CSCI3170: Introduction to Database Systems **Spring 2018**

Project — NEAs Exploration Mission Design System

Prepared by Zitong Chen, Jianwen Zhao, Ada Waichee Fu and Jingsong Chen

1 Introduction

Asteroids, also known as minor planets, have been attracting the interest of scientists for decades, as these small solar system bodies may hold key information about the origin and formation of our solar system and pristine materials or valuable resources [1] for our industries. Sample return missions to these small bodies could yield unique insight and substantially advance our understanding of the solar system[2]. Up to now, millions of asteroids have been identified in the space, most of them are orbiting the sun between Mars and Jupiter within the main asteroid belt, as indicated in Figure 1[3]. Among the asteroids population, a subgroup is referred to as **Near-Earth Asteroids** (**NEAs**), having a maximum perihelion distance of 1.3 AU (Astronomical Unit, 1 AU=1.496×10^9 km) from the sun by definition, they are relatively close to Earth, approaching or intersecting Earth's orbit around the sun (see Figure 2)[4][5]. In Figure 2, the white ball in the center is the sun, the green line shows the orbit of the earth, and the dots are the NEAs around the earth. This subgroup of NEAs is of particular significance for space exploration. On one hand, missions to these asteroids require relatively low energy and offer a wide range of possibilities for space exploration, scientific research, and technology demonstration. On the other hand, some of NEAs have the potential to hit the Earth and cause significant damage, thus exploration to these *Potentially Hazardous Asteroids* may help human survive from an impact event.

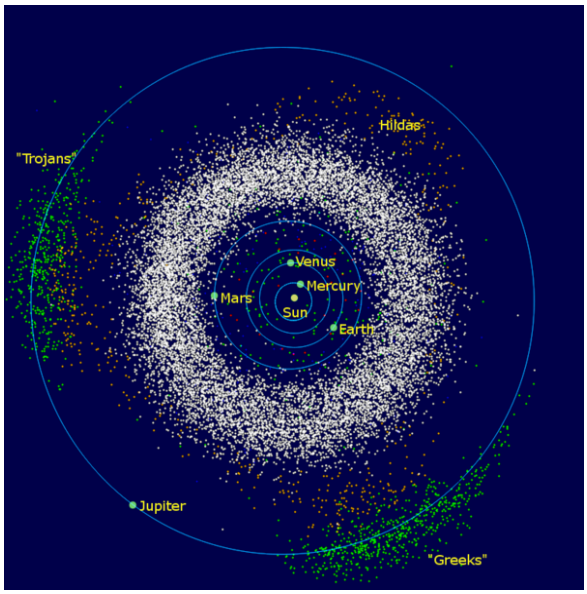

Figure 1: The asteroids spatial distribution

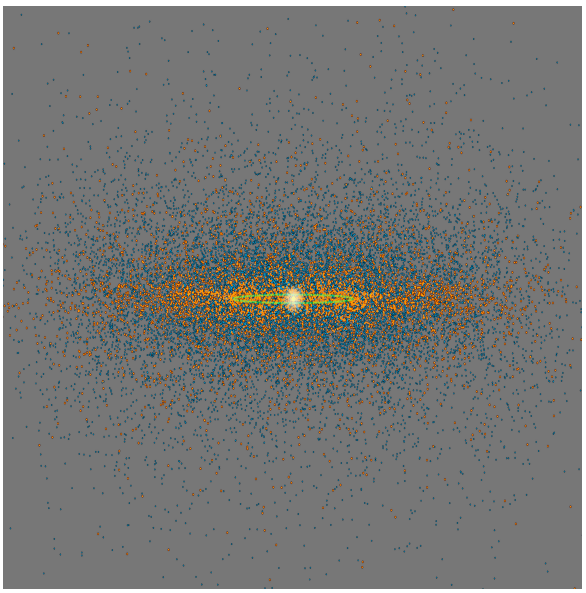

Figure 2: A snapshot of NEAs

Due to the significant value of asteroids (especially NEAs), many space agencies around the world have conducted real missions or proposed their asteroids exploration plans. For example, JAXA (Japan Aerospace Exploration Agency) developed a *Hayabusa Mission* to explore one of the NEAs named *Itokawa* (ID: 25143), they launched the *Hayabusa* spacecraft in 9 May 2003, the spacecraft rendezvoused with Itokawa in mid-September 2005, and finally returned samples to Earth successfully on 13 June 2010; NASA (National Aeronautics and Space Administration) announced a mission on 4 January, 2017 for visiting the Trojan asteroids in 2021; US President Donald Trump announced on 5 October, 2017 that the US will return to the moon, which is also regarded as a stepping stone for future asteroids exploration.

Motivated by the above background, this project asks you to implement a **NEAs Exploration Mission Design System** in a futuristic world as a **Java Command-line Program**, in which data about *Near-Earth Asteroids*, *Space Agencies' Spacecrafts*, *Spacecraft Rental Records* and *Resources Details* are stored. The rest of this document explains the details.

Objective

The objective of the Mission Design System is to assist space exploration companies in their space exploration planning. It is assumed that such companies aim to mine resources from the NEAs, which will bring them benefits that outweigh the costs of the exploration. The system maintains data about NEAs and their properties, the exploration requirements for different NEAs, spacecraft properties, including current rental status of spacecrafts, rental costs of spacecrafts, and benefits of resources that can be returned from explorations. Given such data, the system aims to answer queries from exploration companies on mission design. Given criteria such as budget and targeted resource, the system can return records about spacecrafts that can meet the requirements of exploration companies.

