**Dog Recommendation System**

**ooo, 박정명, ooo**

**1. Introduction**

Abandonment of dogs are a growing concern in Korea. In fact, millions of dogs are abandoned around the world, which is both problematic in terms of animal welfare.[[1]](#footnote-1) According to the <유실유기동물분석> report released by the Korean Animal Welfare Association, 38.8% of pure breed dogs in adoption centers were adopted in 2020, and 38.5% in 2021. However, out of these adopted dogs, 41.3% of adopted dogs were returned to adoption centers in 2020 and 45% in 2021.[[2]](#footnote-2) In worse cases, dogs are abandoned, refueling the problem of stray dogs. Some people adopt dogs based on their physical appearance only, with limited considerations for the differences in specific traits of breeds.[[3]](#footnote-3) Consequently, people choose breeds that are incompatible with their lifestyle, socioeconomic status, and home environment. For example, according to a news article from Press Kookmin, it states that the owners of dogs are unaware of the financial costs related to raising a dog, leading to have to give up ownership of their dog.[[4]](#footnote-4) Thus, there is a growing need to improve transparency about the costs and the specific traits of dog breeds to guide people when adopting dogs. Based on data compiled from various sources, we designed a comprehensive recommendation system that helps users adopt a dog that is compatible with their needs. We hope to implement this system to help prevent dogs from being abandoned.

**2. Scenario**

**2.1 Functions**

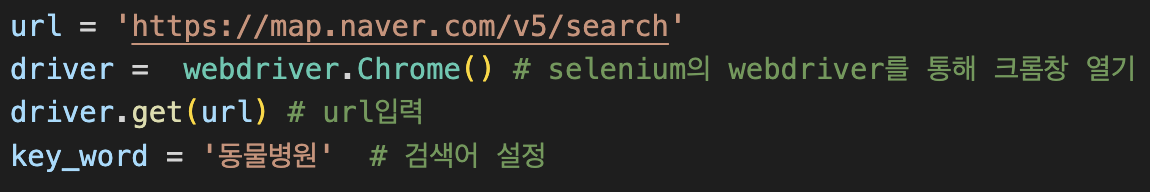
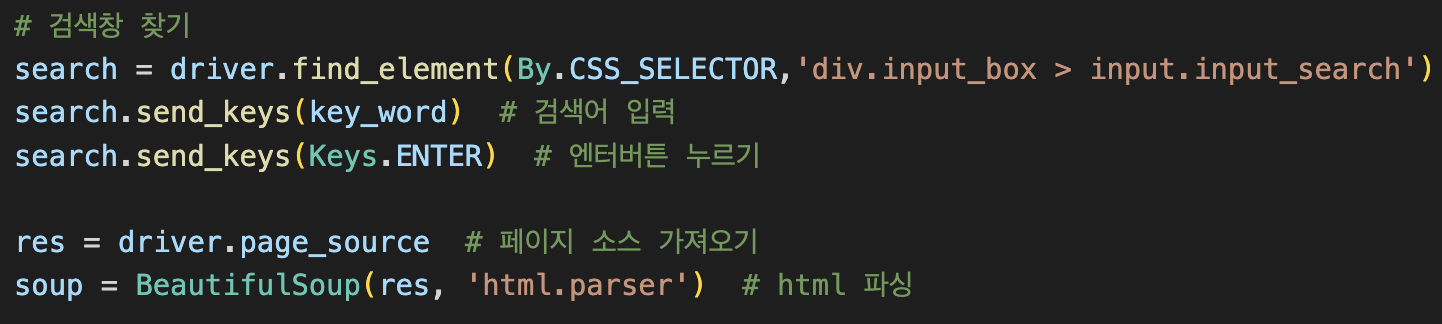
We support a wide range of functions that allow users to flexibly use our services. Our main target users include 1) **those that want to adopt dogs** and 2) **those that already have dogs and want to improve the care for their dogs**

|  |  |
| --- | --- |
| Users who want to adopt a dog: | Users who already have dogs |
| 1. Fill out a questionnaire to filter dog breeds that are compatible to their lifestyle, socioeconomic status, and home environment. This is so that users know which breeds to look for when adopting a dog 2. Choose pictures from a list of adoptable dogs. The inputs are fed into our Neo4J recommendation system and return adoptable dogs that look similar to that of the dogs that were chosen. 3. Users choose the dog that they like and they are matched with the adoption center to adopt the dog. 4. Provide user with information about dog care | 1. Input information about dog breed to find the **following information** about their breed:  * The group of the breed * The size of the breed * The life expectancy of the breed * Common illnesses of the breed * How friendly the breed is to strangers and pets * Lifetime cost of breed * How friendly the breed is with kids * The amount of grooming needed * The amount of shedding of the breed * The energy level of the breed * The demeanor of the breed * The trainability of the breed * The intelligence of the breed * Temperament of the breed  1. Users can find information about the nearest animal hospitals and their reviews |

**2.1 Requirements / Assumptions**

**2.1.1 Collecting Data**

 We collected data using packages Selenium and BeautifulSoup. Selenium was used to control the page of Naver Place, a dynamic website. On the other hand, BeautifulSoup framework was used for html parsing. Using these frameworks, we implemented button clicks on web pages, frame transitions, and search word input.

****

Also, We used ‘css\_collector’ to select the specific section of the webpage and the ‘find\_element’ function to extract the section information.



**2.1.2 Data Preprocessing**

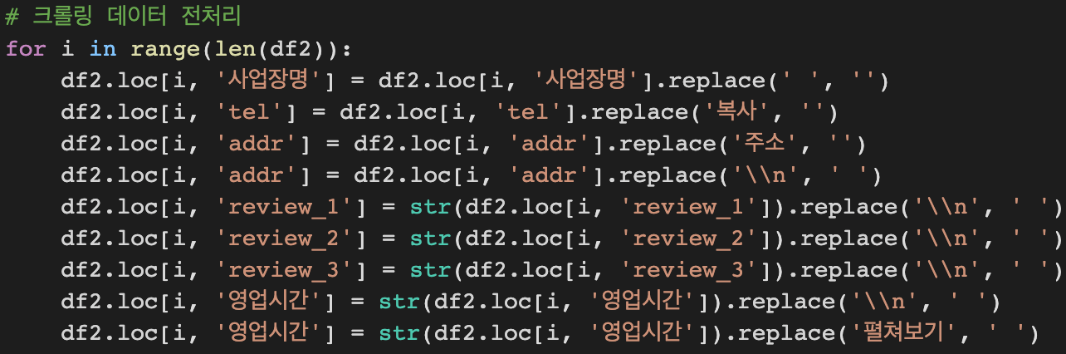
**2.1.2A Dog Breed Dataset Preprocessing**

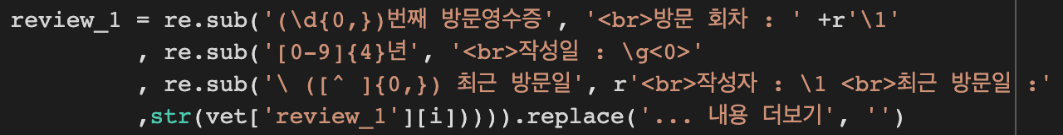
In our data set with dog breed information, there were some variables that had NAs. We imputed missing values for some variables using KNN. For example, as shown below, we imputed values for the size category of dogs in the dog breed dataset. For variables that lacked a clear relationship with other variables, we left as NA values to prevent reporting of false information to users.

