Databases Final Project Phase II

Kiho Kwon, kkwon14@jhu.edu, Section 315
Yifan Zhang, yzhan289@jhu.edu, Section 315

We created an application intended for job-searching users who are looking for a company that would be a good fit for those individuals. There are two search options, a basic search where the user fills out a form and a reverse search where the user gives a company as an example.

Project URL: https://ugrad.cs.jhu.edu/~yzhan289/Databases-Project/home.html

Populating the Database

The following sources were used:

Fortune 1000 data

https://www.kaggle.com/Eruditepanda/fortune-1000-2018

Crime data

https://ucr.fbi.gov/crime-in-the-u.s/2018/crime-in-the-u.s.-2018/tables/table-8/table-8.xls/view

Price parity values

https://www.bea.gov/news/2019/real-personal-income-states-and-metropolitan-areas-2017

State codes

https://github.com/jasonong/List-of-US-States/blob/master/states.csv

First, excel sheets were converted to csv files. Some basic preprocessing was done manually, such as taking out headers and footers of converted excel sheets. Then, Python was used to manipulate the data into a form conducive to inserting into SQL tables. For example, many of the numbers were comma-separated and included symbols such as dollar signs or percents, and these needed to be removed. The script and original csv files can be found in the populate_database/csv_creation directory. Python was also used to insert tables and views with the SQL table definitions and then insert the values from the processed csv files into the tables. This script, called create_tables.py, is inside the populate_database directory. Usage instructions can be found in the README.

Area of Specialization

Our minor area of specialization is in advanced GUI form interface and report generation. We provide the user with two different search options. A basic search, where the user can fill out a form of desired attributes of the company, such as location and number of employees. We also provide an advanced search, where the user selects a company where the user has presumably worked in the past, and then provides attributes that they liked about the company and should not change, and attributes that they did not like and how they should change.