In the following, Section 2 highlights the 2 phases of this project and the expectations from each phase. We specify the data that will be provided in Section 3. Section 4 lists the assumptions made in this project. Section 5 is about the function requirements. The remaining sections explain the submission process and grading. Students are encouraged to read carefully all details in this specification and raise any question or issue to our tutors as early as possible.

2 Milestones

This project is divided into 2 phases.

Phase 1:

You are required to design the database for the system (including an ER-diagram and a relational schema). A suggested solution will be provided after phase 1. You are required to use the suggested solution to complete phase 2.

Phase 2:

You are required to implement the **NEAs Exploration Mission Design System** as a **Java Command-line Program**. The tutors will give tutorials on how to connect to a MySQL database system with JDBC API and deploy your work on the required platform.

3 Data Specification

All data files of the system are in **linux text file format** (i.e. newline character is **\n**) encoded in ASCII. Your Java command line application is required to read records stored in the files and insert them into appropriate tables of the provided MySQL DBMS via JDBC API. There are four input files, a list of *Near-Earth Asteroids*, a list of *Space Agencies' Spacecrafts*, a list of *Spacecraft Rental Records* and a list of *Resources Details*. Each line of each input file is a sequence of attributes delimited by tab (**\t**) character. The definition of each attribute in each input file is defined in the corresponding subsections.

3.1 Near-Earth Asteroids

Records of *Near-Earth Asteroids* that will be provided to you resemble the following.

NID	Distance(AU)	Family	Duration(days)	Energy(km/s)	Resource
1991JW	1.04	Apollo	346	6.394	C
1999AO10	0.91	Aten	115	6.440	Pt
1997YM9	1.10	Apollo	338	6.725	null
...	...	...	...	...	...

The format and description of each attribute are given in Table 1.

Table 1: **Attribute format and description of Input File 1**

Attribute	Format	Description
`NID`	a string with at most 10 characters	the unique identifier of the asteroid
`Distance(AU)`	a positive floating-point number with double precision	the length of the semi-major axis of the asteroid's orbit, measured in Astronomical Units (AU)
`Family`	a string with at most 6 characters	the category the asteroid belongs to
`Duration`	a positive integer with 3 digits	the **minimum** period required for exploring the asteroid
`Energy(km/s)`	a positive floating-point number with double precision	the **minimum** energy required for exploring the asteroid
`Resources`	a string with at most 2 characters	the main resource type the asteroid contains

Note: The `null` value of `Resource` implies that the NEA does not have any of the target resources.

3.2 Resources Details

The *Resources Details* that will be provided to you are the following.

Type	Density(g/cm^3)	Value($/g)
Au	19.26	42.41
Ag	10.49	0.61
C	3.52	22087.80
Pt	21.45	33.96
Pd	12.02	23.97

The format and description of each attribute are given in Table 2.

Table 2: **Attribute format and decription of Input File 2**

Attribute	Format	Description
`Type`	a string with at most 2 characters	the type* of resource, `Au`: Gold, `Ag`: Silver, `C`: Diamond, `Pt`: Platinum, `Pd`: Palladium
`Density(g/cm`^{3}`)`	a positive floating-point number with double precision	the density of the resource type
`Value($/g)`	a positive floating-point number with double precision	the market price of the resource type

Note*: Only 5 types of resources (i.e., `Au, Ag, C, Pt, Pd`) are considered in this project.

3.3 Space Agencies' Spacecrafts

The records of *Space Agencies' Spacecrafts* that will be provided to you resemble the following.

Agency	MID	Num	Type	Energy(km/s)	T(days)	Capacity(m^3)	Charge($/day)
NASA	0001	3	A	10	300	10	1000
NASA	0002	2	A	12	500	20	2000
JAXA	0002	1	E	8	220	null	800
ESA	0001	4	A	12	350	8	1200
...	...	...	...	...	...	...	...

The format and description of each attribute are given in Table 3.

Table 3: **Attribute format and description of Input File 3**

Attribute	Format	Description
Agency	a string with at most 4 characters	the name of a space agency
MID	a string with at most 4 characters	the spacecraft model identifier within an agency
Num	a postive integer with at most 2 digits	the number of spacecraft(s) of this model from this agency
Type	a character	the mission type of the spacecraft, there are two types — A: it can conduct asteroid exploration; and E: it **cannot** conduct asteroid exploration
Energy(km/s)	a positive floating-point number with double precision	the **maximum** one-trip* energy the spacecraft can provide
T(days)	a positive integer with 3 digits	the **maximum** one-trip working time of the spacecraft
Capacity(m^3)	a positive integer with at most 2 digits	the **maximum** sample's volume that the spacecraft can return from NEAs in one trip
Charge($/day)	a positive integer with at most 5 digits	the cost per day for renting the spacecraft

Note 1*: A spacecraft is launched from Earth, after visiting an asteroid, it will directly return to Earth. This whole process is defined as "one trip".

Note 2: The null value of Capacity implies that the spacecraft does not have a capacity (it is a Type E spacecraft).

Note 3: For each spacecraft model of each agency there can be zero or more actual spacecrafts based on that model. The i-th spacecraft of a particular model is given a SNum $= i$. Therefore a unique spacecraft is identified by its Agency, its MID, and its SNum. See Item (2) in the Assumptions in Section 4.2 on Page 6.

3.4 Spacecraft Rental Records

The *Spacecraft Rental Records* that will be provided to you may look like the following.

```
Agency  MID   SNum  Checkout Date  Return Date
NASA    0002  1     01-01-2016     22-05-2017
ESA     0001  3     15-02-2017     null
ESA     0001  2     01-01-2018     null
...     ...   ...
```

The format and description of each attribute are given in Table 4.

