Employee administration program help file

Introduction:
The program should be started using the assignment.sh file on Linux. On Windows systems, run „starter.py“ with a python3 interpreter. An arbitrary number of .csv files can be used as parameters, using relative paths to the project folder. These files will be read and scanned for employees which will subsequently be saved in the default datasource (the program‘s own .csv file)

Once the program is started, the user can control it using the available commands.

Available commands:
help
Syntax: help [topic]
Lists all available commands if called without parameters, otherwise lists detailed information about the supplied command

exit
Syntax: exit
Terminates the program and closes the commandline interface

datasource
Syntax: datasource [datasource]
If called without parameters, displays the currently active data source. Otherwise can be called with „csv“, „db“ or „ser“ to set the current data source to the program‘s CSV file, database or serialized object storage respectively.

list
Syntax: list
Lists all employees available from the currently selected datasource

add_employee
Syntax: add_employee [employee id]
Starts the process of adding a new employee to the current datasource. If an employee already exists with the supplied id, aborts the task. Follow the on-screen instructions to add the employee.

read_csv_file
Syntax: read_csv_file [path to file] [target]
Reads employee information from the supplied CSV file. Path should be relative to the project folder. All information found will be saved in the supplied target. Use „csv“, „db“ or „ser“ to save information in the program‘s CSV file, database or serialized object storage respectively.

update_employee
Syntax: update_employee [employee id]
Starts the process to update the employee with the provided id. Will not work if no employee with this id exists.

delete_employees
Syntax: delete_employees [employee id] ...
Deletes the employees witht the provided ids from the currently active datasource. Can be called with an arbitrary number of ids.

get_info
Syntax: get_info
Displays information about the system and the available default datasources

get_statistic
Syntax: get_statistic [parameter] [group]
Displays the statistic identified by the supplied parameter and group. For example, „get_statistic salary gender“ will create a pie chart displaying the total and average salary across all genders.

Database setup
The program‘s database compatibility relies on the underlying database setup correctly. Use the information below for this:

Database type: MySQL database
Database location: localhost
Database name: bcpr301_assignment1
Login: pyaccess
password: bcpr301

Within the database defined above, a table „employees“ with the following fields should exist:
id: varchar(4)
gender: varchar(1)
age: int
sales: int
bmi: varchar(10)
salary: int
birthday: varchar(10)

Disclaimer:
Despite python‘s cross platform compatibility, this product is best used on a Linux environment and has NOT been tested on Windows, especially regarding the database and file reading functionality.