User's Guide to Run Code

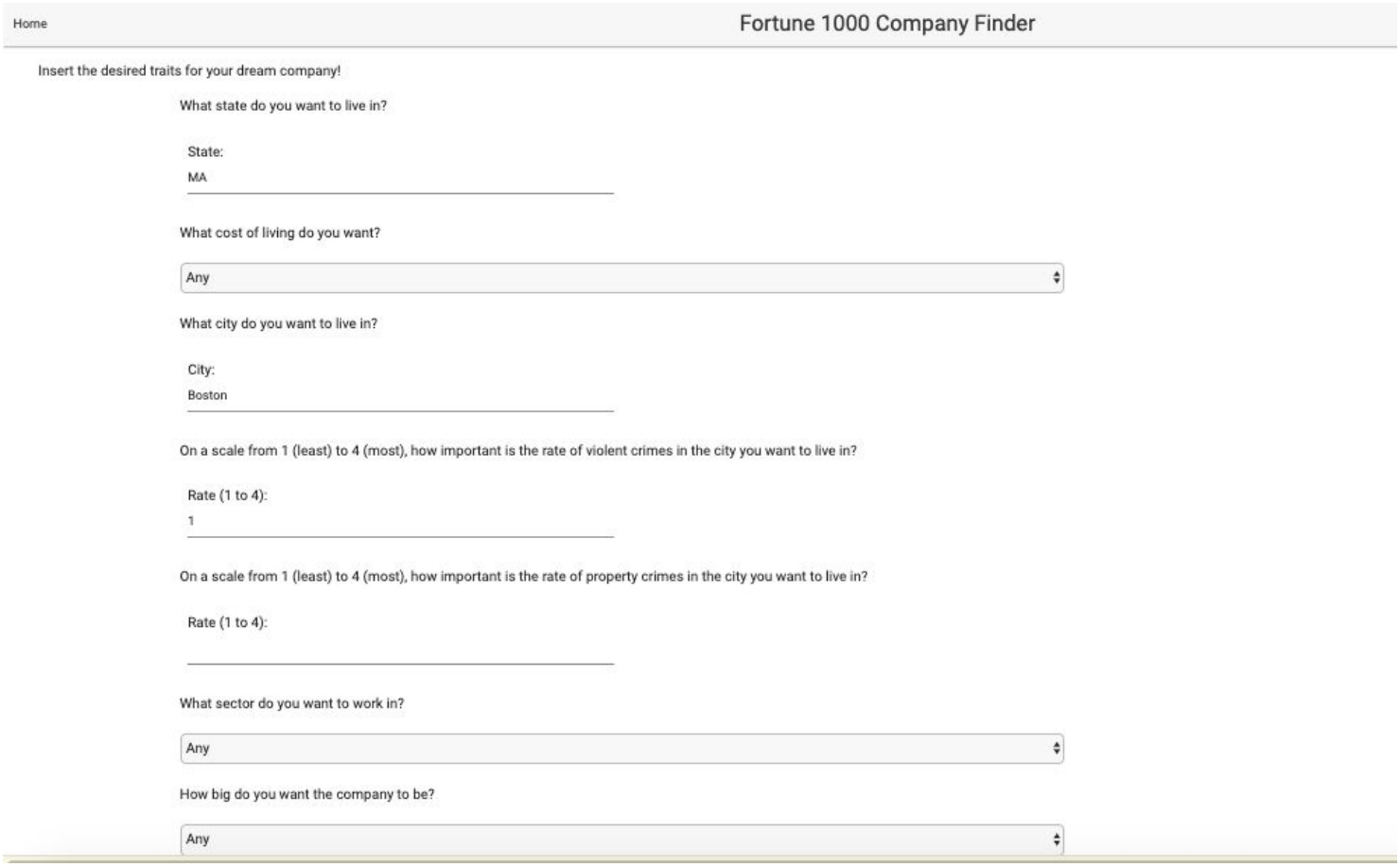
1. Go to https://ugrad.cs.jhu.edu/~yzhan289/Databases-Project/quiz.html to run the program
2. The basic search implementation can be found in quiz.html and quiz.php, and the reverse search implementation can be found in reverse.html and reverse.php.

3. In each of the .php files, it reads the inputs from the user, and creates an SQL query that is sent to the database to be computed.

Formatted Output/Screenshots

Example of basic search:

- Input:

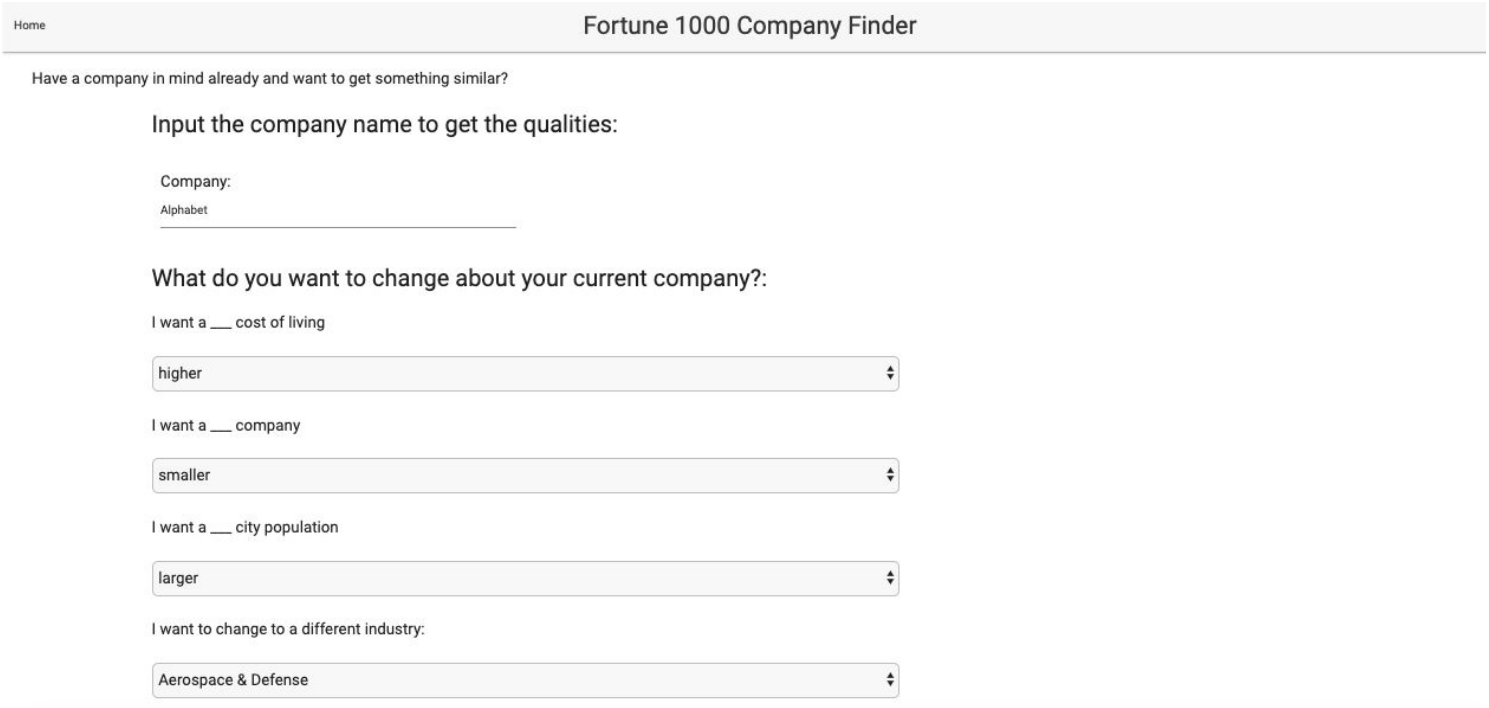


- Output:

Company Name	Fortune 1000 Rank	Profit (Million)	Number of Employees	Sector	HQ State	HQ City
General Electric	18	5786	313000	Industrials	MA	Boston
Liberty Mutual Insurance Group	68	17	50000	Financials	MA	Boston
State Street Corp.	259	2177	36643	Financials	MA	Boston
American Tower	419	1238.9	4752	Financials	MA	Boston
Wayfair	545	244.6	7751	Technology	MA	Boston
LPL Financial Holdings	582	238.9	3736	Financials	MA	Boston
Iron Mountain	619	183.8	24000	Business Services	MA	Boston
Cabot	791	241	4500	Chemicals	MA	Boston
Boston Properties	809	462.4	740	Financials	MA	Boston
Vertex Pharmaceuticals	831	263.5	2300	Health Care	MA	Boston

Example of reverse search:

- Input:



- Output:

Company Name	Fortune 1000 Rank	Profit (Million)	Number of Employees	Sector	HQ State	HQ City
Teledyne Technologies	807	227.2	10340	Aerospace & Defense	CA	Thousand Oaks
Aerojet Rocketdyne Holdings	989	9.2	5157	Aerospace & Defense	CA	El Segundo

Project Strengths

- We provide two types of searches depending on the user's needs.
- The basic search is the perfect tool for job applicants who know what qualities they want in a company, including the company size, industry, cost of living, and many more details.
- In addition to just the basic search of looking for companies, we also allow a reverse search option.
- If a user already has a company in mind, but wants something similar to that company, they can use the reverse search to list companies with similar traits to the listed company.
- The user is also able to modify whatever qualities they want, and if they leave specific options blank, the program will not use those traits to query. This is a very powerful tool, because this allows for the user to have full control on the qualities they deem important in a company, such as city population, safety, and cost of living.