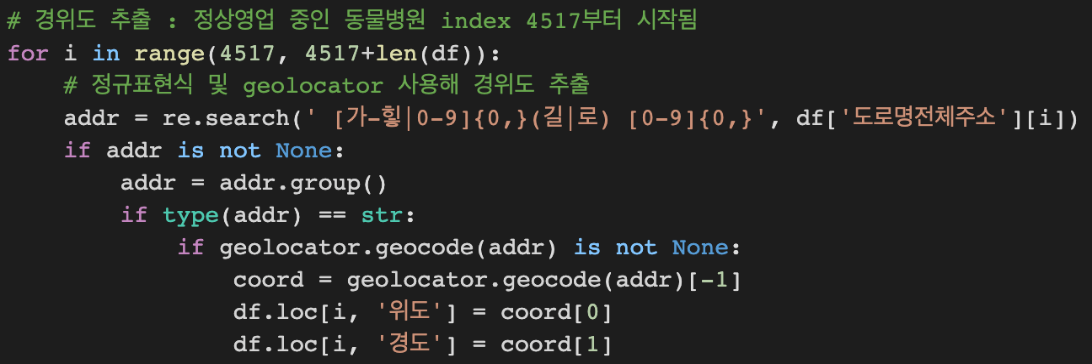
|  |  |
| --- | --- |
| https://lh3.googleusercontent.com/v9FpN09jdEthBu7b-tvN_vD805hZnIk91n2Z0nNeUFQ6G-tbVMBitWTDHUQQyOOkO557zfao8rW5w3k4hGcPoDqwLyqzWGiDmQbARM5qfG2-7JwVtPjveB_gbvVSowPZ8p3Oi6EjRoBRQr0gPdTNkgShGXjEMccC3E2qoZXoI4hEie8XlT-Cy3qzxRKNPA | https://lh3.googleusercontent.com/dLFEiN10vAQLwhAptWse_jUGXvW77n9xcFZtpBMqrVWmi1oODzkdzWcUmgo65Cst3TE9vb8brGmO2ZYPUcBMvbp3RCmEX0r8dlrRWSd2W3xzevJqVdHK2wS6b1Jg4EN4SjjXYjq856Ou5piUk2dPyGEyv_REOaigJKueSTgkVCA41PRmVKTb3RRi55i8sg |

**2.1.2B Hospital Review Dataset Preprocessing**

In addition, we also performed pre-processing of the crawled data. For text data processing, 'regular expression' and 're' were mainly used. Meanwhile, we used 'geolocator' to extract the longitude and latitude to be used in the map function.







**2.1.2C Hospital Review Dataset Preprocessing**

We also performed pre-processing for the adoption center dog dataset. There were too many columns and some special characters in dog’s name. The description variable was too long so it was not used.  


**2.1.3 Query Processing**

|  |  |
| --- | --- |
| **https://lh4.googleusercontent.com/SBrwJHEBHJEl0KI5ajlGCv-oNUsafe39u2kqORHMeppIYTBhdEKTmF9oANBGvJ_eEWH2HN0UGIg9VGKa0ZV5jLvGTCfZng7iNXp7SbxxE3lJuUhUphgeTQAUAQrM5iRtuDN3mB9Y9FxOXuYDI7C7TrmGc5iZacRvfW0Mo89K3DXJavASdrUp6CNRUEj8wQ** | We used psycopg2 to connect python to SQL for query processing. |

**2.2 Dataset**

**2.2.1 dog\_breed.csv**

****

The above table shows metadata of the dog breeds CSV file. The variables in green (query) variables were used to query for dog breeds given user questionnaire input. The yellow (display) variables were shown to users about the dog breed. This includes variables like size, energy, trainability, temperament, intelligence category, and health data like common disease, etc. The red (visualize) variables were used for data visualization. These visualizations are used for easy comparisons between different breeds that the users are interested in. These variables were sufficient to display the information about breeds as listed in the Section 2.1. In Section 2.3, it will be explained how these variables were used in building our system.

**2.2.2** **dog\_adopt.csv**

|  |  |
| --- | --- |
| https://lh6.googleusercontent.com/fH73aJE3yJiSvIMJlrU1kAl20vML06nOGh1ThnydClNOK8k0xS57R8ogoQSuu-UZ4xbNHKQqjXDjWY2ElDW_Xhbw1bggS7hBbD60o4qyvfgNl7mcvepdFs1eNdPxfTxH5sBfQp_ZhA3jOS-HC6k0_-oAIiNiFSyDL0CvQmm-Y41K2ndbAcl3r5o2d9AZrA | The table on the left is the metadata for adopted dogs. To note, the adopted dogs table was derived from Kaggle and is a test data. This is our services doesn't have any real adoption centers that can insert information about their adoptable dogs. These are dogs that are listed for adoption in the USA. The Neo4J variables are in green and are used as nodes for recommendation since they are related to appearance. |

**2.2.3 hospital\_review.csv**

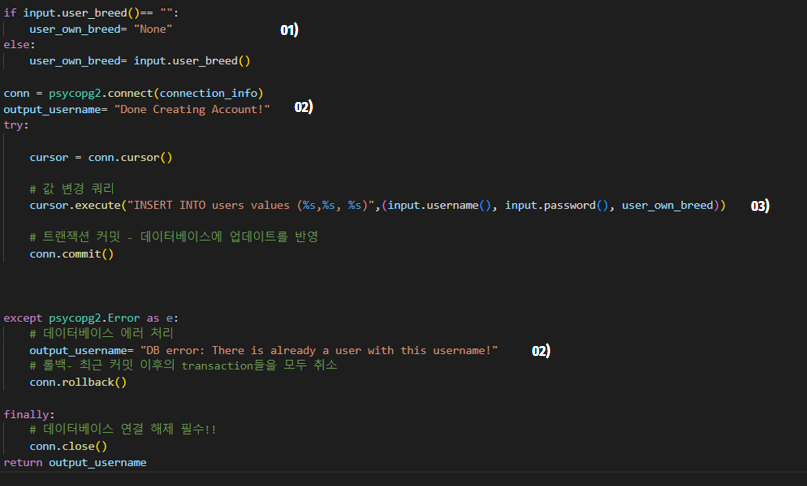
|  |  |
| --- | --- |
| https://lh3.googleusercontent.com/Ouu0VmV3SqHckHbIlwodhQpwL26wByRqoKl645-hi5NAuHyn8-dm0RdeI4zUuhZcoiKINBQBjf-z_Rp7JhVINLj0I9BBDzGw-b7Rb-XTTT2SmfAFGbdWeox2rqCxK1AEoq3pKVjy9444wWwIIayO_EiVAgX-Bp7uveC4J6Ps_wcKBt-IDXpG-hmP28ZO7g | The table to the left is the meta data used for animal hospital reviews. These variables were derived from Naver web-crawling as described in Section 2.1 The green (show) variables are information shown to the user. The yellow (Map) variables were used to plot the location of the hospitals in Folium. |

**2.3 Database workloads**

**There are 4 main database workloads:**

1. Signing up and filling out questionnaire inserts user information into POSTGRESQL
2. Based on user questionnaire input, POSTGRESQL searches for dog breeds that fit user criteria
3. User questionnaire input is updated for user row using POSTGRESQL
4. Information about group breed
5. Neo4J

**2.3.1 Signing Up**



When the user creates an account, they input a username, password, and the breed they own, which gets stored in the database by the code **03)**. If they do not own a breed, the database stores this value as “None” because of the code in **01)**. The username is set as a primary key so cannot have duplicate values. If this integrity constraint is violated, the code from **02)** will inform the user that they have inserted a username that is already being used. The results of the DBMS are shown below Section 2.3.1.

