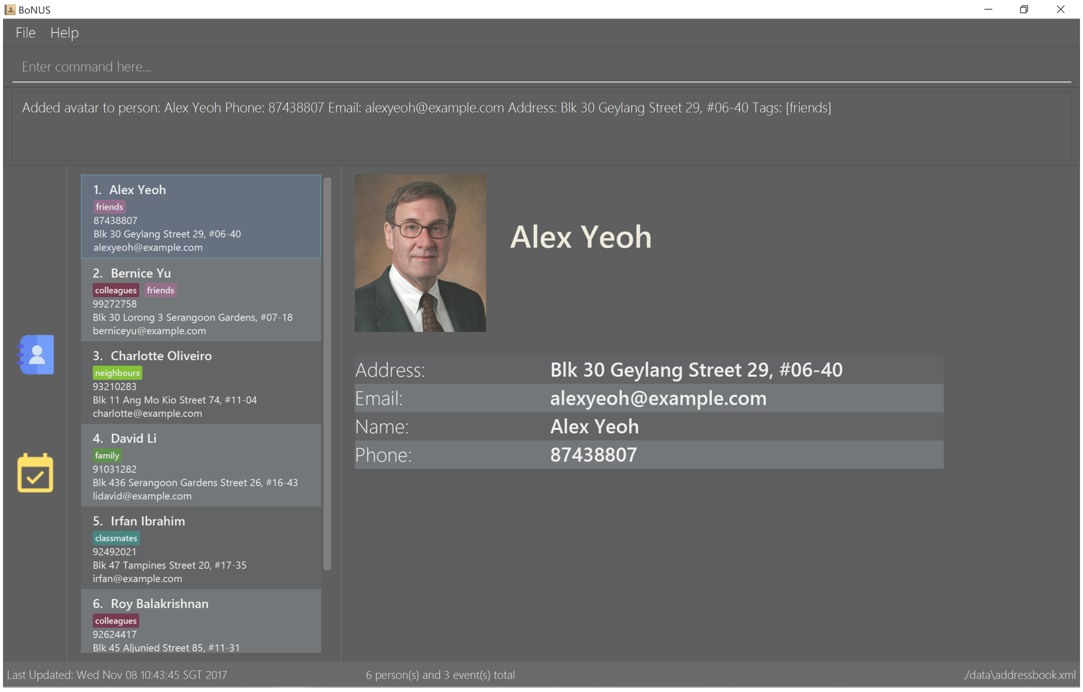
BoNUS - Test Script

By: `CS2103T T09-B3 Team` Since: `Sep 2017` Licence: `MIT`

1. Setting up
    1.1. Prerequisites
    1.2. Importing the Sample Data
    1.3. Troubleshooting
2. Testing
    2.1. Add/view/edit/delete contacts
    2.2. Add/view/edit/delete events
    2.3. Import/export data
    2.4. Email a contact on your contact list
    2.5. Searching up a contact on Google Maps
    2.6. Switch themes
    2.7. Advance setting
    2.8. Miscellaneous

1. Setting up

This test script will guide you on how to test the various features of**BoNUS**.



Added avatar to person: Alex Yeoh Phone: 87438807 Email: alexyeoh@example.com Address: Blk 30 Geylang Street 29, #06-40 Tags: [friends]

1. Alex Yeoh
friends
87438807
Blk 30 Geylang Street 29, #06-40
alexyeoh@example.com

2. Bernice Yu
colleagues friends
99272758
Blk 30 Lorong 3 Serangoon Gardens, #07-18
berniceyu@example.com

3. Charlotte Oliveiro
neighbours
93210283
Blk 11 Ang Mo Kio Street 74, #11-04
charlotte@example.com

4. David Li
family
91031282
Blk 436 Serangoon Gardens Street 26, #16-43
lidavid@example.com

5. Irfan Ibrahim
classmates
92492021
Blk 47 Tampines Street 20, #17-35
irfan@example.com

6. Roy Balakrishnan
colleagues
92624417
Blk 45 Aljunied Street 85, #11-31

Alex Yeoh

Address:	Blk 30 Geylang Street 29, #06-40
Email:	alexyeoh@example.com
Name:	Alex Yeoh
Phone:	87438807

