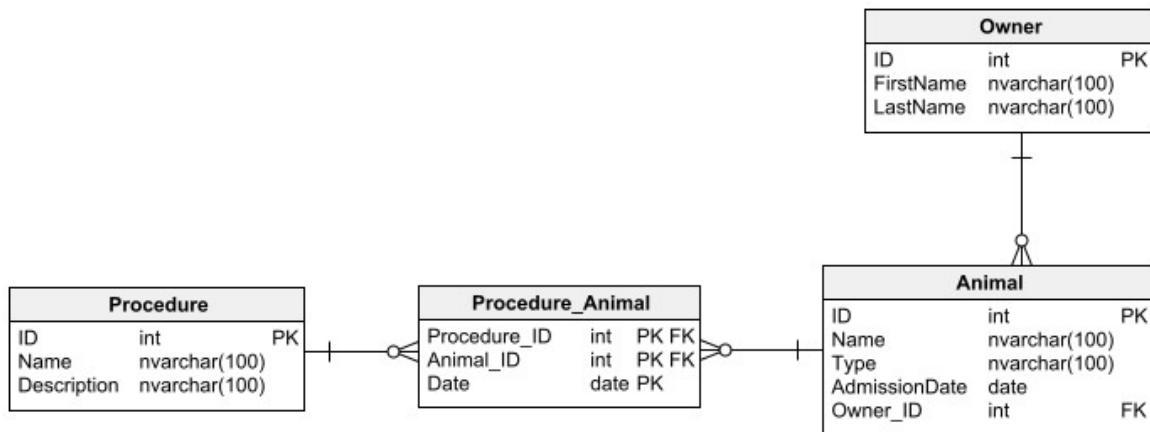


# Przykładowe kolokwium 1

W niniejszym zadaniu musicie zaprojektować aplikację web API, która będzie obsługiwać bazę danych zaprezentowaną na diagramie. Tworzymy serwis, który powinien pozwolić na obsługę lokalnej kliniki dla zwierząt.

## Baza danych

Baza, którą wykorzystujemy zaprezentowana jest poniżej. Skrypt tworzący tabele znajduje się w pliku example-test.sql.



## Końcówki

Zaprojektuj końcówkę, która będzie zwracać informację na temat zwierzęcia o podanym ID. Wraz z danymi zwierzęcia, powinny też być zwrócone dane na temat właściciela oraz wszystkie procedury związane ze zwierzęciem. Jeśli nie ma zwierzęcia o podanym identyfikatorze, powinien zostać zwrócony odpowiedni kod.

Końcówka powinna reagować na zapytanie pod adres `api/animals` np. HTTP GET <http://localhost:5000/api/animals/1>

Przykładowe dane jakie zostaną zwrócone:

```
{
  "id": 1,
  "name": "Wanwan",
  "type": "Dog",
  "admissionDate": "2024-04-19T00:00:00",
  "owner": {
    "id": 1,
    "firstName": "Jan",
    "lastName": "Kowalski"
  },
  "procedures": [
    {
      "name": "Name1",
      "description": "Desc1",
      "date": "2024-04-21T00:00:00"
    },
    {
      "name": "Name2",
      "description": "Desc2",
      "date": "2024-04-21T00:00:00"
    },
    {
      "name": "Name3",
      "description": "Desc3",
      "date": "2024-04-21T00:00:00"
    }
  ]
}
```

Zaprojektuj końcówkę, która pozwoli nam dodać zwierzę do bazy danych. Kończówka powinna pozwolić na przyjęcie zarówno danych zwierzęcia, jak również opcjonalnie listy zabiegów jakim zostało poddane. W takim wypadku chcielibyśmy w ramach jednej transakcji dodać zarówno zwierzę, jak również wpisy do tabeli asocjacyjnej. Pamiętaj o wykonaniu opisanych czynności w ramach jednej transakcji. W bazie danych wszystkie pola są wymagane. Należy sprawdzić, czy właściciel oraz procedury istnieją w bazie.

Kończówka powinna reagować na zapytanie pod adres `api/animals/` np.

HTTP POST <http://localhost:5000/api/animals/>

Przykładowe dane jakie są przesłane wraz z żądaniem:

```
{
  "name": "AnimalName",
  "type": "AnimalType",
  "admissionDate": "2024-04-29",
  "ownerId": 2,
  "procedures": [
    {
      "procedureId": 1,
      "date": "2024-04-18"
    },
    {
      "procedureId": 3,
      "date": "2024-04-19"
    }
  ]
}
```

## Punktacja

Każda końcówka jest wyliczana na **10 punktów**. Czyli za całość można zdobyć **20 punktów**.

Punkty mogą być odjęte za:

- Niepoprawne lub nieczytelne nazwy zmiennych: **-2 pkt**
- Niepoprawne kody HTTP: **-4 pkt**
- Brak wydzielenia komunikacji bazy danych do oddzielnego pliku/serwisu/repozytorium: **-10 pkt**
- Brak wstrzykiwania zależności: **-8 pkt**

## Uwagi

- Program, który się nie kompiluje - **0 pkt**
- Program, który jest plagiatem - **2 na koniec**