Table 4: **Attribute format and description of Input File 4**

Attribute	Format	Description
`Agency`	a string with at most 4 characters	the name of the space agency
`MID`	a string with at most 4 characters	the spacecraft model identifier within the agency
`SNum`	a positive integer with at most 2 digits	The i-th spacecraft(copy) of the spacecraft model of an agency is given a SNum $= i$.
`Checkout Date`	`DD-MM-YYYY`	the date the spacecraft was last rented from the space agency
`Return Date`	`DD-MM-YYYY`	the date the spacecraft was last returned to the space agency

Note 1: All dates (`Y=year, M=month, D=day`) should have the same time zone as of Hong Kong.
Note 2: When a spacecraft has been rented out but has not been returned to the space agency, the value of its `Return Date` is set to `null`.
Note 3: We only keep the dates for the most recent rental for any spacecraft.
Note 4: For a spacecraft that has never been rented out both the Checkout Date and the Return Date are set to the date when the spacecraft was built.

4 Assumptions and Regulations

4.1 System

- All numerical values will not be larger than the maximum integer value that can be handled by Java.
- The system is case sensitive.
- There is no duplicate row in any input and output.
- Your Java program may assume that any value entered into any input field is **correct in format only**.
- Your Java program may assume that all input data files are **correct in format and content**.

4.2 Data

(1) **Near-Earth Asteroids**

- Each NEA is uniquely identified by its `NID`.
- Each NEA contains at most one type of resource, and many NEAs have no resource.

(2) **Space Agencies' Spacecrafts**

- Assumptions about spacecrafts are highlighted in the Notes in Section 3.3.

(3) **Spacecraft Rental Records**

- Each rental record can be uniquely identified by the values of: {`Agency, MID, SNum`}.
- Assumptions about the rental records are stated in the Notes in Section 3.4.

(4) **Resources Details**

- Each kind of resource can be uniquely identified by its `type`.

4.3 Mission

A complete NEAs exploration mission in charged by an exploration company includes the following steps:

(a) Launch a spacecraft from Earth.

(b) After a period of flight, the spacecraft will rendezvous with the target asteroid, and make a soft landing on the asteroid's surface.

(c) Each spacecraft will perform sample mining automatically, after being fully loaded, it takes off from the asteroid's surface and return.

(d) After another period of flight, the spacecraft will return samples to Earth.

This whole process is regarded as "**one trip**" and we have some **important assumptions** here, considering a given NEA.

1. The "one trip" mission period (denoted as T_M) is exactly the same as the minimum required period (i.e., the `Duration` attribute) of the NEA that is being explored.
2. An exploration company will always rent a suitable spacecraft for a "one trip" mission period, namely, the rental period T_R of a spacecraft is equal to T_M ($T_R = T_M$).
3. A minimum energy E_M is required for the exploration of the NEA (see Section 3.1).
4. A spacecraft is suitable if it is not rented out, its maximum working time is at least T_M, and the maximum energy provided by the spacecraft is at least E_M. (see Section 3.3).
5. A spacecraft on a mission always returns to Earth with the maximum weight of one resource (fully loaded).
6. The `Cost` of a mission is calculated as follows:

$$C_M = C_R \times T_M \tag{1}$$

   where C_M is the mission cost, C_R is the rental charge per day of the spacecraft.

7. The `Benefit` of a mission is calculated as follows:

$$B_M = V \times \rho \times S_C - C_M \tag{2}$$

   where B_M is mission benefit, V is the value of the resource, ρ is the density of the resource, S_C is the spacecraft capacity.

8. The exploration company has a certain `budget` and $C_M \leq$ `budget`.

5 System Function Requirements

You are required to write a simple command line application in Java. After performing an operation, the program should display the last appeared menu. There are 3 types of users: **Database Administrator**, **Exploration Companies** and **Spacecraft Rental Staff**. Therefore, your Java program should provide the following functions for different users.

5.1 Database Administrator

The system should allow a database administrator to perform the following operations:

- **5.1.1 (Q1) Create tables in the database**

  This function creates all the tables for this system based on the relational schema given.

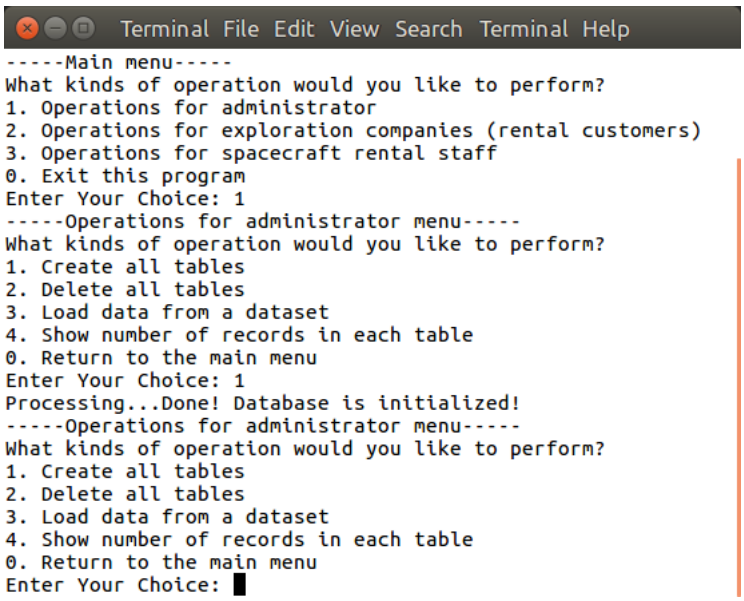

Figure 3: Expected input and output while creating table schemas in MySQL DBMS

- **5.1.2 (Q2) Delete tables in the database**

  This function deletes all existing tables in the system.