Last Updated: Wed Nov 08 10:43:45 SGT 2017   6 person(s) and 3 event(s) total   ./data\addressbook.xml

Figure 1 : User Interface Demo

1.1. Prerequisites

1. Have Java (JRE) version `1.8.0_60` or later installed in your Computer.
2. Download the latest release `[T09-B3][BoNUS].jar`, the sample data `[T09-B3][BoNUS]SampleData.xml` and test instructions `[T09-B3][BoNUS]TestScript.pdf` (this document itself).
3. A folder you want to use as home folder for testing**BoNUS** and copy the `.jar` you have downloaded there.

1.2. Importing the Sample Data

1. Rename the downloaded `[T09-B3][BoNUS]SampleData.xml` to `addressbook.xml`.
2. Create a folder called `data` under the home folder used for testing **BoNUS**.
3. Move `addressbook.xml` into the `data` folder created just now.

1.3. Troubleshooting

If you have any doubt or need further clarification, you may want to visit ourproject website for more information.

2. Testing

2.1. Add/view/edit/delete contacts

2.1.1. Add a new contact

To add a new contact, we need to use the `add` command.

Positive test case

Let’s add a new person called `Martin` into **BoNUS**
Command:
`add n/Martin Smith p/85930167 e/martin.smith@example.com a/25 Scene Road, 119587 t/friends`
Result:
`New person added: Martin Smith Phone: 85930167 Email: martin.smith@example.com Address: 25 Scene Road, 119587 Tags: [friends]`

Negative test cases

1. The value for one property is invalid
   Command:
   `add n/Den Smith p/85930167 e/www.smith.com a/25 New York City`
   Result:
   `Person emails should be 2 alphanumeric/period strings separated by '@'.`
2. The person already exists
   Command:
   `add n/Martin Smith p/85930167 e/martin.smith@example.com a/25 Scene Road, 119587 t/friends`
   Result:
   `This person already exists in the application.`

2.1.2. Edit an existing contact

To edit an existing contact, we need to use the `edit` command.

Positive test case

1. First, let’s list all the contacts in **BoNUS**
   Command:
   `list`
   Result:
   `Listed all persons`
2. Let’s edit the person called `Martin` we added just now (assume the index is `8`)

   Command:
   ```
   edit 8 p/86129854
   ```
   Result:
   ```
   Edited Person: Martin Smith Phone: 86129854 Email: martin.smith@example.com Address: 25 Scene Road, 119587 Tags: [friends]
   ```

Negative test cases

1. The index is invalid (assume there are fewer than 100 contacts in the application)
   Command:
   ```
   edit 100 a/Beijing, China
   ```
   Result:
   ```
   The person index provided is invalid.
   ```

2.1.3. View the details of a person

To view the details of an existing contact, we need to use the `select` command.

1. First, let’s list all the contacts in **BoNUS**
   Command:
   ```
   list
   ```
   Result:
   ```
   Listed all persons
   ```
2. Then, select the person we want to view (assume there is at least 1 contact in the application)
   Command:
   ```
   select 1
   ```
   Result:
   The person’s details will be shown on the right-hand side of the interface.

2.1.4. Delete an existing person

To delete an existing contact, we need to use the `delete` command.

1. First, let’s list all the contacts in **BoNUS**
   Command:
   ```
   list
   ```
   Result:
   ```
   Listed all persons
   ```
2. Then, delete the first person (assume there is at least 1 contact in the application)
   Command:
   ```
   delete 1
   ```
   Result:
   ```
   Deleted Person: Alex Yeoh Phone: 87438807 Email: alexyeoh@example.com Address: Blk 30 Geylang Street 29, #06-40 Tags: [friends]
   ```

2.1.5. Add avatar to an existing person

Positive test case

To add avatar to a contact, we need to use the `avatar` command.

