SHIFT

BACK END

DOCUMENTATION

Table of Contents

Variable References (CASE MATTERS):....8
- Strings....8
- Booleans....8
- Gender Enums....8
- Role Enums....8
- Hall Enums....8
- DutyLevel Enums....9
- Time Zones....9
- Dates....9

Resident Assistant Links....11
- http://localhost:8080/ra/{id}....11
- http://localhost:8080/ra/{id}/delete....11
- http://localhost:8080/ra/{id}/set-name....12
- http://localhost:8080/ra/{id}/set-email....12
- http://localhost:8080/ra/{id}/set-gender....13
- http://localhost:8080/ra/{id}/set-hall....13
- http://localhost:8080/ra/{id}/enable....14
- http://localhost:8080/ra/{id}/disable....14
- http://localhost:8080/ra/{id}/set-floor....14
- http://localhost:8080/ra/{id}/set-timezone....15
- http://localhost:8080/ra/{id}/set-all....16
- http://localhost:8080/ra/{id}/create-account....17
- http://localhost:8080/ra/{id}/clock-in....17
- http://localhost:8080/ra/{id}/clock-out....18
- http://localhost:8080/ra/{id}/clear-preferences....18
- http://localhost:8080/ra/{id}/add-preference....19
- http://localhost:8080/ra/{id}/get-preferences....19
- http://localhost:8080/ra/{id}/get-events....20
- http://localhost:8080/ra/{id}/get-shifts....21
- http://localhost:8080/ra/{id}/get-meetings....22
- http://localhost:8080/ra/{id}/delete-event/{eventId}....23
- http://localhost:8080/ra/{id}/delete-event-day....24

http://localhost:8080/ra/{id}/delete-event-week....25
http://localhost:8080/ra/{id}/delete-event-month....26
http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-title....27
http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-description....27
http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-duty-level....28
http://localhost:8080/ra/{id}/add-event/{eventId}....29
http://localhost:8080/ra/{id}/add-shift/{eventId}....31
http://localhost:8080/ra/{id}/request-drop/{eventId}....31
http://localhost:8080/ra/{id}/request-drop-delete/{eventId}....33
http://localhost:8080/ra/{id}/find-ra-in-hall....34
http://localhost:8080/ra/{id}/find-rea-in-hall....35
http://localhost:8080/ra/{id}/find-rec-in-hall....36
http://localhost:8080/ra/{id}/report-ra/{raId}....36
http://localhost:8080/ra/{id}/report-rea/{reaId}....37
Resident Education Assistant Links....38
http://localhost:8080/rea/{id}/delete....38
http://localhost:8080/rea/{id}/delete-ra/{raId}....38
http://localhost:8080/rea/{id}/set-name....38
http://localhost:8080/rea/{id}/set-email....39
http://localhost:8080/rea/{id}/set-gender....39
http://localhost:8080/rea/{id}/set-hall....40
http://localhost:8080/rea/{id}/is-ra-clocked-in/{raId}....40
http://localhost:8080/rea/{id}/is-ra-enabled/{raId}....41
http://localhost:8080/rea/{id}/enable....41
http://localhost:8080/rea/{id}/enable-ra/{raId}....41
http://localhost:8080/rea/{id}/disable....42
http://localhost:8080/rea/{id}/disable-ra/{raId}....42
http://localhost:8080/rea/{id}/set-floor....42
http://localhost:8080/rea/{id}/set-timezone....43
http://localhost:8080/rea/{id}/set-all....43
http://localhost:8080/rea/{id}/create-account....44
http://localhost:8080/rea/{id}/create-welcome-message....45
http://localhost:8080/rea/{id}/get-welcome-message....45