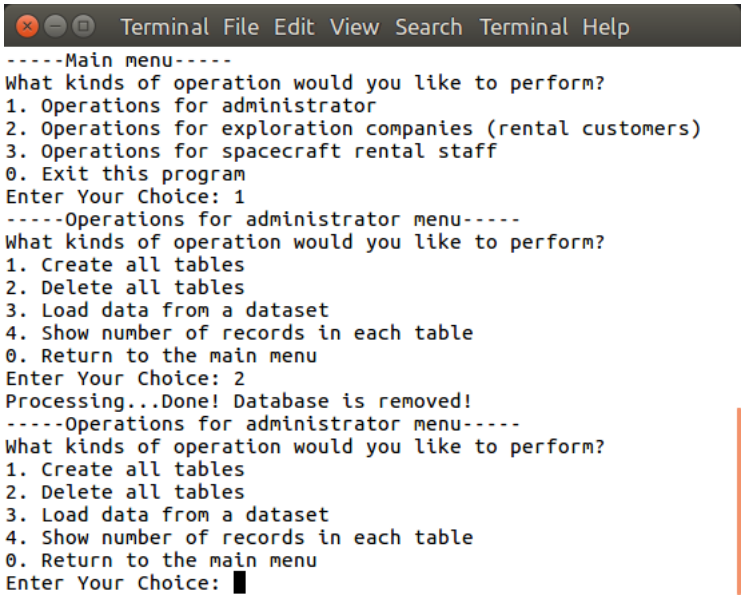

Figure 4: Expected input and output while deleting table schemas from MySQL DBMS

- **5.1.3 (Q3) Load data from a dataset**

  Prompt for user input, the system should then read all data files from the input folder path and load them into the appropriate tables in the database. Your program can assume that the user input folder contains all 4 data files, namely neas.txt, spacecrafts.txt, rentalrecords.txt and resources.txt. See **Section 3** for data specifications.

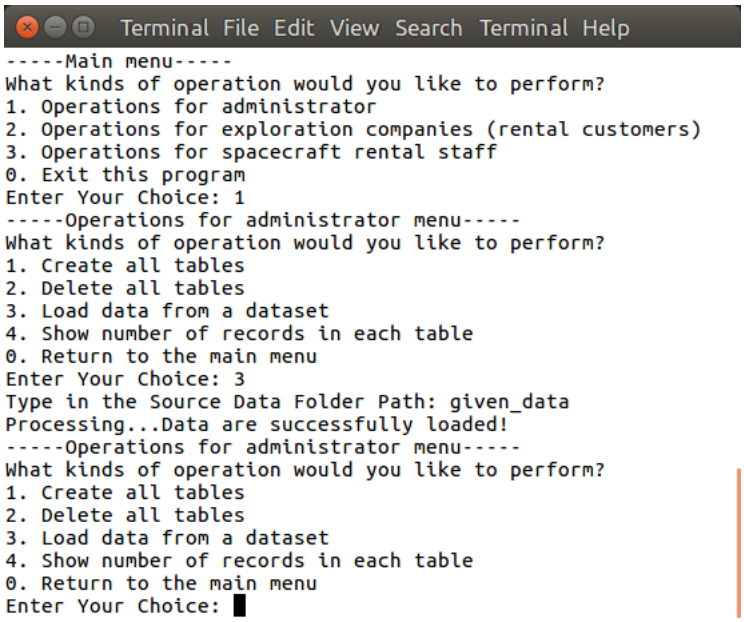

Figure 5: Expected input and output while loading table schemas from the database

- **5.1.4 (Q4) Show the number of records in each table**

  For each existing table, display the number of records in it.

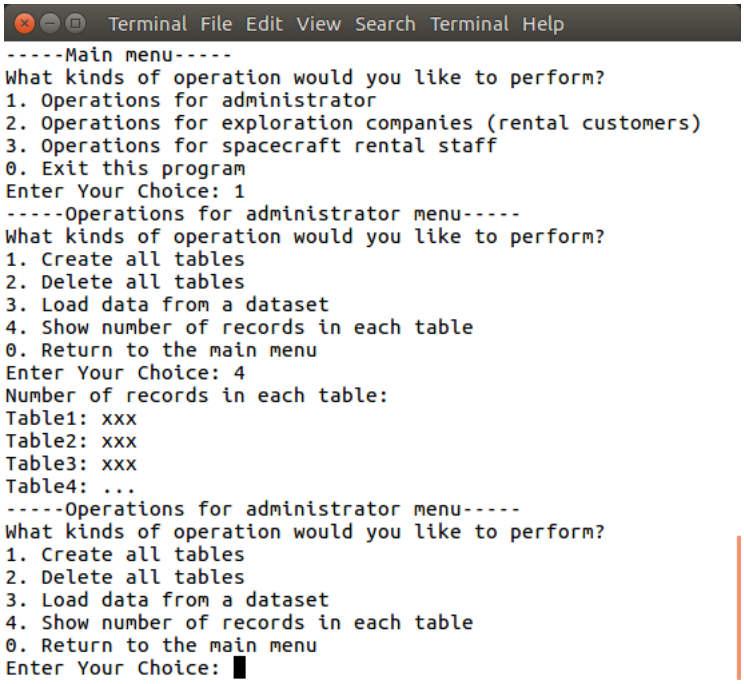

Figure 6: Expected input and output while showing number of records in each table

Remark: When tables are created, incorporate checking of constraints on the attribute values.