1. First, let’s list all the contacts in **BoNUS**
   Command:
   ```
   list
   ```

Result:
`Listed all persons`

2. Then move an image file (`.jpg` or `.png`) into the home folder where you use **BoNUS**. Maybe you want to use the logo of Java from [here]() as a sample.
3. Add avatar to the third person in the listing (assume there are at least three contacts)
   Command:
   `avatar 3 java-logo.png`

> - Do not move, rename or delete the image you use as an avatar; otherwise, it will not be shown in **BoNUS**.
> - You are suggested to move the image into the home folder where you use**BoNUS**.
> - Do not be *nasty* about the input file. If you provide an invalid file that looks like an image but is not an actual image, the area reserved for avatar will simply become transparent.

Negative test cases

1. Use an invalid index
   Command:
   `avatar -1 java-logo.png`
   Result:
   `Invalid command format!`
   `Index is not a non-zero unsigned integer.`
2. Use a non-existing file
   Command:
   `avatar 1 no-java.png`
   Result:
   `Invalid command format!`
   `The provided image path does not exist.`

2.2. Add/view/edit/delete events

2.2.1. Listing of events

1. First, let’s start by listing all the events in**BoNUS**
   Positive test case
   Command:
   `listE`
   Result:
   `Listed all events`

You should see that the application automatically switches from contact list to event list after the command listE.