**2.3.2 User Fills Questionnaire**

|  |  |
| --- | --- |
| **https://lh5.googleusercontent.com/gCm0276-ys_kfU4PinMYF-0lCXIu6RQW9RN9liaopon_UQ83YfTzeLP7h_DduK2oUjakj4gq7xpn6t-y3T-LN8qpPIa7xJWbcopCZqJhHtUPFmu5-LT_Ip0nT9IrqAuyFp2Z771TJK7MoNJyKEGUiAlrTPpdXFhFmVZFs3HKm84L8_PRpofjFp3B2QgqVA** |  |

The answers of the users were taken as input and used for querying. The conversion of question input to query input is explained in the table below. The SQL query statement is just for explanation purposes, the actual query used is shown in the picture below

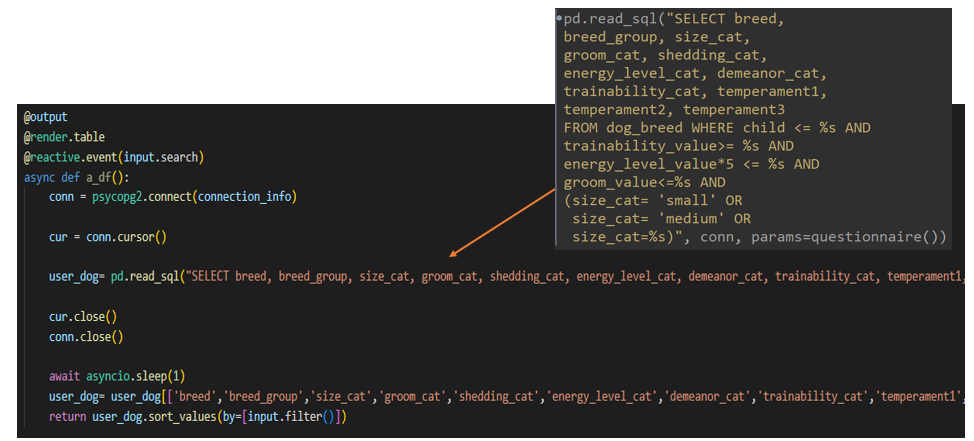
|  |  |  |  |
| --- | --- | --- | --- |
| Question | If Statement Result | | SQL STATEMENT  SELECT \* from dog\_breed… |
| Do you have kids under the age of 10? | Kid | Yes: 2 | where child<= %s |
| No: 3 |
| Are you new to raising dogs? | Exp | Yes: 0.6 | where trainability>= %s |
| No: 0 |
| How energetic are you? | Energy | Energy | where energy\_level\_value\*5 <= %s |
| Does grooming frequency matter for you? | Groom | Yes: 04 | where groom\_value<= %s |
| No: 1 |
| What is your home like? | Size | An apartment: Medium | where size\_cat = ‘small’ or size\_cat= ‘medium’ or size\_cat= %s |
| Else: Large |

To note, the filtering was aimed to provide the broadest range of dog breeds. For the first question, if the user has kids under the age of 10, then the kid variable becomes 3. In the dog\_breed.csv, child has values 1,2, or 3. (1 - highly suitable. 2 - medium suitability. 3 - low suitability). If the user has children, dogs that are highly suitable or medium suitable with kids are shown.

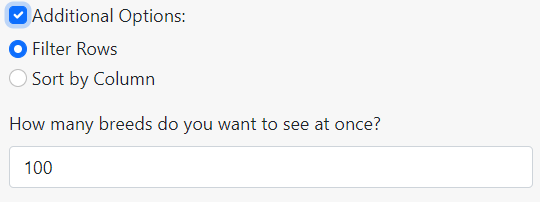
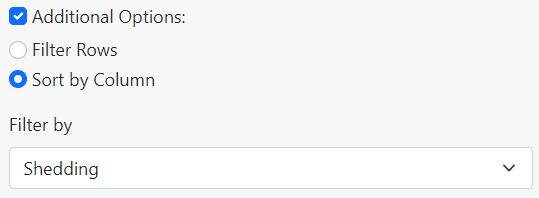
Next, if the users are not experienced in raising dogs then the exp variable is assigned to 0.6, otherwise 0. In the dog\_breed.csv, trainability has values 0.2: Independent, 0.4: May be Stubborn, 0.6: Agreeable, 0.8: Easy Training, 1.0: Eager to Please. So if the user is not experienced with dogs, they would be recommended Agreeable, Easy to Train and Eager to Please dogs.

Next, for the question how energetic are you, it takes in an input of a scale from 1 to 5 and this is set to the energy variable. The energy\_level\_variable in the dog\_breed.csv can have the values: 0.2: Couch Potato, 0.4: Calm, 0.6: Regular Exercise, 0.8: Energetic, 1: Needs Lots of Activity. Someone who has an energy level of 3 will query breed with energy\_level\_values of 0.2, 0.4, 0.6.

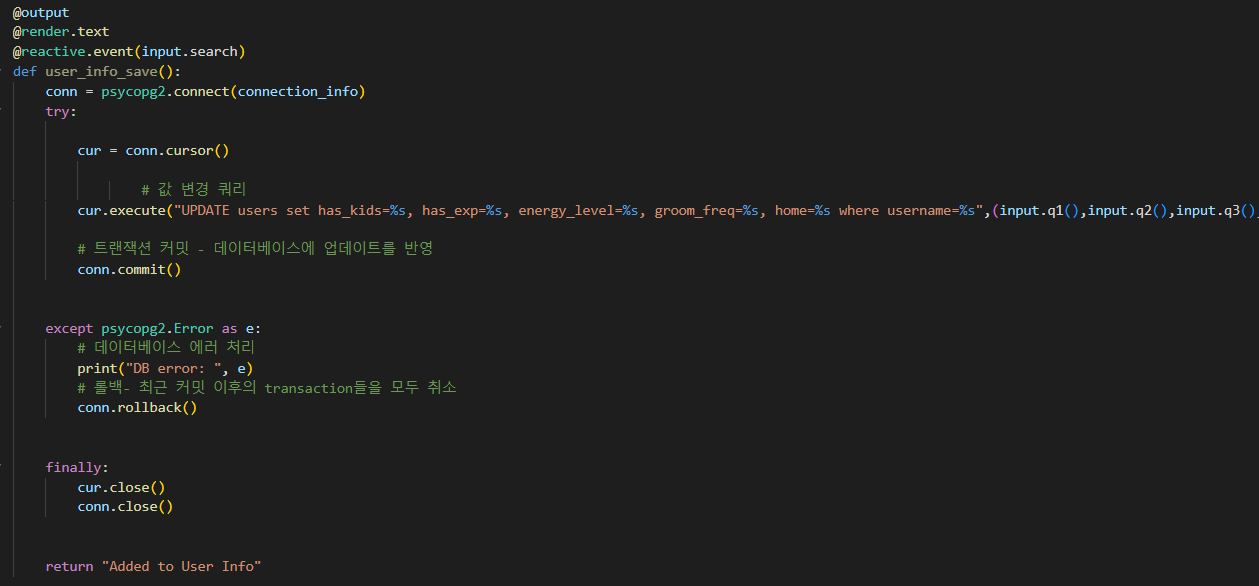
The fourth question is similar to previous questions so the explanation is omitted. For reference, groom\_value can have the values: 0.2: Occasional Bath/Brush, 0.4: Weekly Brushing, 0.6: 2-3 times a week, 0.8-1: Daily Brushing, 1: For the last question What is your home like, if it is an “apartment”, the size variable is set as median. Else, if the user chooses "a home with a small yard” or “a home with a large yard", the size variable is set to large. So if the user lives in an apartment, they are filtered dogs that are small or medium size. If they live the other two types of houses, users query all sized dogs.



Once again, the filtering was performed to maximize the dog breeds outputted, rather than looking for the most relevant dog breed. The user can afterwards look for a specific criterion using the additional option buttons. For example, they may really hate dogs that shed a lot so would sort by shedding value as shown below.

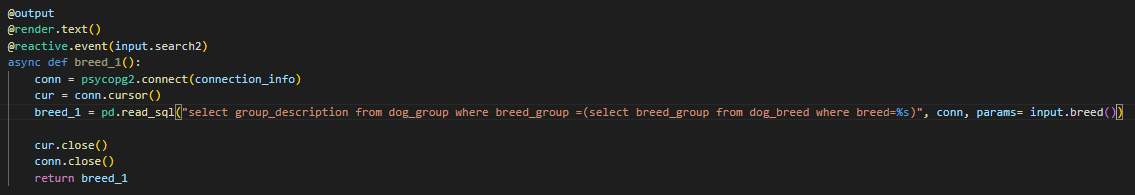
**2.3.3 Questionnaire Answers Are Updated to User’s Rows**



|  |  |
| --- | --- |
|  |  |
| **Section 2.3.1**  User ‘ILOVEDBMS’ signs up. He has no dog currently | **Section 2.3.3**  User ‘ILOVEDBMS’ fills out filter questionnaire. The questionnaire inputs are updated in the database. |

