as well as make the code more dynamic. Ideally the zoomed-in views would be a randomly chosen segment; the windows used for downsampling would be based on heart rate to grab complete cardiac cycles, and the max ranges would be dependent on the size of the sampling arrays coming through to name a few things. That and adding more polish to the project.

Overall, I have put together a modular and expandable project that can be easily polished and moved from the 80% line to the 100% with another workday or two. It would be interesting to see this project include other bio markers available on vitaldb like fem art bp etc. There are potentially many invasive measures that could be replaced. It would also be worth expanding the number of samples and methods of deep learning with more time.

Full conversations with Claude are available, but they are too long to reproduce here outside of brief summary and may be presented on request.