http://localhost:8080/rea/{id}/delete-welcome-message 46
http://localhost:8080/rea/{id}/add-ra/{raId} 46
http://localhost:8080/rea/{id}/remove-ra/{raId} 46
http://localhost:8080/rea/{id}/get-ras 47
http://localhost:8080/rea/{id}/assign-shifts 47
http://localhost:8080/rea/{id}/assign-shifts-randomly 48
http://localhost:8080/rea/{id}/assign-shifts-automatically 48
http://localhost:8080/rea/{id}/shift-complete/{raId} 49
http://localhost:8080/rea/{id}/ra-primary/{raId} 49
http://localhost:8080/rea/{id}/ra-secondary/{raId} 50
http://localhost:8080/rea/{id}/ra-tertiary/{raId} 50
http://localhost:8080/rea/{id}/ra-stats/{raId} 50
http://localhost:8080/rea/{id}/ra/{raId}/get-events 51
http://localhost:8080/rea/{id}/ra/{raId}/get-shifts 51
http://localhost:8080/rea/{id}/ra/{raId}/delete-event/{eventId} 51
http://localhost:8080/rea/{id}/ra/{raId}/delete-event-day 51
http://localhost:8080/rea/{id}/ra/{raId}/delete-event-week 51
http://localhost:8080/rea/{id}/ra/{raId}/delete-event-month 51
http://localhost:8080/rea/{id}/ra/{raId}/edit-event/{eventId}/edit-title 52
http://localhost:8080/rea/{id}/ra/{raId}/edit-event/{eventId}/edit-description 52
http://localhost:8080/rea/{id}/ra/{raId}/edit-event/{eventId}/edit-duty-level 52
http://localhost:8080/rea/{id}/ra/{raId}/add-event/{eventId} 52
http://localhost:8080/rea/{id}/ra/{raId}/add-shift/{eventId} 52
http://localhost:8080/rea/{id}/ra/{raId}/view-drop-requests 53
http://localhost:8080/rea/{id}/ra/{raId}/deny-drop/{eventId} 53
http://localhost:8080/rea/{id}/ra/{raId}/delete-shift/{eventId} 54
http://localhost:8080/rea/{id}/create-shift/{eventId} 55
http://localhost:8080/rea/{id}/get-events 57
http://localhost:8080/rea/{id}/get-shifts 58
http://localhost:8080/rea/{id}/get-meetings 59
http://localhost:8080/rea/{id}/get-availability/{eventId} 60
http://localhost:8080/rea/{id}/create-event/{eventId} 61
http://localhost:8080/rea/{id}/delete-event/{eventId} 63

http://localhost:8080/rea/{id}/delete-event-day 63
http://localhost:8080/rea/{id}/delete-event-week 64
http://localhost:8080/rea/{id}/delete-event-month 65
http://localhost:8080/rea/{id}/find-ra-in-hall 66
http://localhost:8080/rea/{id}/find-rea-in-hall 67
http://localhost:8080/rea/{id}/find-rec-in-hall 68
http://localhost:8080/rea/{id}/report-ra/{raId} 68
http://localhost:8080/rea/{id}/false-report-ra/{raId} 69
http://localhost:8080/rea/{id}/clear-reports-ra/{raId} 69
http://localhost:8080/rea/{id}/report-rea/{reaId} 69
http://localhost:8080/rea/{id}/report-rec/{recId} 70
Resident Education Coordinator Links 71
http://localhost:8080/rec/{id}/delete 71
http://localhost:8080/rec/{id}/delete-rea/{reaId} 71
http://localhost:8080/rec/{id}/delete-ra/{raId} 71
http://localhost:8080/rec/{id}/set-name 72
http://localhost:8080/rec/{id}/set-email 72
http://localhost:8080/rec/{id}/set-gender 73
http://localhost:8080/rec/{id}/set-hall 73
http://localhost:8080/rec/{id}/is-ra-clocked-in/{raId} 74
http://localhost:8080/rec/{id}/is-ra-enabled/{raId} 74
http://localhost:8080/rec/{id}/enable 74
http://localhost:8080/rec/{id}/enable-ra/{raId} 75
http://localhost:8080/rec/{id}/disable 75
http://localhost:8080/rec/{id}/disable-ra/{raId} 75
http://localhost:8080/rec/{id}/set-floor 75
http://localhost:8080/rec/{id}/set-timezone 76
http://localhost:8080/rec/{id}/set-all 77
http://localhost:8080/rec/{id}/create-account 78
http://localhost:8080/rec/{id}/create-welcome-message 78
http://localhost:8080/rec/{id}/get-welcome-message 79
http://localhost:8080/rec/{id}/delete-welcome-message 79
http://localhost:8080/rec/{id}/add-ra/{raId} 79