**2.3.4 Information about group breed**

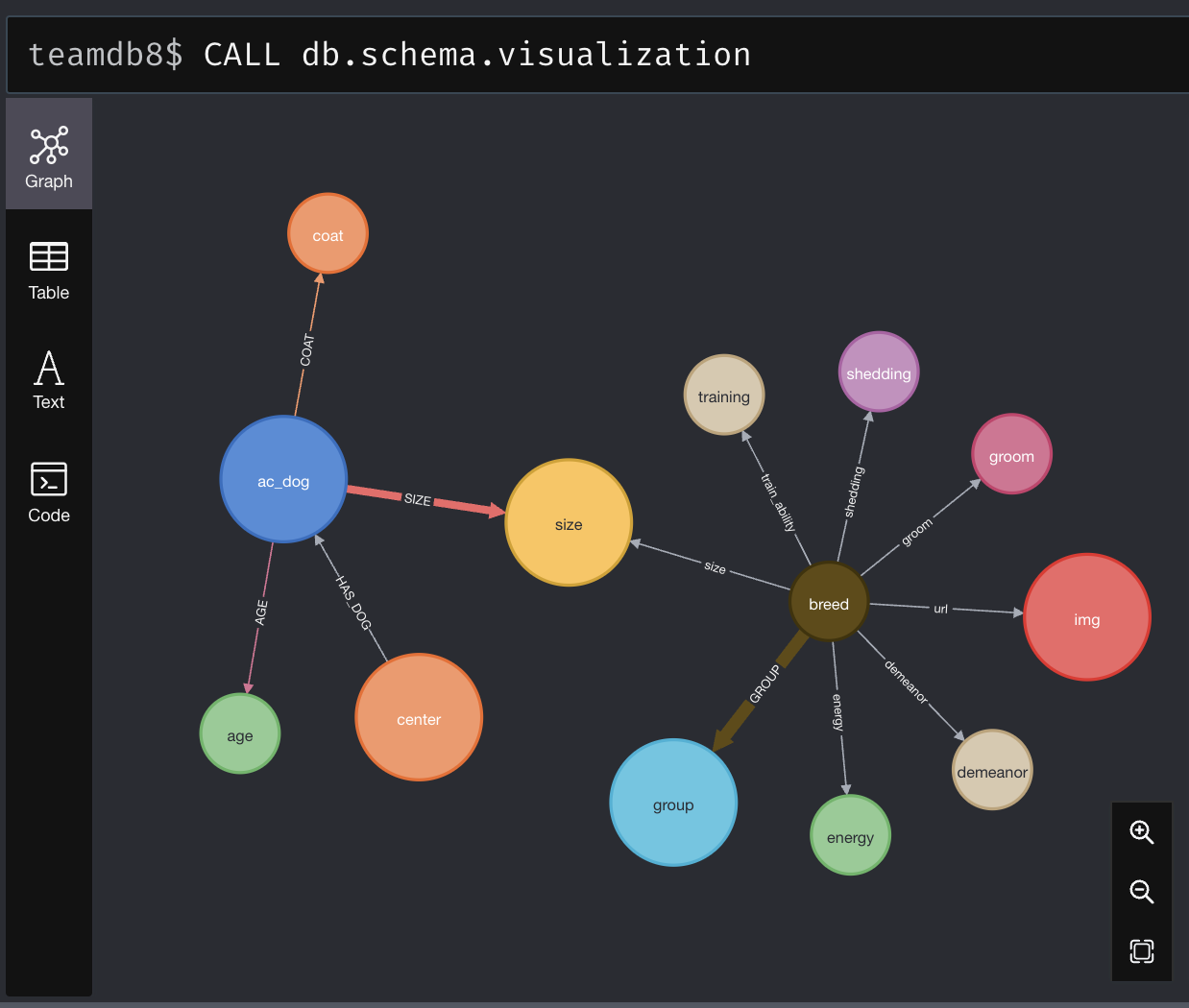




In this query, we use a nested query to return the group of the breed the user inputs. This is then input into the dog\_group table. This then returns dogs in the same group. The results will be shown in the results section.

**2.3.5 Neo4J**

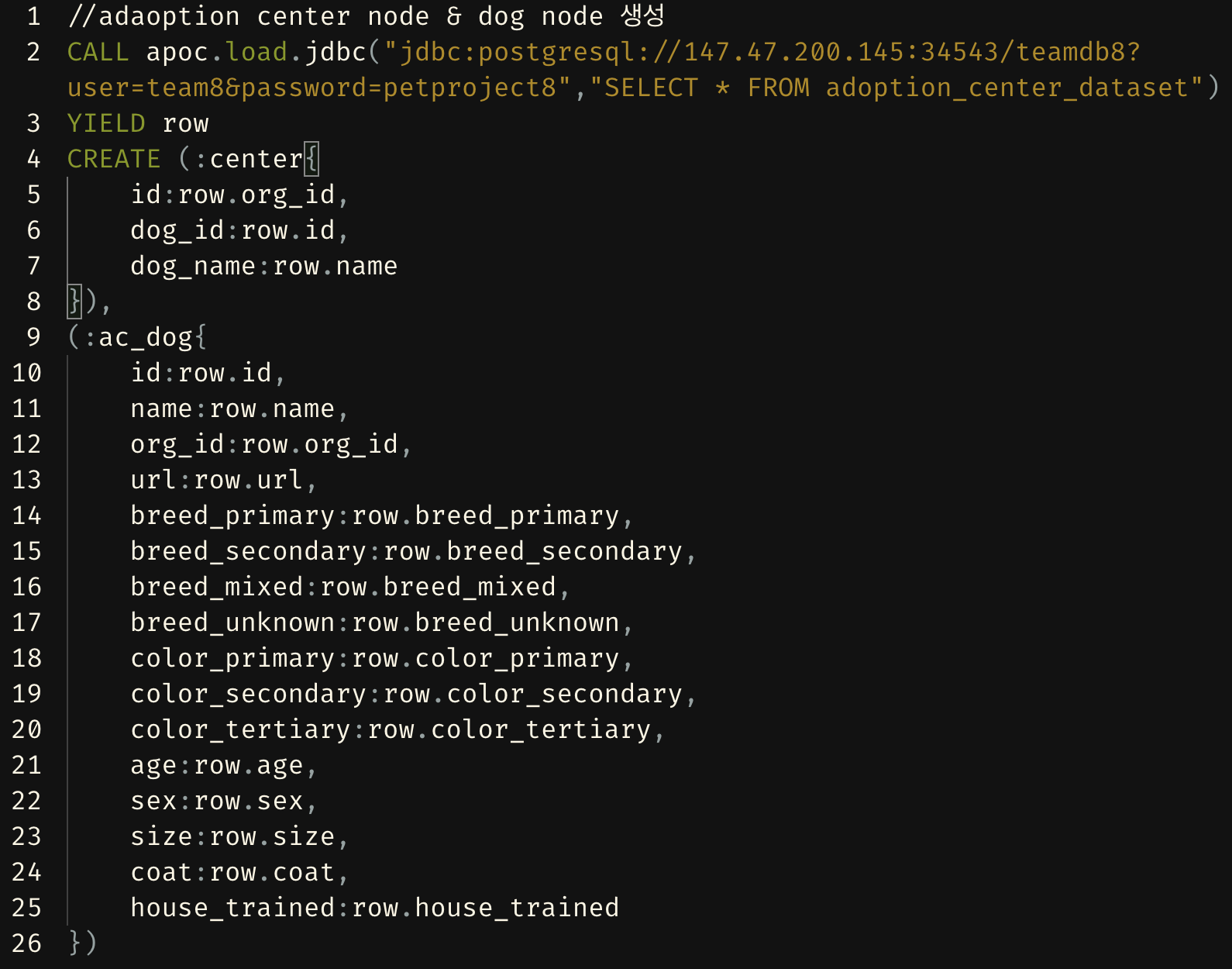
We tried to embed the content-based recommendation system but failed. We already made node and relationships on Neo4j but it was difficult to make the recommendation system.