5.2 Space Exploration Companies

The system should allow exploration companies (rental customers) to perform the following operations:

- **5.2.1 (Q5) Search for NEAs based on some criteria**

  The system has to provide an interface to allow an exploration company (rental customer) to search for the NEAs in three different ways:

  - By `NID` (exact matching)
  - By `Family` (partial matching)
  - By `Resource type` (partial matching)

  You can assume that only one searching method can be selected by the exploration company for each query and the whole string entered by the user is considered as one search word (e.g. When a user entered "Database Concept", the system will take "Database Concept" as one and only one search keyword instead of two search keywords "Database" and "Concept"). After the exploration company inputs the search keyword, the system should perform the query and return all matching records in terms of their `NID, Distance, Family, Duration, Energy, Resources`. The results of the query should be output as a table as follows:

```
Terminal File Edit View Search Terminal Help
-----Main menu-----
What kinds of operation would you like to perform?
1. Operations for administrator
2. Operations for exploration companies (rental customers)
3. Operations for spacecraft rental staff
0. Exit this program
Enter Your Choice: 2
-----Operations for exploration companies (rental customers)-----
1. Search for NEAs based on some criteria
2. Search for spacecrafts based on some criteria
3. A certain NEA exploration mission design
4. The most beneficial NEA exploration mission design
0. Return to the main menu
Enter Your Choice: 1
Choose the search criterion:
1. ID
2. Family
3. Resource type
My criterion:1
Type in the search keyword: 1991JW
|ID|Distance|Family|Duration|Energy|Resources|
|1991JW|1.04|Apollo|346|6.394|C|
End of Query
-----Operations for exploration companies (rental customers)-----
1. Search for NEAs based on some criteria
2. Search for spacecrafts based on some criteria
3. A certain NEA exploration mission design
4. The most beneficial NEA exploration mission design
0. Return to the main menu
Enter Your Choice:
```

Figure 7: Expected input and output while searching for NEAs

Remark: When tables are created, incorporate checking of constraints on the attribute values.

- **5.2.2 (Q6) Search for spacecrafts based on some criteria**

  The system has to provide an interface to allow an exploration company (rental customer) to search for the spacecrafts in five different ways (all are exact matching):

  - By `Agency Name` (return spacecrafts belonging to a certain agency)
  - By `Type` (return spacecrafts belonging to a certain type)
  - By `Least energy` (return spacecrafts which can provide more than the least one-trip energy)
  - By `Least working time` (return spacecrafts which can work longer than the least working time entered)
  - By `Least capacity` (return spacecrafts which have more than the least capacity entered)

  Similar with searching for NEAs, you can assume that only one searching method can be selected by the exploration company for each query and the whole string entered by the user is considered as one search word. All matching records will be returned in terms of their `Agency, MID, Num, Type, Energy, Capacity, Charge` and output as a table as follows:

```
Terminal File Edit View Search Terminal Help
-----Main menu-----
What kinds of operation would you like to perform?
1. Operations for administrator
2. Operations for exploration companies (rental customers)
3. Operations for spacecraft rental staff
0. Exit this program
Enter Your Choice: 2
-----Operations for exploration companies (rental customers)-----
1. Search for NEAs based on some criteria
2. Search for spacecrafts based on some criteria
3. A certain NEA exploration mission design
4. The most beneficial NEA exploration mission design
0. Return to the main menu
Enter Your Choice: 2
Choose the search criterion:
1. Agency Name
2. Type
3. Least energy [km/s]
4. Least working time [days]
5. Least capacity [m^3]
My criterion: 5
Type in the search keyword: 10
|Agency|MID|SNum|Type|Energy|T|Capacity|Charge|
|NASA|0001|1|A|10|300|10|1000|
|NASA|0001|2|A|10|300|10|1000|
...
End of Query
-----Operations for exploration companies (rental customers)-----
1. Search for NEAs based on some criteria
2. Search for spacecrafts based on some criteria
3. A certain NEA exploration mission design
4. The most beneficial NEA exploration mission design
0. Return to the main menu
Enter Your Choice:
```

Figure 8: Expected input and output while searching for spacecrafts

- **5.2.3 (Q7) An NEA exploration mission design**

  The system has to allow the exploration company to obtain all possible exploration solutions (all spacecrafts that can be used for conducting the mission currently and the corresponding cost and benefit) for a specified NEA. You may assume that the mission starting time is exactly the same with the current **system time**. When the exploration company enters a NEA ID, the system should search for possible spacecrafts by checking the following conditions:

  - `type` of a spacecraft (requires `type` A)
  - `Energy` of a spacecraft
  - T of a spacecraft
  - **availability** of a spacecraft

  For example, the exploration company wants to visit the NEA `1999AO10` and bring back some resource sample. The minimum period and energy required for achieving the purpose are $T_r =$ 115 days, $E_r = 6.440$ km/s (see **Section 3.1**), respectively.

  The spacecraft belonging to `NASA` and whose `MID` is 0001 has the following properties:

  – it is `type A` (meaning that it can be used for asteroid exploration)
  – the energy it can provide is $E_p = 10$ km/s,
  – its maximum working time is $T_p = 300$ days (see **Section 3.3**).

  We see that

  – $E_p > E_r$, $T_p > T_r$ and
  – the rental records show that `NASA 0001` currently is available (see **Section 3.4**).

  As a result, the spacecraft `NASA 0001` fulfills all the four conditions aforementioned and thus is regarded as a possible solution.