http://localhost:8080/rec/{id}/add-rea/{reaId} 80
http://localhost:8080/rec/{id}/remove-ra/{raId} 80
http://localhost:8080/rec/{id}/remove-rea/{reaId} 80
http://localhost:8080/rec/{id}/get-ras 81
http://localhost:8080/rec/{id}/get-reas 81
http://localhost:8080/rec/{id}/assign-shifts 82
http://localhost:8080/rec/{id}/assign-shifts-randomly 82
http://localhost:8080/rec/{id}/assign-shifts-automatically 82
http://localhost:8080/rec/{id}/shift-complete/{raId} 83
http://localhost:8080/rec/{id}/ra-primary/{raId} 83
http://localhost:8080/rec/{id}/ra-secondary/{raId} 84
http://localhost:8080/rec/{id}/ra-tertiary/{raId} 84
http://localhost:8080/rec/{id}/ra-stats/{raId} 84
http://localhost:8080/rec/{id}/ra/{raId}/get-events 85
http://localhost:8080/rec/{id}/ra/{raId}/get-shifts 85
http://localhost:8080/rec/{id}/ra/{raId}/delete-event/{eventId} 85
http://localhost:8080/rec/{id}/ra/{raId}/delete-event-day 85
http://localhost:8080/rec/{id}/ra/{raId}/delete-event-week 85
http://localhost:8080/rec/{id}/ra/{raId}/delete-event-month 85
http://localhost:8080/rec/{id}/ra/{raId}/edit-event/{eventId}/edit-title 86
http://localhost:8080/rec/{id}/ra/{raId}/edit-event/{eventId}/edit-description 86
http://localhost:8080/rec/{id}/ra/{raId}/edit-event/{eventId}/edit-duty-level 86
http://localhost:8080/rec/{id}/ra/{raId}/add-event/{eventId} 86
http://localhost:8080/rec/{id}/ra/{raId}/add-shift/{eventId} 86
http://localhost:8080/rec/{id}/ra/{raId}/view-drop-requests 87
http://localhost:8080/rec/{id}/ra/{raId}/deny-drop/{eventId} 87
http://localhost:8080/rec/{id}/ra/{raId}/delete-shift/{eventId} 88
http://localhost:8080/rec/{id}/create-shift/{eventId} 89
http://localhost:8080/rec/{id}/get-events 91
http://localhost:8080/rec/{id}/get-shifts 92
http://localhost:8080/rec/{id}/get-meetings 93
http://localhost:8080/rec/{id}/get-availability/{eventId} 94
http://localhost:8080/rec/{id}/create-event/{eventId} 95

http://localhost:8080/rec/{id}/delete-event/{eventId} 97
http://localhost:8080/rec/{id}/delete-event-day 97
http://localhost:8080/rec/{id}/delete-event-week 98
http://localhost:8080/rec/{id}/delete-event-month 99
http://localhost:8080/rec/{id}/find-ra-in-hall 100
http://localhost:8080/rec/{id}/find-rea-in-hall 101
http://localhost:8080/rec/{id}/find-rec-in-hall 102
http://localhost:8080/rec/{id}/report-ra/{raId} 102
http://localhost:8080/rec/{id}/false-report-ra/{raId} 103
http://localhost:8080/rec/{id}/clear-reports-ra/{raId} 103
http://localhost:8080/rec/{id}/report-rea/{reaId} 103
http://localhost:8080/rec/{id}/false-report-rea/{reaId} 104
http://localhost:8080/rec/{id}/clear-reports-rea/{reaId} 104
http://localhost:8080/rec/{id}/report-rec/{recId} 104
http://localhost:8080/rec/{id}/false-report-rec/{recId} 105
http://localhost:8080/rec/{id}/clear-reports-rec/{recId} 105

Variable References (CASE MATTERS):

Strings

String can contain any commonly used character **EXCEPT** for spaces (“ ”), tabs/indentations (“\t”), new lines (“\n”), and **PIPES** (“|”).

Booleans

The string representation for booleans must be any of the following:

```
true
false
```

Gender Enums

The string representation for Gender enums must be any of the following:

```
MALE
FEMALE
NONBINARY
```

Role Enums

The string representation for Role enums must be any of the following:

```
RA
REA
REC
```

Hall Enums

The string representation for Hall enum must be any of the following:

```
/* All Male */
CARY
MCCUTCHEON
TARKINGTON

/* All Female */
MEREDITH
```

```
MEREDITH_SOUTH
WINDSOR

/* Co-Ed */
EARHART
FIRST_STREET_TOWERS
FRIEDA_PARKER
WINIFRED_PARKER
HARRISON
HAWKINS
HILLENBRAND
HONORS_COLLEGE
OWEN
SHREVE
WILEY

/* Apartments */
URBA
```

DutyLevel Enums

The string representation for DutyLevel enums must be any of the following:

```
/* For Shifts Only*/
PRIMARY
SECONDARY
TERTIARY

/* For events */
null
```

Time Zones

The string representation for time zones is any integer representing the number of hours away from UTC:

Integer from [-12, 14] inclusive (default is 0).

Dates

Any dates related times such as day, month, or year are any valid integer as seen on a calendar

day: Integer from [1, 31]* inclusive (* depending on the month).

month: Integer from [1, 12] inclusive.

year: Integer from [2024, INTEGER_MAX - 1] inclusive.