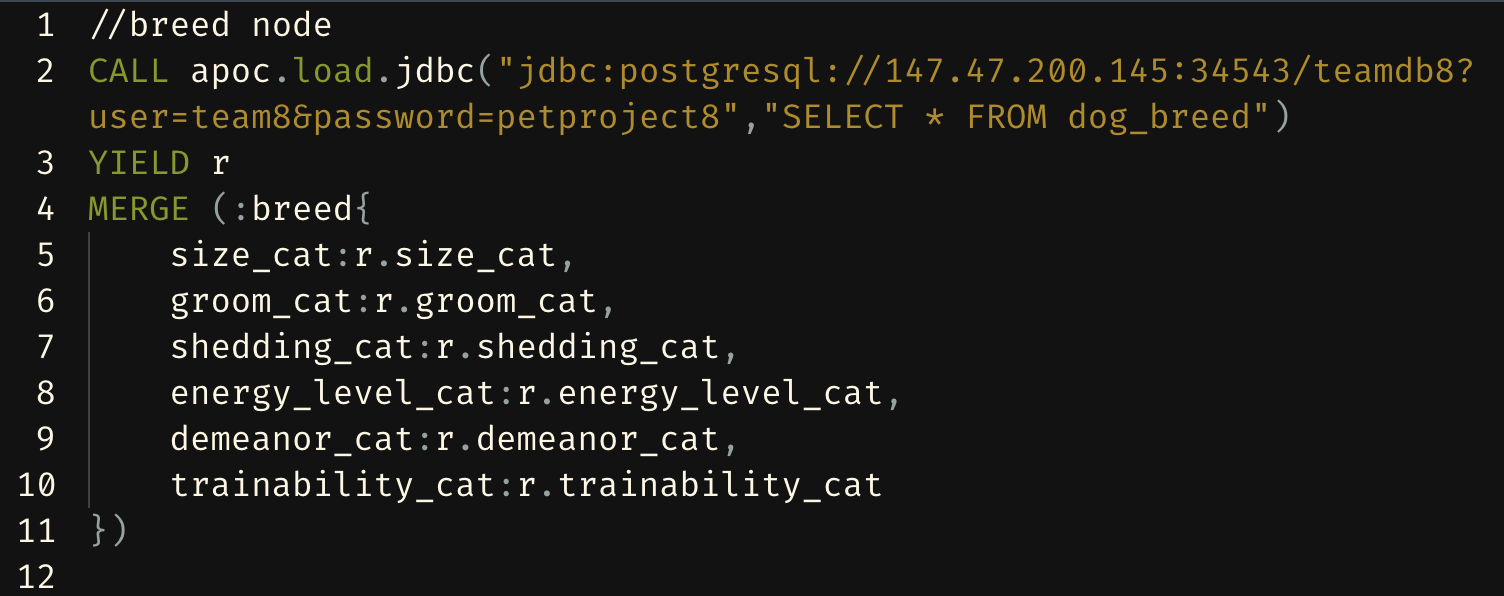
The adoption center dog dataset is very large (58108 rows after pre-processing) so it took a long time and made many errors during the query. Some planning was done through the example syntax, but it was not implemented well. The below query is the example syntax that we planned.

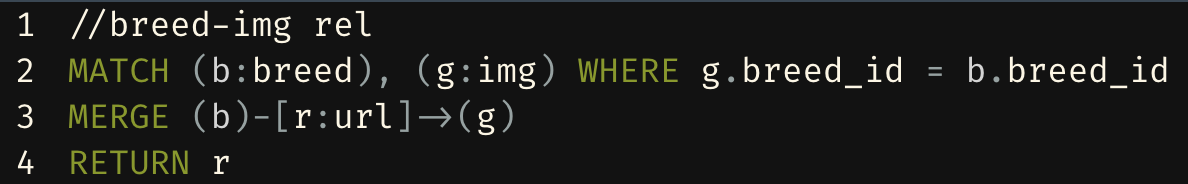
|  |
| --- |
| // Create the Adoption Center node  CREATE (ac:AdoptionCenter {name: "Paws and Claws"})  // Create dog nodes and add their breed, color, coat, and size as properties  CREATE (dog1:Dog {name: "Fido", breed: "Labrador Retriever", color: "Yellow", coat: "Short", size: "Medium"})  CREATE (dog2:Dog {name: "Bella", breed: "Beagle", color: "Brown", coat: "Short", size: "Small"})  CREATE (dog3:Dog {name: "Max", breed: "Boxer", color: "Fawn", coat: "Short", size: "Large"})  CREATE (dog4:Dog {name: "Lucy", breed: "Poodle", color: "Apricot", coat: "Curly", size: "Small"})  CREATE (dog5:Dog {name: "Charlie", breed: "Golden Retriever", color: "Golden", coat: "Medium", size: "Medium"})  // Create relationships between the adoption center and the dogs  CREATE (ac)-[:HAS]->(dog1)  CREATE (ac)-[:HAS]->(dog2)  CREATE (ac)-[:HAS]->(dog3)  CREATE (ac)-[:HAS]->(dog4)  CREATE (ac)-[:HAS]->(dog5)  // Create another Adoption Center node  CREATE (ac2:AdoptionCenter {name: "Bark and Meows"})  // Create more dog nodes and add their breed, color, coat, and size as properties  CREATE (dog6:Dog {name: "Daisy", breed: "Poodle", color: "White", coat: "Curly", size: "Small"})  CREATE (dog7:Dog {name: "Molly", breed: "Beagle", color: "Tricolor", coat: "Short", size: "Small"})  CREATE (dog8:Dog {name: "Rocky", breed: "Bulldog", color: "White", coat: "Short", size: "Medium"})  CREATE (dog9:Dog {name: "Sam", breed: "Labrador Retriever", color: "Yellow", coat: "Short", size: "Medium"})  CREATE (dog10:Dog {name: "Toby", breed: "Poodle", color: "Apricot", coat: "Curly", size: "Small"})  // Create relationships between the second adoption center and the dogs  CREATE (ac2)-[:HAS]->(dog6)  CREATE (ac2)-[:HAS]->(dog7)  CREATE (ac2)-[:HAS]->(dog8)  CREATE (ac2)-[:HAS]->(dog9)  CREATE (ac2)-[:HAS]->(dog10)  //Plan for content-based recommendation algorithm MATCH (a:AdoptionCenter)-[:HAS\_DOG]->(d:Dog)  WITH a, d  MATCH (b:AdoptionCenter)-[:HAS\_DOG]->(e:Dog)  WHERE d.breed = e.breed AND d.color = e.color AND d.coat = e.coat AND d.size = e.size  WITH a, b, collect(e) as similarDogs  CALL algo.similarity.jaccard.stream(b, similarDogs, {similarityCutoff: 0.5})  YIELD item1, item2, count2, similarity  RETURN b, similarDogs, similarity  ORDER BY similarity DESC  LIMIT 10 |

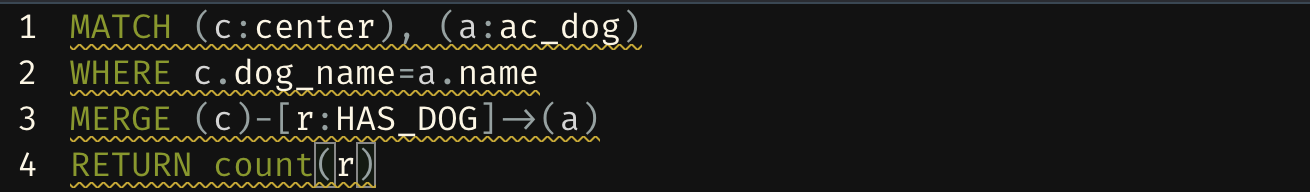
At first we planned to use Jaccard similarity for our recommendation system, but Jaccard similarity required weight values. We don’t have the standard for the value, so instead, we used kNN. If we do more after this project end, we will try to embed the kNN model for recommendation system. Below is some key cypher queries that we performed.