  The results should be output as a table with attributes `Agency, MID, SNum, Cost, Benefit`, and ordered by `Benifit` in descending order. For the definition of `Cost` and `Benefit`, please refer to **Section 4.3**.

```
Terminal File Edit View Search Terminal Help
-----Main menu-----
What kinds of operation would you like to perform?
1. Operations for administrator
2. Operations for exploration companies (rental customers)
3. Operations for spacecraft rental staff
0. Exit this program
Enter Your Choice: 2
-----Operations for exploration companies (rental customers)-----
1. Search for NEAs based on some criteria
2. Search for spacecrafts based on some criteria
3. A certain NEA exploration mission design
4. The most beneficial NEA exploration mission design
0. Return to the main menu
Enter Your Choice: 3
Typing in the NEA ID: 1999AO10
All possible solutions:
|Agency|MID|SNum|Cost|Benefit|
|NASA|0001|1|115000|224600|
| ESA|0001|2|138000|133680|
...
End of Query
-----Operations for exploration companies (rental customers)-----
1. Search for NEAs based on some criteria
2. Search for spacecrafts based on some criteria
3. A certain NEA exploration mission design
4. The most beneficial NEA exploration mission design
0. Return to the main menu
Enter Your Choice:
```

Figure 9: Expected input and output while designing mission for a certain NEA

- **5.2.4 (Q8) The most beneficial NEA exploration mission design**

  The system has to be able to generate a most beneficial solution, with two requirements specified by the exploration company in advance:

  - `budget`: the upper bound of the cost
  - `resource type` of NEAs: a certain type of resource.

  After these two are entered, the system should look for all possible solutions (the checking procedure is similar with that of a certain NEA exploration mission design) which satisfy all the constraints and output the best solution in terms of `NEA ID, Family, Agency, MID, SNum, Duration, Cost, Benefit`. If there are two or more solutions with the greatest benefit, output any one of the solutions. Limit your result to one answer. An example is shown as follows:

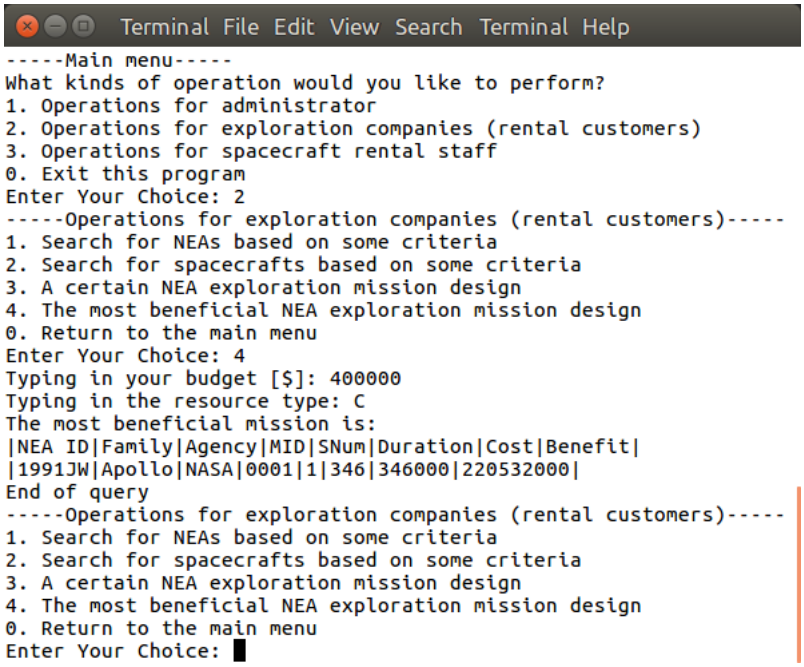

Figure 10: Expected input and output while designing the most beneficial mission

5.3 Spacecraft Rental Staff

The system should allow spacecraft rental staff to perform the following operations:

- **5.3.1 (Q9) Rent a spacecraft**

  A rental staff can perform the spacecraft rental procedure through the system. First, he/she needs to input the `Agency`, the `MID` and the `SNum`. Then the system should check whether that spacecraft is available to be rented (i.e. There is no rental record of the specified spacecraft with `null` return date). If the spacecraft is available, it is then rented and a new rental record of the specified spacecraft with `null` return date should be added to the database accordingly. Finally, there should be an informative message stating whether the spacecraft can be rented successfully or not.

```
Terminal  File  Edit  View  Search  Terminal  Help
-----Main menu-----
What kinds of operation would you like to perform?
1. Operations for administrator
2. Operations for exploration companies (rental customers)
3. Operations for spacecraft rental staff
0. Exit this program
Enter Your Choice: 3

-----Operations for spacecraft rental staff-----
1. Rent a spacecraft
2. Return a spacecraft
3. List all spacecraft currently rented out (on a mission) for a certain period
4. List the number of spacecrafts currently rented out by each Agency
0. Return to the main menu
Enter Your Choice: 1
Enter the space agency name: JAXA
Enter the MID: 0003
Enter the SNum: 1
Spacecraft rented successfully!

-----Operations for spacecraft rental staff-----
1. Rent a spacecraft
2. Return a spacecraft
3. List all spacecraft currently rented out (on a mission) for a certain period
4. List the number of spacecrafts currently rented out by each Agency
0. Return to the main menu
Enter Your Choice:
```

Figure 11: Expected input and output while a rental staff processes a spacecraft rental request

- **5.3.2 (Q10) Return a spacecraft**

  A rental staff can perform the spacecraft return procedure through the system. First, he/she needs to input the `Agency`, the `MID` and `SNum` of the spacecraft being rented. Then the system should check if a rental record corresponding to the specified spacecraft exists. If such record is found, the spacecraft can be returned and the return date of the rental record found is updated to be the **current date of the system**. Finally, there should be an informative message stating whether the spacecraft can be returned successfully or not.