Resident Assistant Links

http://localhost:8080/ra/{id}

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Returns RA information as a string with string in JSON formatting, as seen below.

```
{
“name“ : “...”,
“email“ : “...”,
“id“ : “...”,
“gender“ : “...”,
“role“ : “...”,
“hall“ : “...”,
“enabled“ : “...”,
“timezone“ : “...”,
“floor“ : “...”,
“clockedIn“ : “...”
}
```

http://localhost:8080/ra/{id}/delete

Input:

The “id” associated with the RA. This is included in the link for the desired RA to be deleted.

Output:

Deletes the RA with the given id. Returns string with the message:

“Deleted User: [id]”

http://localhost:8080/ra/{id}/set-name

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The “name” to be set for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“name” : “...”
}
```

Output:

Changes the name of the RA with the given id. Returns string with the message:

“Name Edited”

http://localhost:8080/ra/{id}/set-email

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The “email” to be set for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“email” : “...”
}
```

Output:

Changes the email of the RA with the given id. Returns string with the message:

“Email Edited”

http://localhost:8080/ra/{id}/set-gender

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The “gender” to be set for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“gender” : “...”
}
```

Output:

Changes the gender of the RA with the given id. Returns string with the message:

“Gender Edited”

http://localhost:8080/ra/{id}/set-hall

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The “hall” to be set for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“hall” : “...”
}
```

Output:

Changes the hall of the RA with the given id. Returns string with the message:

“Hall Edited”

http://localhost:8080/ra/{id}/enable

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Enables the account of the RA with the given id.

http://localhost:8080/ra/{id}/disable

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Disables the account of the RA with the given id.

http://localhost:8080/ra/{id}/set-floor

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The “floor” to be set for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“floor” : “...”
}
```

Output:

Changes the floor of the RA with the given id. Returns string with the message:

“Floor Edited”

http://localhost:8080/ra/{id}/set-timezone

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The “timezone” to be set for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“timezone” : “...”
}
```

Output:

Changes the floor of the RA with the given id. Returns string with the message:

“Timezone Edited”

http://localhost:8080/ra/{id}/set-all

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

All basic attributes to be set for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“name“ : “...”,
“email“ : “...”,
“gender“ : “...”,
“hall“ : “...”,
“enabled“ : “...”,
“floor“ : “...”
}
```

Output:

Sets basic attributes RA with the given id. Returns string with the message:

“Basic Attributes Set”

http://localhost:8080/ra/{id}/create-account

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Minimum number of attributes to be set for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“name“ : “...”,
“email“ : “...”,
“enabled“ : “...”
}
```

Output:

Sets minimum number of attributes to create an RA account with the given id. Returns string with the message:

“Basic Attributes Set”

http://localhost:8080/ra/{id}/clock-in

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Clocks in the RA with the given id. Returns string with the message:

“Clocked In”

http://localhost:8080/ra/{id}/clock-out

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Clocks out the RA with the given id. Returns string with the message:

“Clocked Out”

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Clocks out the RA with the given id. Returns string with the message:

“Clocked Out”

http://localhost:8080/ra/{id}/clear-preferences

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Clears list of preferences for RA with the given id. Returns string with the message:

“Preferences Cleared”

http://localhost:8080/ra/{id}/add-preference

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

String called “preference” to be added for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“preference” : “...”
}
```

Output:

Sets basic attributes RA with the given id. Returns string with the message:

“Preference Added”

http://localhost:8080/ra/{id}/get-preferences

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Sets basic attributes RA with the given id. Returns string array with the preferences formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
“preference” : “...”
}
```

http://localhost:8080/ra/{id}/get-events

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Returns all the events of the RA with the given id. Returns string array with the events formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/ra/{id}/get-shifts

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Returns all the shifts of the RA with the given id. Returns string array with the shifts formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/ra/{id}/get-meetings

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

Output:

Returns all the meetings of the RA with the given id. Returns string array with the shifts formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/ra/{id}/delete-event/{eventId}

Input:

The “id” associated with the RA as well as the “eventId” of the event to be deleted. These are included in the link for the desired RA.

Output:

Deletes event with the given eventId in the RA with the given id. Returns string with the message:

“Deleted: [eventId]”

http://localhost:8080/ra/{id}/delete-event-day

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The starting “day” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The starting “month” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The “timezone” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

```
{
"day" : "..."
"month" : "..."
"timezone" : "..."
}
```

Output:

Deletes events on starting day in the RA with the given id. Returns string with the message:

“Deleted Day”

http://localhost:8080/ra/{id}/delete-event-week

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The starting “day” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The starting “month” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The “year” of the starting day events to be deleted. This is given as a string in JSON formatting, as seen below.