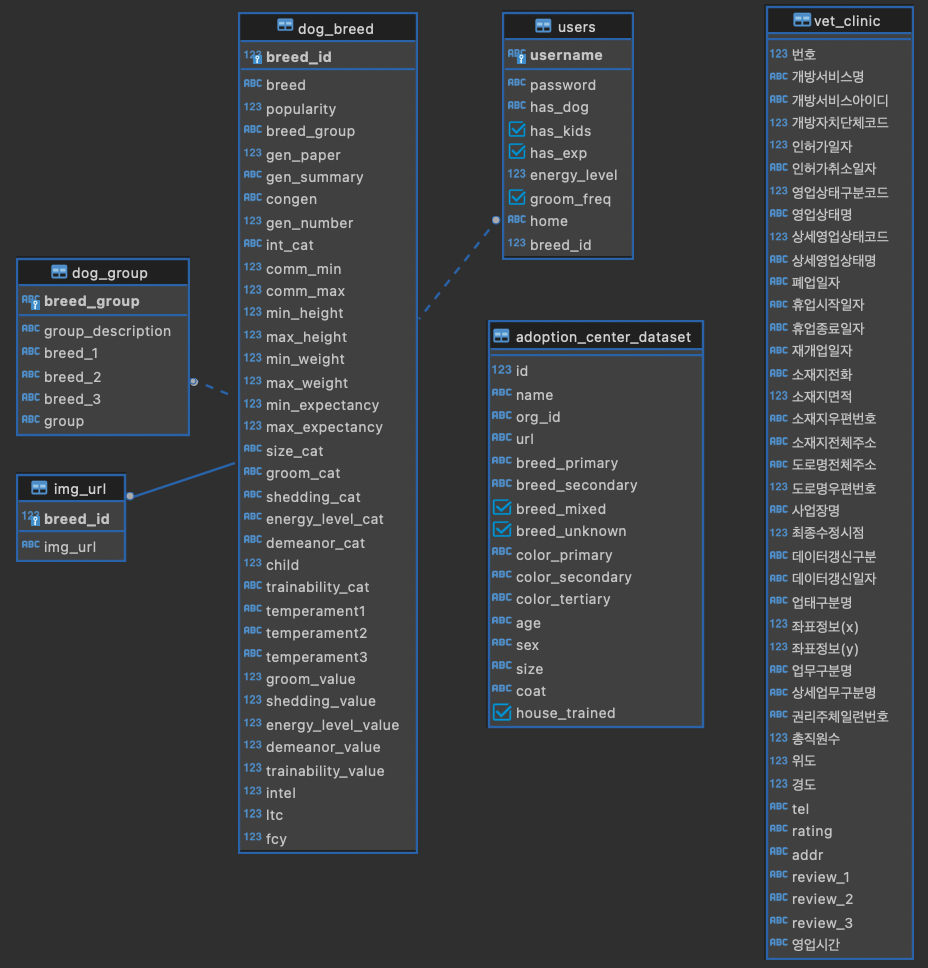




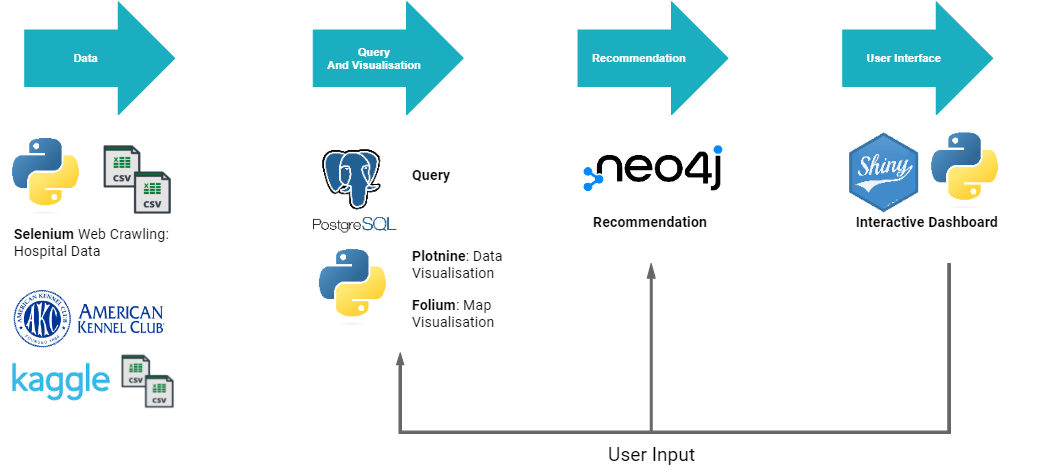


**3. ER Diagram**

This is our ER Diagram. Adoption center dataset and vet\_clinic data is instance data at the front-side



**4. Architectural design**



As mentioned in Section 2.1.1, Selenium, Beautiful Soap, CSS\_Collector was used for web crawling of hospital data. The other datasets were retrieved from already web-scraped data. The data were pre-processed with python or R. Python shiny was used to create the front-end interactive dashboard. Shiny is an interactive application that can take real time input from users by using reactive programming. Therefore, it can take in user input, query and return relevant dog breeds and return dog breed traits for visualization. Data visualization was done by Python’s Plotnine package. The map data to display hospital data was presented through Python’s Folium package. Next, the Neo4J was used for recommending adoptable dogs to the user. Neo4J was used because of its ability to create content based filtering.

**5. Results / demonstration**

**5.1 Dog Breed Data Filtering**

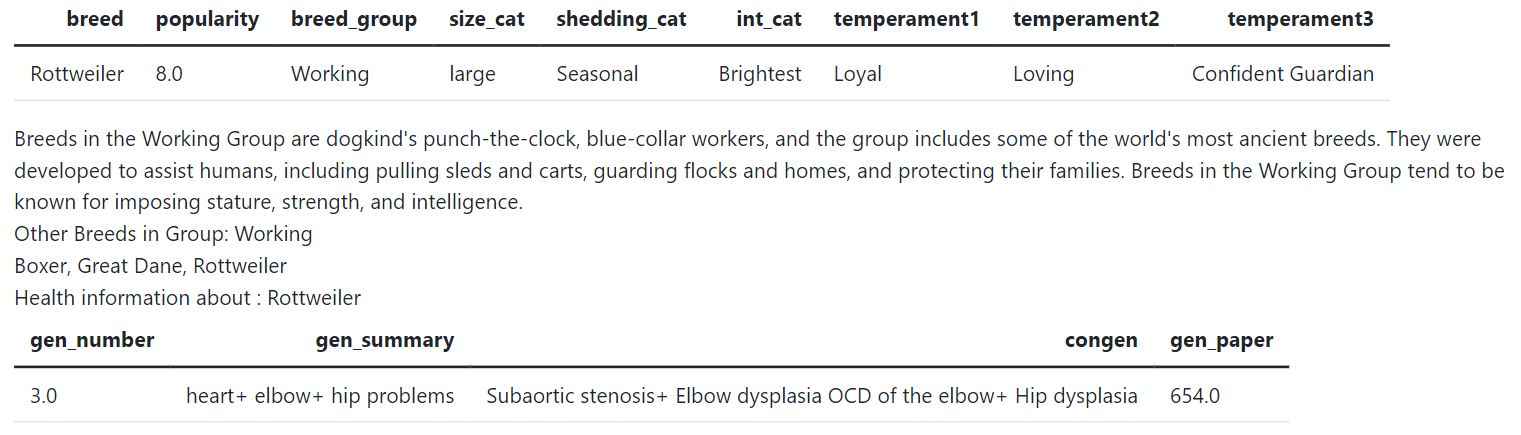
**Taking input from user**

|  |  |
| --- | --- |
|  | The user inputs questionnaire inputs to query dog breeds. |

Breeds relevant are returned as shown in the table below. We can see that the Couch Potato and Calm dog breeds are returned. For a user who doesn’t like to leave the house a lot, Basset Hounds, Cavalier Kind Charles Spaniel, French Bulldog, Pekingese, Tibetan Spaniels are suitable for this user.