  The expected input an output for a spacecraft return request is similar to that of Figure 11 except that the choice is 2 instead of 1 and the returned message for a successful return is "Spacecraft returned successfully!".

- **5.3.3 (Q11) List all spacecrafts currently rented out (on a mission) with Checkout Dates within a certain period**

  The system has to provide an interface to allow a rental staff to list all unreturned spacecrafts which are rented out within a given period. After the rental staff enters the period, the program will perform the query and return a list of all unreturned spacecrafts in terms of `Agency`, `MID`, `SNum` and `Checkout Date` in descending order of checkout date within the input period inclusively.

```
Terminal  File  Edit  View  Search  Terminal  Help
-----Main menu-----
What kinds of operation would you like to perform?
1. Operations for administrator
2. Operations for exploration companies (rental customers)
3. Operations for spacecraft rental staff
0. Exit this program
Enter Your Choice: 3

-----Operations for spacecraft rental staff-----
1. Rent a spacecraft
2. Return a spacecraft
3. List all spacecraft currently rented out (on a mission) for a certain period
4. List the number of spacecrafts currently rented out by each Agency
0. Return to the main menu
Enter Your Choice: 3
Typing in the starting date [DD-MM-YYYY]: 01-01-2016
Typing in the ending date [DD-MM-YYYY]: 11-01-2018
List of the unreturned spacecraft:
|Agency|MID|SNum|Checkout Date|
|NASA|0004|2|01-01-2018|
|NASA|0012|1|12-12-2017|
|JAXA|0004|5|04-05-2017|
| ESA|0037|4|17-08-2016|
...
End of Query

-----Operations for spacecraft rental staff-----
1. Rent a spacecraft
2. Return a spacecraft
3. List all spacecraft currently rented out (on a mission) for a certain period
4. List the number of spacecrafts currently rented out by each Agency
0. Return to the main menu
Enter Your Choice:
```

Figure 12: Expected input and output while listing all unreturned spacecrafts

- **5.3.3 (Q12) List the number of spacecrafts currently rented out by each agency**

  The system has to provide an interface to allow a rental staff to list the number of spacecrafts which are rented out for each agency. After the rental staff enters the query, the program will execute the query and return the number of unreturned spacecrafts in terms of `Agency` and `number` in alphabetical order of the Agency name (sorted by Agency). An example is shown in the following figure.

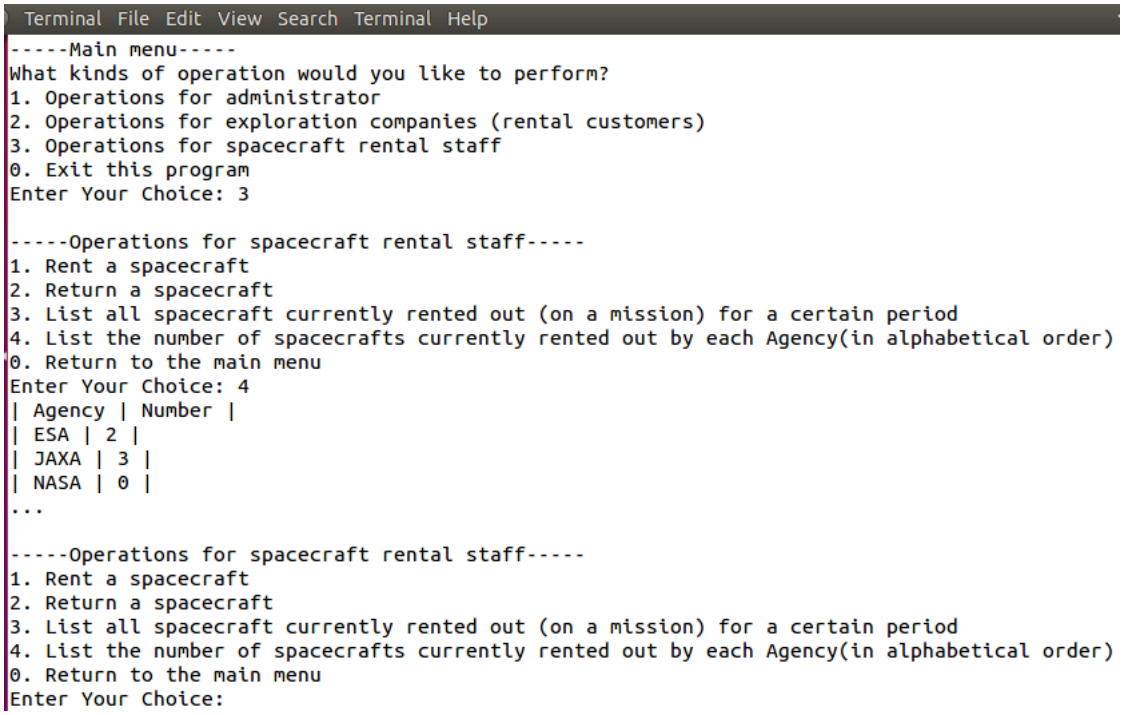

Figure 13: Expected input and output while listing the number of unreturned spacecrafts

5.4 Error Handling

If a runtime error occurs, the system should output a meaningful information message in layman terms and in a new line as shown below.

