## LIDP Behavioral Questions

1. Tell me about a time when you faced a difficult technical problem. How did you approach it, and what was the outcome?
   1. In a recent project, I was using a script my team member wrote that extracts data from multiple csv files, joins them together, and performs complex calculations to generate a summary report. While looking at the outputs, I noticed that there were some runs that seemed unreasonable while others seemed to be fine. It was very inconsistent and the code was heavily nested in for loops.  
        
      To tackle the problem, I broke the code into pieces and debugged each section carefully using logger statements. I also created more test cases to try and see if there was a pattern. Eventually, I pinpointed the problem to a combination of regex pattern issues and a missing “continue” statement in a for loop (ended up being year and month switched and some files were being read twice for different states). Once I found the bug, I proposed and implemented a solution, added more test cases, and submitted the merge request.
2. Describe a situation where you had to work on a challenging project with tight deadlines. How did you prioritize tasks and ensure successful completion?
   1. As a pricing actuary, many of my projects have strict deadlines due to regulatory restrictions, business demands, resource availability, and other concerns. Since I am also the most senior member on the actuarial pricing team, I am in charge of leading our bi-quarterly pricing cycles as well, which include data pulling/cleaning, analysis, development, and deployments.  
        
      As we’re nearing the launch of our new rating platform application, things have been getting more hectic with high priority projects and competing deadlines. On top of all my usual projects, I’ve also proposed a challenging project of my own to build a pipeline to process the new outputs and generate reports.  
        
      One thing I’ve learned from having worked on all these projects is that communication and proactiveness are vital to delivering successfully. To achieve this, I try to break every project into smaller deliverables and maintain clear communication with stakeholders so that the moment any deadline is in jeopardy, we may act upon it quickly and early. I use a variety of tools such as JIRA and Teams Tasks, depending on the scope and who I’m working with to make sure the projects are on track.  
        
      Through effective prioritization and team coordination, I was successful in delivering on my data pipeline project, completing my analysis work to support our pricing release, and leading and deploying the pricing release.
3. Can you provide an example of a time when you had to collaborate with team members from different disciplines or departments? How did you ensure effective communication and cooperation? (Employee Address Implementation -> Territory Analysis)
   1. I’m pretty fortunate in that because I’m so familiar with our pricing application, databases, and business knowledge, I get to work with so many talented folks from all areas such as our Underwriting teams, Product teams, Development teams, and IT.  
        
      Recently, we had a cross-functional project to implement a new pricing feature for our rating application. This required collaboration with multiple teams including Reserving, Pricing, Underwriting, IT, and Product. To ensure effective communication and cooperation, we had frequent standups to align on goals, expectation, concerns, and deliverables. We used tools such as JIRA and Microsoft Teams for content sharing and progress tracking. By listening to everyone’s input and addressing their concerns, we were able to release a successful product that adjusted our pricing more accurately and fairly.
4. Talk about a time when you disagreed with a colleague or team member about the best approach to a problem. How did you handle the situation and come to a resolution?
   1. My recent project involved creating a data pipeline to process batches of outputs from our rating application. I went over it with my actuarial team who initially skeptical of my design choices.   
        
      Most of the code that is maintained by this team has been straightforward scripts. Though this design worked well for a variety of tasks, I saw that things were starting to be inefficient due to the need for code reuse and slightly varied uses. My approach was to utilize the factory pattern in order to provide flexibility and reuse as much code as possible, minimizing mistakes made in the code.  
        
      To make sure everyone was on board (since ultimately we’d all be using and maintaining this code), I started with a few introductory training meetings on object oriented programming and the need for design patterns (as much as I understood them anyways). Armed with the shared knowledge, we then held several meetings discussing our concerns and collaborated on different approaches to solve them.   
        
      We ended up with a mixture of tactics that resulted in code that was not only easy to maintain, but was flexible and extremely efficient.
5. Describe a situation where you had to quickly learn a new technology or programming language to complete a project. How did you approach the learning process, and what was the outcome?
   1. I’m not going to lie.. When I first started my current position, I didn’t know very much about programming. My coursework included a few required computer science courses, but I was mainly interested in solving mathematical and statistical problems at the time.   
        
      Once I actually started working at ICW however, it became quickly apparent that I needed to up my programming game. There were a lot of processes that involved manual work with low tolerances for errors. The most important of which involved pulling data from various sources, joining them together, and performing complex calculations to form a report for management to consume. These processes would take weeks to complete due to resource bottlenecking and data validation.  
        