**5.2 Dog Breed Information and Visualization**

**Example: Rottweiler**

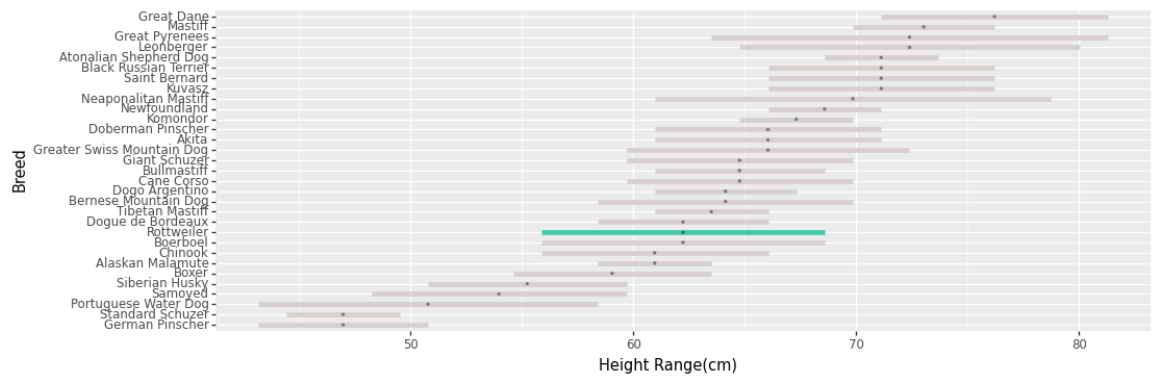


We can see that Rottweilers have three common illnesses: Subaortic stenosis, Elbow Dlysplasia OCD of the elbow and Hip dysplasia. In layman terms, they have heart, elbow and hip problems. These illnesses in Rottweilers have been studied in 654 papers. There is also information about the breed group and other dogs in this group. This adds a second layer of recommendation for users who are interested in dogs of the same group. For example, some people like Toy breed dogs like Chihuahuas, Pugs, and Shih Tzu.

****

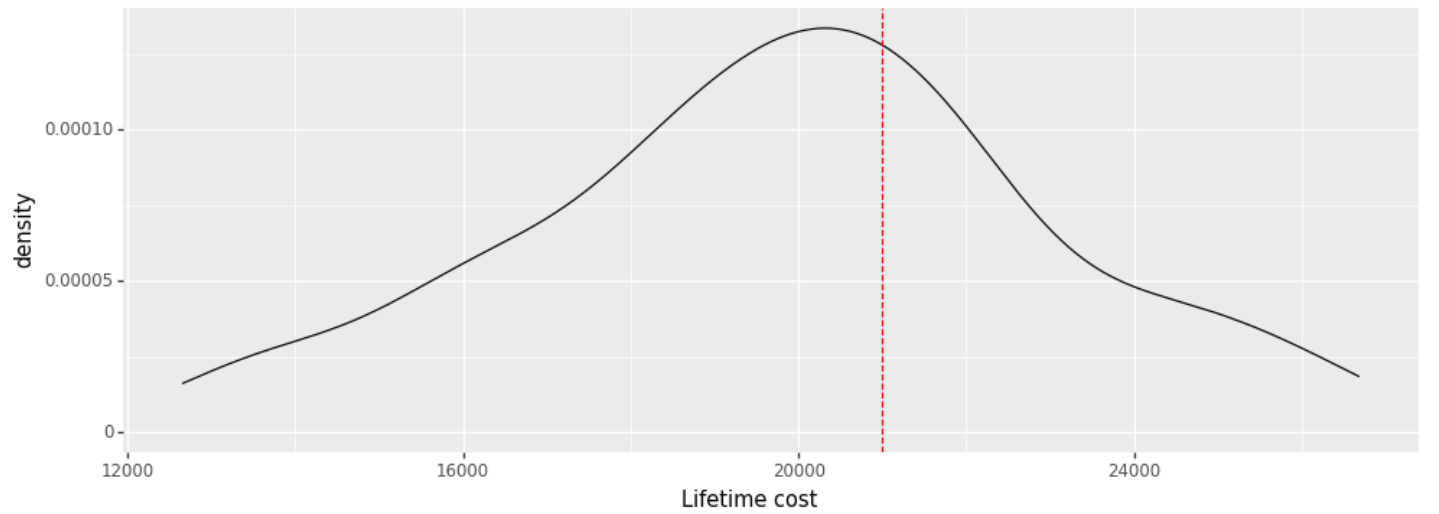
**Figure 1 Quick Visualization of 0 to 1 scale of grooming frequency, shedding frequency, energy level, demeanor value, and trainability.**

We can see that Rottweilers need to be groomed 2-3 times a week, shed frequently and need regular exercise. They are friendly to other strangers and pets and are somewhat trainable. An amazing breed!

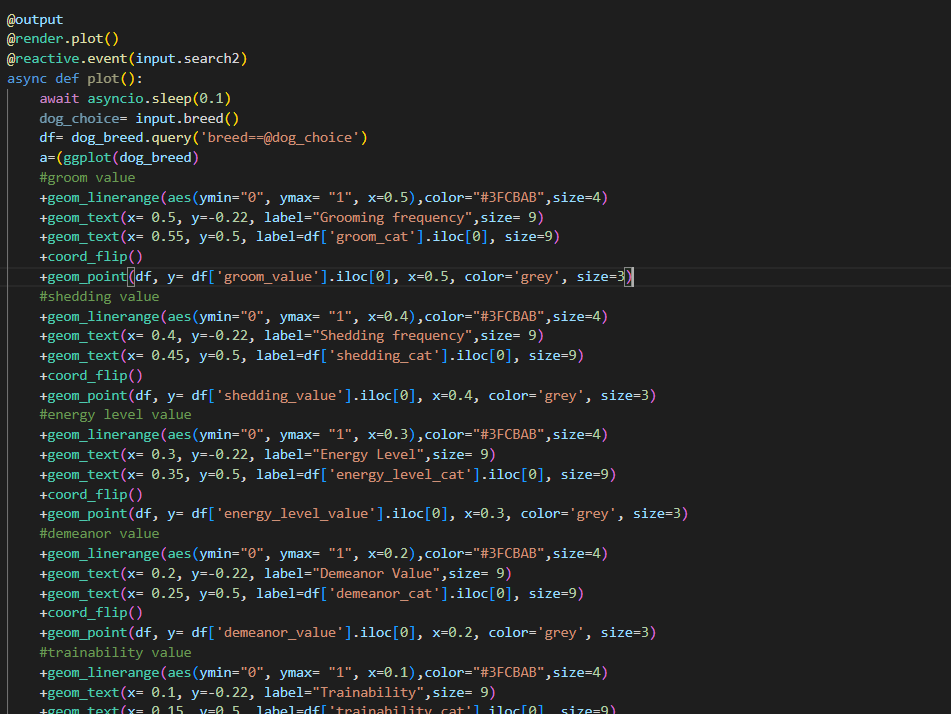


**Figure 2 Height range of Rottweilers in comparison to dogs in the same Working breed group.**

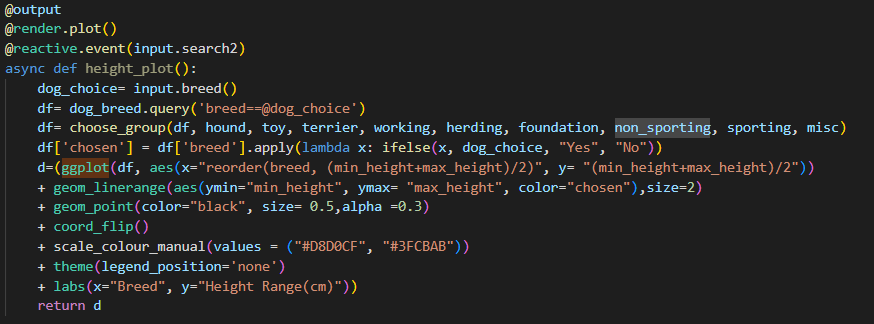
We can see that Rottweilers are not that big compared to breeds in the working breed group. They have a height range of approximately 57 to 68cm. The working breed group are known to be big because they have been used have been used for driving, herding and guarding livestock, pulling sleds, hunting small and large game and more.



This graph shows the life time cost (LTC) of the Rottweiler breed. The distribution is based on an arbitrary normal distribution based around the mean LTC of all breeds and is not reflective of a true life time cost distribution of different breeds. This is because there were many NA values, causing a multimodal distribution that was unpleasant. As we can see Rottweilers cost around 20500 USD throughout their lifetime. Users who find this economically burdensome should not adopt a Rottweiler.

