ZipHR Backend Developer Task

ZipAirlines

Thank you for taking part in our technical interview here at ZipHR. The goal of this interview assignment is to give you an opportunity to demonstrate your software engineering skills, especially around code organization, data structure use, problem solving and testing.

Problem

Your assignment is to create and submit a working digital version of the aircraft passenger capacity issue, ZipAirlines, along with instructions for the interviewer to run the code. Please use Python to create the solution. More detailed requirements are listed below.

Timeline

Please complete the task within 1 week upon receiving this assignment to make a submission.

Please respond to this assignment email with a date you plan to submit the assignment, and suggest a 1-hour time slot for the following day that you would be able to attend an online pair coding session. Your assignment will be reviewed within 24 hours of your submission.

Requirements

We are creating a new software for a airline company called “ZipAirlines”.

- The company is assessing 10 different airplanes.
- Each airplane has a fuel tank of (200 liters * id of the airplane) capacity. For example, if the airplane id = 2, the fuel tank capacity is 2*200 = 400 liters.
- The airplane fuel consumption per minute is the logarithm of the airplane id multiplied by 0.80 liters.
- Each passenger will increase fuel consumption for additional 0.002 liters per minute.

Write a RESTful API using Django Rest Framework to:

- Allow for input of 10 airplanes with user defined id and passenger assumptions
- Print total airplane fuel consumption per minute and maximum minutes able to fly

Code should

While implementing the task, please consider the following.

Code should:

- Be written with Python 3 (*include type hints syntax*).
- Come with instructions how to use and run it.
- Come with tests to verify the functionality.
- Use Coverage to show good amount of code coverage after running tests.
- Be stored in some versioning system (*we prefer and work with git*).
- Be able to extend and new features when required.

Evaluation

We are most interested in learning about your thought process and coding style and expect this to be displayed within a final result of the task.

The follow-up paired coding session will allow you to discuss your solution and challenge you to implement additional features as quickly as possible.

Your solution will be evaluated on the following:

- Appropriate use of software design principles.
- Proficiency with Python.
- Correctness: How do you verify the correctness of the solution?
- Maintainability: How easy is the solution to understand and modify?
- Extensibility: How easily could the solution be adapted to specification changes?
- Performance: What is the algorithmic complexity of the solution?
- Completeness: How much of the specification does the solution implement?

Remember to have fun with the problem and thanks, again, for coming
this far with us. Good luck! We're looking forward to seeing your solution to the task.