# Full Fibre Back-End Developer Technical Test

Please answer all of the following questions, you may submit answers either in this document or as external files (e.g. .sql files for questions requiring an SQL query, or .py files for Python questions)

1. Using the Entity Relationship Diagram in Fig. 1 (below,) write a query which returns the appointment details, owner’s contact details, and pet’s details for all scheduled Nurse appointments at the Exeter practice between July 5th 2021 and July 11th 2021 (inclusive) in ascending chronological order. Data for some tables in this ERD can be found below.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **practice** | | | | | |
| uid | name | telephone | manager | address | branch |
| 1 | Plymouth Vets | 01752 000000 | 2 | … | Plymouth |
| 2 | Exeter Vets | 01392 000000 | 1 | … | Exeter |
| … | … | … | … | … | … |

|  |  |
| --- | --- |
| **employee\_types** | |
| uid | type |
| 1 | Veterinarian |
| 2 | Nurse |
| … | … |

|  |  |  |
| --- | --- | --- |
| **appointment\_timeslots** | | |
| uid | timeslot\_start | timeslot\_end |
| 1 | 09:00:00 | 09:30:00 |
| 2 | 09:30:00 | 10:00:00 |
| … | … | … |

1. What improvements, if any, can be made to the ERD in Fig 1?

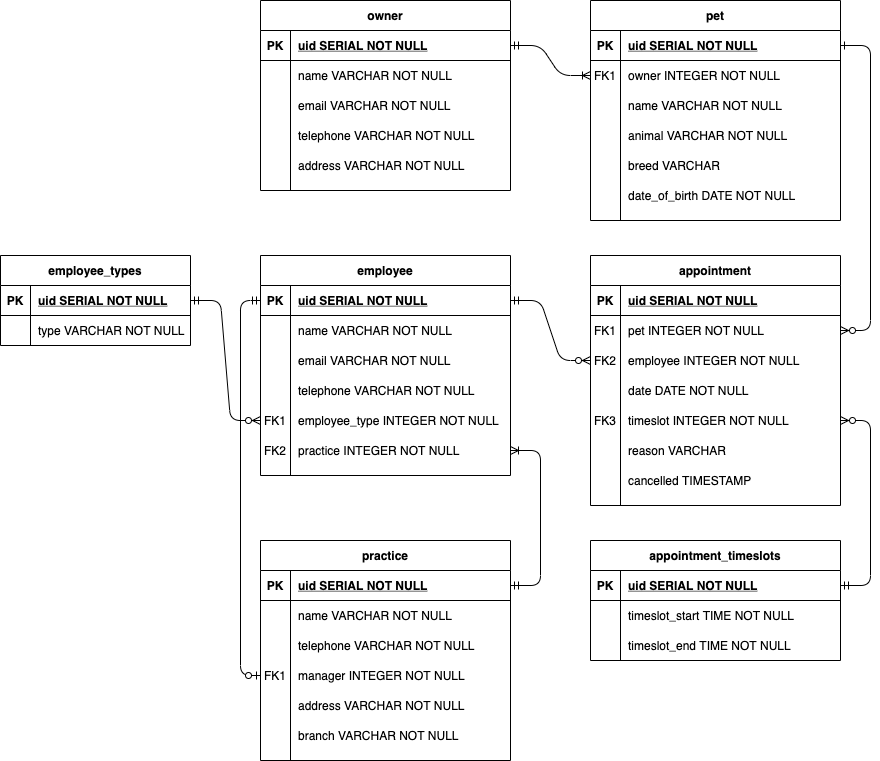


Fig 1. Entity Relationship Diagram

1. Using the template Flask application from this Git repository (<https://github.com/Full-Fibre/technical-test>), create endpoints to handle CRUD functionality for appointments using the ERD from Fig 1. Instructions on how to setup and use this template can be found in the README.md file within.  
     
   Please provide a git repository link below as your submission. We will need to be able to pull down the code so that we can test it.