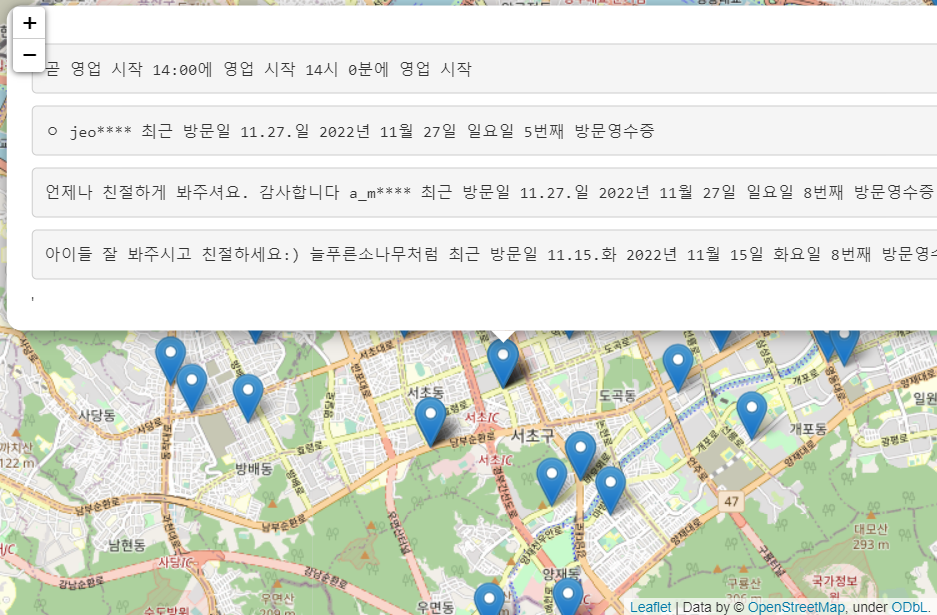


**Code for Figure 1.**

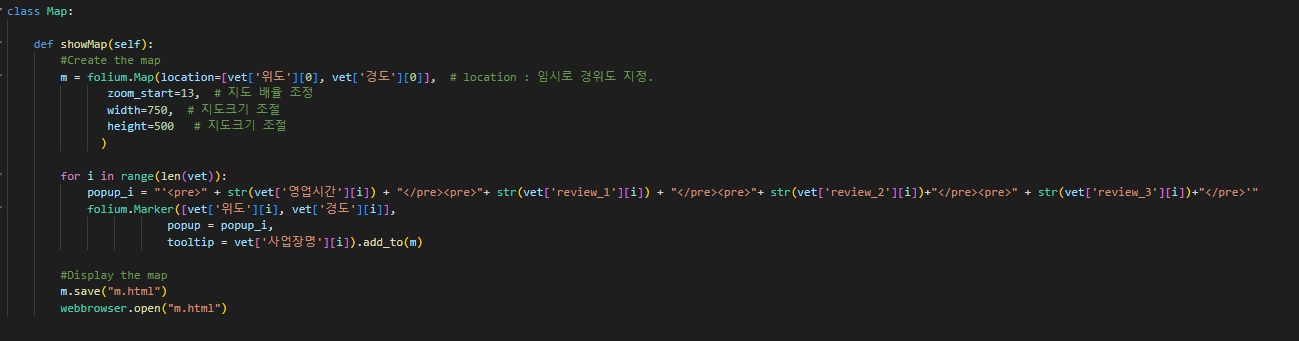


**Code for Figure 2**

**5.3 Closest Hospitals Information**



**Figure 3 Hospital Reviews**

**Code for Figure 3**

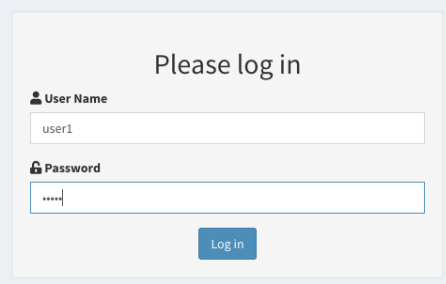
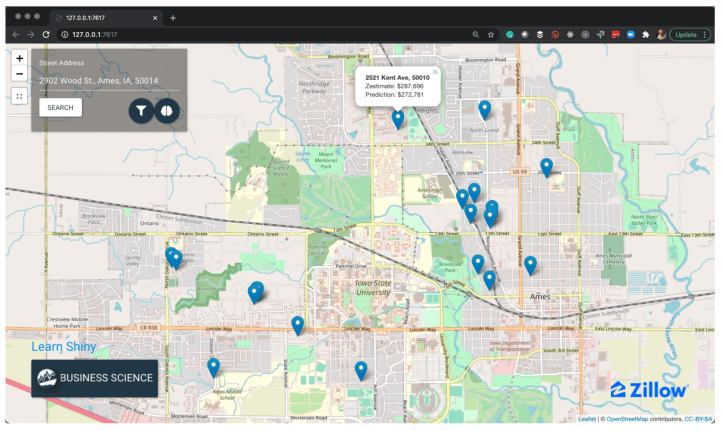
**Demo Video Links of 5.1~5.3**

[**https://youtu.be/ddgXhrWABoc**](https://youtu.be/ddgXhrWABoc)

[**https://www.youtube.com/watch?v=dgY-zvHb3cY**](https://www.youtube.com/watch?v=dgY-zvHb3cY)

**6. Limitation and Reflections**

Initially, the project was aimed to emulate a tinder-esque adoptable dog swiping style format, however, due to the lack of experience in architectural and app design, it was difficult to implement in our demo. Specifically, no member had experience in front-end development and but saw that shiny had a low learning curve, so we learn python shiny and implemented the front-end using Python shiny. However, Python shiny is still in the beta development stage and lacks important libraries. R shiny includes the library shinyauthr that allows it to add a login page and format website navigation or interactive map visualizations. If we had more experience in developing front-end applications, then we would be able to find the right tools for our desired function. Therefore, we learnt the importance of architectural design.



Also, for the questionnaire to dog breed filtering system, whether the filtering method can be improved is also something for consideration. For example, in the question "How energetic are you?”, the question is subjective and open for interpretation. Possibly, applying information retrieval methods may be of use where high relevance of queried results are important. However, even this may be excessive as there are only 277 dog breeds, not 100,000 relevant searches. Ultimately, deeper consideration of what questions to use will likely improve user experience, and help users to find the most compatible dog breed.

Additionally, in the visualization of life time cost of different dog breeds, the distribution was an arbitrary normal distribution due to prevalence of NA values in our dog\_breed.csv dataset. As mentioned in section 2.1.2A, although we did impute some variables, we did not impute all variables. This leads to some breeds not having data to present. To improve our service, manual imputation based on research may be required.

**7. Team member introduction, roles, and contribution**

**ooo -** Recommendation System

**ooo-** Naver Place Review Data Crawling, Folium map function.

**박정명-** Made python shiny website, SQL Query, and Data visualization

**8. Future work / possible extensions**

Our application currently has many tools for users who want to adopt dogs. We plan on adding functions to help improve care for dogs. For example, helping users take care of the dog's life cycle, real-time alert of vaccinations, feed information, etc. As we lacked members and time, we could not implement these functions, but these functions would help users take care of their dog. We also plan on changing from using python shiny to a different web development or interactive dashboard program like Django or JavaScript.

1. Fatjó J, Bowen J, García E, et al. Epidemiology of Dog and Cat Abandonment in Spain (2008-2013). *Animals (Basel)*. 2015;5(2):426-441. Published 2015 Jun 12. doi:10.3390/ani5020364 [↑](#footnote-ref-1)
2. Korean Animal Wellfare (2020) 유실유기동물분석 Page 42 [↑](#footnote-ref-2)
3. Holland, Katrina. (2019). Acquiring a Pet Dog: A Review of Factors Affecting the Decision-Making of Prospective Dog Owners. Animals. 9. 124. 10.3390/ani9040124. [↑](#footnote-ref-3)
4. 박윤정 윤서영 (2019) *반려동물 문제, 얼마나 알고 계신가요?*, *국민대학교 신문방송사*. https://press.kookmin.ac.kr/news/articleView.html?idxno=13260 (Accessed: December 19, 2022). [↑](#footnote-ref-4)