      Because of our small team and limited resources, I decided to automate and distribute the work as much as I could. I taught myself Microsoft VBA in my freetime after work hours and on weekends and built automated workflows and tools for my team to use. (I chose VBA due to its approachability and integration with our excel data files).  
        
      Eventually, over the course of a few months of just working there, I automated the entire workflow and brought down the report generation to just a few hours.
   2. When my company decided to re-platform our pricing application from a SOAP API based in excel to a Rest API using Java Spring Boot I had my work cut out for me. I was tasked to help write out some of the business logic for a handful of APIs and connect them to a few MSSQL databases.  
        
      I approached the learning process by first studying documentations, watching as many online tutorials as I could, and reading relevant materials. I also created small personal projects to experiment with Spring Boot and gain more experience with different kinds of projects and their approaches. I’ve also asked for advice from other engineers when I was completely stuck.  
        
      Through self-guided learning, and some discussions with my team, I was able to quickly get up to speed with the framework and implement the business logic I was working on. The outcome was efficient and clean APIs that received positive feedback from my team and the end-users.
6. Can you share an example of a time when you encountered a setback or failure in a project? How did you handle it, and what did you learn from the experience?
   1. Earlier on in my position, I was tasked with removing a feature from our pricing application. A few months later after its removal, it was discovered that we were non-compliant and needed to add the feature back in. Due to the application, its vcs, and the time lag, this was a much more complicated task. I ended up missing a key value, and caused mispricing in specific circumstances.  
        
      From this experience, I’ve learned valuable lessons about the importance of testing meticulously and implementing a thorough review system. I spoke to my manager about this incident and together, we put together a process of peer review and having thorough validations.
7. Tell me about a project where you had to deal with changing or ambiguous requirements. How did you adapt and ensure successful delivery?
   1. CEO vague about request to conduct a territory analysis and suggest pricing adjustment factors based on industry data. Left a lot of details out. Territory Analysis.  
      Went back and forth with manager. Researched topic and read about how others might approach it.   
      Eventually created our own methodology. CEO loved it.
8. Describe a situation where you had to communicate technical concepts or complex ideas to non-technical stakeholders. How did you ensure effective communication and understanding?
   1. Dept was very much into cross training. We encouraged lunch and learns. I put together series on oop. Used relatable ideas and comparisons. Created hw to help reinforce. Asked open ended questions.
   2. If no time, then start with big concepts. Drill down into details as necessary.
9. Talk about a time when you had to resolve a conflict within your team. How did you approach the situation, and what steps did you take to find a resolution?
10. Can you provide an example of a time when you took the initiative to improve a process or implement a new solution? What was the outcome, and how did it benefit the team or organization?

## About LIDP

1. Formed in 1979 to provide consulting services and solutions to life insurance industry.
2. Titanium is a policy admin system that replaced the old Cobol based Administrator. It offers hundreds of RESTful APIs to help insurance streamline their operations across the entire policy life cycle, from product configuration to claims processing to everything in between.
3. The tech is cloud native and single tenant, allowing horizontal scaling with cost savings. By 2023, LIDP has set a goal to build it entirely on open source tech. This cuts licensing costs.
4. “..actuaries can experiment with policy features and pricing and test new life insurance products, define the underwriting requirements, build the application, and bring the product to market within just a few weeks”
5. Titanium goes beyond a mere admin system. It offers plenty of services that can address multiple needs of customers.
6. “People who work here are like family”

## Questions for LIDP

1. How is LIDP leveraging the rise of AI technologies to service its customers?
2. Insurance industry is heavily regulated. I read about how Titanium is now cloud native. How does LIDP ensure compliancy with feature implementations for its clients? i.e. data privacy, cloud db restrictions, etc? How closely do SWEs work with Legal? Or what is the review process like?
3. How would you describe the ideal person to fulfil this role?
4. What are the top priorities for this role?
5. What tech stack would this role use?
6. I saw an article mentioned that LIDP had set a goal by 2023 to build Titanium to solely use open source tech. Has this been achieved? What technologies are still transitioning to open source?
7. What learning resources are available for employees to use to learn new skills?
8. What operating systems are work machines offered in? Laptop specs? Admin rights?
9. Has the legacy system Administrator been completely retired?
10. Looking at LinkedIn, most team members seem to be SWEs. Are there any data analytics teams? Data science? Analysts? Or SWEs serving both roles?