Negative test case

1. Incorrect capitalization of command
   Command:
   `ListE`
   Result:

```
    Unknown command
    ```

2. Invalid spelling
   Command:
   ```
   listEE
   ```
   Result:
   ```
   Unknown command
   ```

Commands are case sensitive.

2.2.2. Adding of events

> - The standard format for time should be `DDMMYYYY HH:MM` in 24-hour format.
> - However, the application may sometimes be *smart* enough to interpret what you typed. For example, if you type `this afternoon` or `tomorrow evening`, it will be automatically converted to the standard format.
> - Do NOT try to *challenge* the application, it is not promised to produce an expected result if you use non-standard format or if the expression is not simple or clear enough.
> - Avoid putting the year first when using the non-standard format.

1. Let's start by adding an Event called `CS3201 Project Meeting` into **BoNUS**.
   Command:
   ```
   addE n/CS3201 Project Meeting dt/17112017 15:30 a/School Of Computing, SR09, NUS
   ```
   Result:
   ```
   New event added: Event: CS3201 Project Meeting | Date/Time: 17 Nov, 2017 15:30 | Address:
   School Of Computing, SR09, NUS
   ```
   You should see a new Event Card with the name CS3201 Project Meeting added into **BoNUS**.

2.2.3. Deleting of Events

Positive Test Case

1. Deletes an Event with index 8 on the Event List
   Command:
   ```
   deleteE 8
   ```
   Result:
   ```
   Deleted Event: Event: Project Meeting for CS2101 | Date/Time: 01 Dec, 2017 09:00 | Address:
   SR09, School of Computing
   ```

Negative Test Cases

1. Invalid index
   Command:
   ```
   deleteE -1
   ```
   Result:
   ```
   Invalid command format! deleteE: Deletes the event identified by the index number used in
   the last event listing. Parameters: INDEX (must be a positive integer)
   ```

2. Person with the index not exist
   2) Deletes an Event with index 11 on the Event List
   Command:
   ```
   deleteE 300
   ```

Result:
`The event index provided is invalid`

2.2.4. Editing of Events

Positive test case

1. First, let’s list all the events in **BoNUS**
   Command:
   `listE`
   Result:
   `Listed all events`
2. Let’s try to edit an event from index 6 of Event list.
   Command:
   `editE 6 n/Movie Date with Bro`
   Result:
   `Edited Event: Event: Movie Date with Bro | Date/Time: 22 Nov, 2017 22:00 | Address: Golden Village Yishun`

At this point in time, you should see an event with name `Movie Date with Bro` at index 6 of Event List.

2.3. Import/export data

2.3.1. Import data from `.xml` format

To import data from a `.xml` file, we need to use the `import` command

Positive test cases

1. If the XML file at `<BoNUS home folder>/bonus.xml` contains data recognizable by **BoNUS**, this command imports data that does not exist in **BoNUS**
   Command:
   `import bonus.xml`
   Result:
   `Imported data from: bonus.xml`

Negative test cases

1. If the specified file is not found
   Command:
   `import bonus.xml`
   Result:
   `The specified file does not exist.`
2. If the specified file does not end with an extension of `.xml`
   Command:
   `import bonus.notxml`
   Result:
   `The file path does not point to an XML file.`
3. If a file name is not provided
   Command:
   `import .xml`
   Result:
   `The file path contains a file name or folder names with prohibited characters (?!%*+:|"<>),`

`or a file name is not provided.`

4. If the specified file name and any non-existent folder names contain any prohibited characters
   Command:
   `import b%nus.xml`
   Result:
   `The file path contains a file name or folder names with prohibited characters (?!%*+:|"<>), or a file name is not provided.`

5. For `Windows` users, if the specified file path contains `/` as a name-separator
   Command:
   `import folder/bonus.xml`
   Result:
   `The file path contains name-separators (/ or \) that are not defined in your operating system.`

6. For `maxOS` and `Linux` users, if the specified file path contains `\` as a name-separator
   Command:
   `import folder\bonus.xml`
   Result:
   `The file path contains name-separators (/ or \) that are not defined in your operating system.`

7. For `Windows` users, if the specified file path contains consecutive name-separators `\\`
   Command:
   `import folder\\bonus.xml`
   Result:
   `The file path contains consecutive name-separators (/ or \) or extension-separators (.).`

8. For `maxOS` and `Linux` users, if the specified file path contains consecutive name-separators `//`
   Command:
   `import folder//bonus.xml`
   Result:
   `The file path contains consecutive name-separators (/ or \) or extension-separators (.).`

9. If the specified file path contains consecutive extension-separators `..`
   Command:
   `import bonus..xml`
   Result:
   `The file path contains consecutive name-separators (/ or \) or extension-separators (.).`

10. If the data in the specified file is not recognizable by**BoNUS**
    Command:
    `import bonus.xml`
    Result:
    `The data in the specified XML file is not in the correct format as recognized by BoNUS.`
    `Please try to import from another XML file.`

2.3.2. Import timetable from NUSMods website

To import timetable from NUSMods website, we need to use the `import --nusmods` command.

> Make sure you have stable Internet connection when you are going to test this command.

Positive test cases

1. Open your browser, and copy-paste your NUSMods timetable URL
   Result:
   It should look like https://nusmods.com/timetable/2017-2018/sem1?CS2103T[TUT]=C01.

2. Let's display the event listing first
   Command:
   `listE`
   Result:
   You should see the event listing interface.

3. Use the import command
   Command:
   `import --nusmods https://nusmods.com/timetable/2017-2018/sem1?CS2103T[TUT]=C01`
   Result:
   You should see `CS2103T examination` has been added as an event into **BoNUS**.

Negative test case

Try to import the same NUSMods timetable into **BoNUS** again
Command:
`import --nusmods https://nusmods.com/timetable/2017-2018/sem1?CS2103T[TUT]=C01`
Result:

```
0 examinations have been added as events.
However, some examination were not added since they already exist in the application.
```

2.3.3. Export data to `.xml` format

To export data to a `.xml` file, we need to use the `export` command

> Existing files will be overwritten.

Positive test cases

1. Exports the current data in **BoNUS** to `<BoNUS home folder>/bonus.xml`
   Command:
   `export bonus.xml`
   Result:
   `Exported address book data to: bonus.xml`

2. For `Windows` users, this command exports the current data in **BoNUS** to `<BoNUS home folder>\new_folder\bonus.xml`, after creating the non-existing folder `<BoNUS home folder>\new_folder`
   Command:
   `export new_folder\bonus.xml`
   Result:
   `Exported address book data to: new_folder\bonus.xml`

3. For `macOS` and `Linux` users, this command exports the current data in **BoNUS** to `<BoNUS home folder>/new_folder/bonus.xml`, after creating the non-existing folder `<BoNUS home folder>/new_folder`
   Command:
   `export new_folder/bonus.xml`
   Result:
   `Exported address book data to: new_folder/bonus.xml`

Negative test cases

1. If the specified file does not end with an extension of `.xml`

   Command:

   ```
   export bonus.notxml
   ```

   Result:

   ```
   The file path does not point to an XML file.
   ```

2. If a file name is not provided

   Command:

   ```
   export .xml
   ```

   Result:

   ```
   The file path contains a file name or folder names with prohibited characters (?!%*+:|"<>), or a file name is not provided.
   ```

3. If the specified file name and any non-existent folder names contain any prohibited characters

   Command:

   ```
   export b%nus.xml
   ```

   Result:

   ```
   The file path contains a file name or folder names with prohibited characters (?!%*+:|"<>), or a file name is not provided.
   ```

4. For `Windows` users, if the specified file path contains `/` as a name-separator

   Command:

   ```
   export folder/bonus.xml
   ```

   Result:

   ```
   The file path contains name-separators (/ or \) that are not defined in your operating system.
   ```

5. For `maxOS` and `Linux` users, if the specified file path contains `\` as a name-separator

   Command:

   ```
   export folder\bonus.xml
   ```

   Result:

   ```
   The file path contains name-separators (/ or \) that are not defined in your operating system.
   ```

6. For `Windows` users, if the specified file path contains consecutive name-separators `\\`

   Command:

   ```
   export folder\\bonus.xml
   ```

   Result:

   ```
   The file path contains consecutive name-separators (/ or \) or extension-separators (.).
   ```

7. For `maxOS` and `Linux` users, if the specified file path contains consecutive name-separators `//`

   Command:

   ```
   export folder//bonus.xml
   ```

   Result:

   ```
   The file path contains consecutive name-separators (/ or \) or extension-separators (.).
   ```

8. If the specified file path contains consecutive extension-separators `..`

   Command:

   ```
   export bonus..xml
   ```

   Result:

   ```
   The file path contains consecutive name-separators (/ or \) or extension-separators (.).
   ```

2.4. Email a contact on your contact list

To email anyone on your contact list, you can use the `email` command.

Positive test case

1. First, let’s list all the contacts in **BoNUS**
   Command:
   `list`
   Result:
   `Listed all persons`
2. Emailing Alex Yeoh
   Command:
   `email 1`
   Result:
   `Opened email to Alex Yeoh Phone: 87438807 Email: alexyeoh@example.com Address: Blk 30 Geylang Street 29, #06-40 Tags: [friends]`

At this point in time, you should see the default email application of your OS to be opened with the email of Alex already input at To:

Negative test cases

1. Invalid command and Invalid index
   Command:
   `email -1`
   Result:
   `Invalid command format! The person index provided is invalid`.
2. Invalid index *Command:*
   `email 300000`
   Result:
   `The person index provided is invalid`.

2.5. Searching up a contact on Google Maps

To search up a person’s address on Google Maps, you can use the `gmap` command

Positive test case

1. First, let’s list all the contacts in **BoNUS**
   Command:
   `list`
   Result:
   `Listed all persons`
2. Next, let’s try to search up a person’s address on Google Maps!
   Command:
   `gmap 1`
   Result:
   `Opened Google Maps to get to Alex Yeoh Phone: 87438807 Email: alexyeoh@example.com Address: Blk 30 Geylang Street 29, #06-40 Tags: [friends]`

At this point in time, you should see a browser opening Google Maps with `Alex Yeoh` 's address.

Negative test cases

1. Invalid command and invalid index