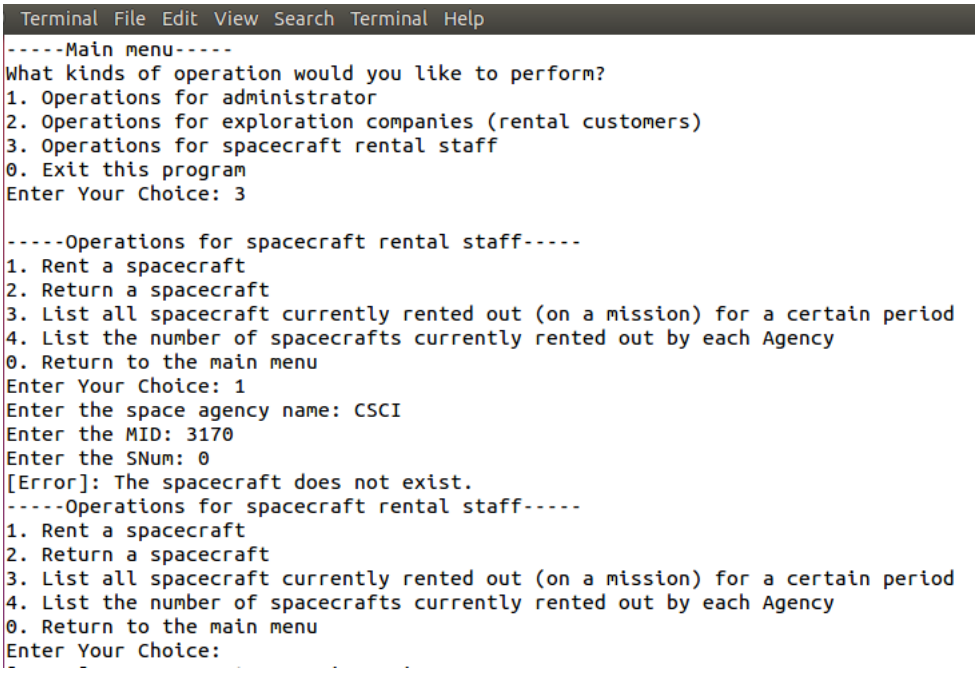

Figure 14: Expected input and output when an error occurs

Some other examples of an error message are:
When handling a spacecraft rental, and the spacecraft has not yet been returned:
"Rental not possible because the spacecraft has not yet been returned."
When handling a spacecraft rental, and the space craft does not exists:
"Rental not possible because the spacecraft is not found."

6 Tasks and Hints

The schedule of this project is set as follows:

Preparation

- Read the document thoroughly and make sure you understand all the tasks and requirements.

Group formation (Due: 29 Jan (Mon))

- This is a group project and each group should have at most three members. Please fill out the group registration form on the **Blackboard System** before the group registration deadline.

Phase 1 (Due: 14 Feb (Wed))

- According to the data specifications in **Section 3**, design an ER-diagram and transform it into a relational schema without any redundant fields and tables.
- Hint: we expect a weak entity set and a class hierarchy in the diagram.

Phase 2 (Due: 20 April (Fri))

- A suggested solution of Phase 1 will be provided on our course website after 17 Feb. According to the suggested solution of phase 1, implement a Java application with SQL statements that fulfills all requirements stated in **Section 5**. In your coding, after the dataset is loaded, it should be accessed or updated only via SQL statements.
- Each group will be assigned a MySQL account and a preferred Linux machine (one of linux1-8) for development. JDBC drivers have been installed on these machines for connections from your programs. You can also develop with other machines, we shall explain how to set up the environment in a tutorial.
- Debug your system with different datasets and user inputs.
- Write a `readme` file to describe the compilation and deployment of your system.
- Hint: you may use CREATE VIEW to create multiple temporary views to help in answering a query.

Presentation (9:00am - 6:00pm, 23, 24 April (Mon, Tue), CSE Lab: SHB˙924)

- You are asked to give a live demonstration of your system (see **Section 8** for more details).

7 Grading Policy

The marks are distributed as follows.

Table 5: Marks distribution

Phase	Content	Mark
1	ER-diagram	10%
	Relational schema (based on your ER-diagram)	10%
2	Java application	80%

- Note that each of queries Q7 and Q8 carries double the weight of any other query.
- There will be some mark deduction if your application is terminated unexpectedly during the demonstration.
- You are not allowed to modify any source code during the demonstration.
- In order to encourage every student to participate in the project, a question about this project may be set in the final examination.

8 Demonstration

- All groups need to sign up for a demonstration on their works for phase 2, the registration page would be posted on the Blackboard System later.
- All group members should attend the demonstration. Marks will be deducted for any member not showing up.
- The duration for the demonstration for each group is about 30 minutes. Each group member may need to answer a question about the project.
- The Java application will be **complied** and **tested** in a Linux 64-bit machine in the CSE department.
- The dataset used in the demonstration may be different from the dataset provided for testing.

9 Submission Methods

9.1 Phase 1

- Submit a PDF file (one copy for each group) to the collection box on the **Blackboard System**.
- The PDF file should consist of your groups ER diagram, relational schema, the group number, the names and the student IDs of all group members.

9.2 Phase 2

- Submit a `ZIP` file (one copy for each group) to the collection box on the **Blackboard System**. The `ZIP` file should consist of all your source codes and a `readme` file (`readme.txt`), which contains:
  - The group number of your group
  - The name and the student ID of each group members
  - List of files with description
  - Methods of compilation and execution

References

[1] https://en.wikipedia.org/wiki/Asteroid.

[2] AK Zimmer and Ernst Messerschmid. Going beyond: Target selection and mission analysis of human exploration missions to near-earth asteroids. *Acta Astronautica*, 69(11):1096–1109, 2011.

[3] https://upload.wikimedia.org/wikipedia/commons/f/f3/InnerSolarSystem-en.png.

[4] https://solarsystem.nasa.gov/planets/asteroids/indepth.

[5] https://solarsystem.nasa.gov/galleries/edge-on-view-of-near-earth-asteroids.