Project Limitations/Possibilities for Improvement

- Unfortunately, much of the data that would be useful is not released by most companies (salaries, gender/diversity data, etc.) and therefore is not usually readily available. Given more time, we could potentially parse some websites that compile some of this information.
- We could implement a web crawler to find the job application page for each company and present that information to the user.
- We could also expand this project to include more than just the Fortune 1000 companies. We originally chose this list because it would be easy to find data on these companies.
- Given more time, we can allow users to add their own companies to the database, and allow smaller companies to gain exposure.
- We could also directly add the company website to the database. We were unable to do it this time because the dataset did not include the URL of the companies.

Miscellaneous Phase I Information

The phase I information that were not relevant in this report.

Questions:

1. Compute the companies with the highest growth that are located in cities with a population higher than 500,000 people
2. Find companies in the same sector as Walmart but had higher profit growth than Walmart
3. Compute the companies with HQ with the highest value per dollar grouped by state
4. List the companies headquartered in the city with the lowest murder rate
5. Find all companies that lost money in the last year
6. Find all finance companies that have an annual profit of more than 1,000 million dollars
7. Compute all companies located in they city that has the highest rape rate
8. Find all companies that increased in both ranking and experienced profit growth in the last year
9. Compute all technology companies that are not headquartered in California
10. Compute the average revenue by sector
11. List the proportion of the city population to number of employees for each company
12. Find the finance company with the smallest amount of assets
13. Compute the average revenue change of all tech companies ranked in the top 50
14. Compute the company with the highest profits per employee
15. Find the sector with the highest rent prices

Relational Model:
(note: was slightly changed in phase II due to datasets used and changes in opinion)

Company

name	Walmart
sector	Retailing
years_on_list	25
change_in_rank	0

industry	General Merchandisers
ceo	C. Douglas McMillon
profit_mil	6,670.0
profit_change	-32.4
rank	1
num_employees	2,200,000
hq_city	Bentonville
hq_state	Arkansas
website	http://www.walmart.com

CityInfo

c_name	Cupertino
s_name	CA
c_population	60,777
rape_rate_per_1000	0.15
murder_rate_per_1000	0.03

StateInfo

s_name	CA
s_population	39,000,000
relative_rent	1.506
dollar_value	1.214

Example SQL queries
1. Find all companies that lost money in the last year

```
SELECT
  name
FROM
  Company
```

```
WHERE
  profit_mil < 0;
```

2. Find companies in the same sector as Walmart but had higher profit growth than Walmart

```
SELECT
  Company.name,
  Company.profit_change
FROM
  Company,
  (SELECT
    *
  FROM
    Company
  WHERE
    Company.name = ‘Walmart’) as Walmart
WHERE
  Company.name != ‘Walmart’
  AND Company.profit_change > Walmart.profit_change;
```

3. Compute all companies located in the HQ city that has the highest rape rate

```
SELECT Company.name
FROM Company, (
      SELECT c_name, s_name
      FROM Company, CityInfo
      WHERE hq_city = c_name AND hq_state = s_name
      ORDER BY rape_rate_per_1000 DESC LIMIT 1
) AS highest_rape_city
WHERE hq_city = highest_rape_city.c_name AND hq_state =
highest_rape_city.s_name;
```

4. Compute the average revenue by sector

```
SELECT
  avg(revenue) as average_revenue
FROM
  Company
GROUP BY
  sector;
```

5. Compute all technology companies that are not headquartered in California

```
SELECT
  Company.name
FROM
  Company
WHERE
  Company.state != ‘California’;
```

TODO Delete below this

```
$query = "SELECT * FROM Company WHERE 1 = 1\n"; # Base query with some true condition

if ($state) {
      $state_condition = sprintf("AND Company.state = '%s'\n", $state);
      $query .= $state_condition;
}

# More conditions would go here

# Attach ending semicolon
$query .= ";\n";

$mysqli->multi_query(query);
```

This is a version that includes changes to the select (optional, you could probably just always select everything you need) and an example of how the

employees thing would work for the reverse search (where you can choose greater than or less than).