   Command:
   `gmap -1`
   Result:
   `gmap: Opens Google Maps in default browser with the address of the person identified by the index number used in the last person listing being the Destination. Parameters: INDEX (must be a positive integer)`

2. Invalid index *Command:*
   `gmap 300000`
   Result:
   `The person index provided is invalid`

2.6. Switch themes

To switch themes, we need to use the `theme` command.

Positive test case

1. Let's try to switch themes *Command:*
   `theme`
   Result:
   `Theme switched!`

At this point in time, you should see the theme being switched from dark to bright theme.

2.7. Advance setting

2.7.1. Add a new property

To add a new property, we need to use the `config --add-property` command.

Positive test cases

1. Add a new property with the short name of `h` and full name of `hobby`
   Command:
   `config --add-property s/h f/hobby`
   Result:
   `Added a new property: s/h f/hobby`

2. Adds a new property with the short name of `m`, full name of `major` and specify the regular expression constraint.
   Command:
   `config --add-property s/m f/major m/Major should only contain letters and it should not be blank r/[{Alpha}]+`
   Result:
   `Added a new property: s/m f/major m/Major should only contain letters and it should not be blank r/[{Alpha}]+`

3. Set the value of customize property added just now in `AddCommand`
   Command:
   `add n/Linda p/89658712 e/linda@example.com a/School of Computing h/Programming`
   Result:
   `New person added: Linda Phone: 89658712 Email: linda@example.com Address: School of Computing Tags:`
   In the meantime, a new person card belonging to Linda should appear in the person listing panel.

4. Verify the value of hobby has been set correctly
   Command:
   `find Linda`
   `select 1`
   Result:
   Linda’s personal details are shown on the right-hand side of the interface. Properties are ordered by their full names and each property takes one line. You should see the value of the row `Hobby` is `Programming`.

Negative test cases

1. If there already exists a property with the same short name
   Command:
   `config --add-property s/a f/attitude`
   Result:
   `Another property with the same short name already exists in the application.`
2. If the regular expression provided is invalid
   Command:
   `config --add-property s/w f/weight m/Weight must be a positive integer. r/*asf`
   Result:
   `The regular expression you provided is invalid.`

2.7.2. Change the color of an existing tag

To change the color of an existing tag, we need to use the `config --set-tag-color` command.

Positive test cases

1. First, let’s list all the contacts to see what tags exist now
   Command:
   `list`
   Result:
   You should see all contacts stored in **BoNUS** are listed. Let’s assume there is a tag called `friends`.
2. Change the color of `friends` tag to a pre-defined color name
   Command:
   `config --set-tag-color friends BurlyWood`
   Result:
   You should notice the color of `friends` tag has been changed to BurlyWood.
3. Change the color of `friends` tag to a valid RGB value
   Command:
   `config --set-tag-color friends #FF6347`
   Result:
   You should notice the color of `friends` tag has been changed to a color similar to red (its legal name is tomato).

Negative test cases

1. If the tag does not exist in **BoNUS** yet
   Command:
   `config --set-tag-color somethingNotExists blue`
   Result:

```
There is no such tag.
```

2. If the provided color name is neither a valid color name defined in CSS standard nor a valid RGB value
   Command:
   ```
   config --set-tag-color friends someNonExistColor
   ```
   Result:
   The background color of `friends` tag becomes transparent. You may want to use a valid name to change it back.

2.8. Miscellaneous

2.8.1. View the help window

1. Via command-line input
   Command:
   ```
   help
   ```
   Result:
   The help window should appear and display user guide to you.

2. Via F1
   Command:
   ⌨ Press the `F1` key on your keyboard. *Result:*
   The help window should appear and display user guide to you.

3. Via mouse
   Command:
   Click `Help` → `Help` on the menu bar at the top of the interface.
   Result:
   The help window should appear and display user guide to you.

2.8.2. View the command input history

Use the history command to view the history of all input commands
Command:

```
history
```

Result:

```
Entered commands (from most recent to earliest):
…
```

(All commands will be listed below.)

2.8.3. Undo/redo previous commands

Use `undo` or `redo` to undo/redo the last command entered.

Last updated 2017-11-13 14:42:52 +00:00