The “timezone” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

```
{
"day" : "..."
"month" : "..."
"year" : "..."
"timezone" : "..."
}
```

Output:

Deletes all events up until a week after the starting day in the RA with the given id. Returns string with the message:

“Deleted Week”

http://localhost:8080/ra/{id}/delete-event-month

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The starting “month” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The “year” of the starting day events to be deleted. This is given as a string in JSON formatting, as seen below.

The “timezone” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

```
{
"month" : "..."
"year" : "..."
"timezone" : "..."
}
```

Output:

Deletes all events for the entire month day in the RA with the given id. Returns string with the message:

“Deleted Month”

http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-title

Input:

The “id” associated with the RA. The “eventId” to have its title edited. This is included in the link for the desired RA.

The new “title” to be used in the current event for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
"title" : "..."
}
```

Output:

Changes the title of the given event belonging to the RA with the given id. Returns string with the message:

“Edited Title: [title]”

http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-description

Input:

The “id” associated with the RA. The “eventId” to have its description edited. This is included in the link for the desired RA.

The new “description” to be used in the current event for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
"description" : "..."
}
```

Output:

Changes the description of the given event belonging to the RA with the given id. Returns string with the message:

“Edited Description: [description]”

http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-duty-level

Input:

The “id” associated with the RA. The “eventId” to have its duty level edited. This is included in the link for the desired RA.

The new “dutyLevel” to be used in the current event for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
"dutyLevel" : "..."
}
```

Output:

Changes the duty level of the given event belonging to the RA with the given id. Returns string with the message:

“Edited Description: [dutyLevel]”

http://localhost:8080/ra/{id}/add-event/{eventId}

Input:

The “id” associated with the RA. The “eventId” to be added. This is included in the link for the desired RA.

The new event to be added for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

Output:

Outputs all the current events belonging to the RA with the given id. Returns string with the message:

For String[0], ..., String[n] =>

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
```

```
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

http://localhost:8080/ra/{id}/add-shift/{eventId}

Input:

The “id” associated with the RA. The “eventId” to be added. This is included in the link for the desired RA.

This function assumes that this RA has already been assigned an REA.

Output:

Outputs all the current shifts belonging to the RA with the given id. This is given as a string in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

http://localhost:8080/ra/{id}/request-drop/{eventId}

Input:

The “id” associated with the RA. The “eventId” to be added. This is included in the link for the desired RA.

This adds a drop request to the assigned REA to request a shift with the “eventId” to be dropped.

Output:

Outputs all the current shifts belonging to the RA with the given id.
Returns string formatted as a JSON:

For String[0], ..., String[n] =>

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

http://localhost:8080/ra/{id}/request-drop-delete/{eventId}

Input:

The “id” associated with the RA. The “eventId” to be added. This is included in the link for the desired RA.

This removes a drop request of an “eventId” to the assigned REA that was requested to be dropped.

Output:

Outputs all the current shifts belonging to the RA with the given id. Returns string with the message:

For String[0], ..., String[n] =>

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

http://localhost:8080/ra/{id}/find-ra-in-hall

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The new “hall” to be searched. This is given as a string in JSON formatting, as seen below.

```
{
"hall" : "..."
}
```

Output:

Returns an array of the RA ids that match the given hall.
This is given as an array of strings each in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"raId" : "..."
}
```

http://localhost:8080/ra/{id}/find-rea-in-hall

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The new “hall” to be searched. This is given as a string in JSON formatting, as seen below.

```
{
"hall" : "..."
}
```

Output:

Returns an array of the REA ids that match the given hall.
This is given as an array of strings each in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"reaId" : "..."
}
```

http://localhost:8080/ra/{id}/find-rec-in-hall

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The new “hall” to be searched. This is given as a string in JSON formatting, as seen below.

```
{
"hall" : "..."
}
```

Output:

Returns an array of the REC ids that match the given hall.
This is given as an array of strings each in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"recId" : "..."
}
```

http://localhost:8080/ra/{id}/report-ra/{raId}

Input:

The “id” associated with the RA. The new “raId” to be reported. These are included in the link for the desired RA.

Output:

Reports the RA with the raId from the RA with the given id. Returns string with the message:

“RA Id: [raId] Reported”

http://localhost:8080/ra/{id}/report-rea/{reaId}

Input:

The “id” associated with the RA. The new “reaId” to be reported. These are included in the link for the desired RA.

Output:

Reports the REA with the reaId from the RA with the given id. Returns string with the message:

“REA Id: [reaId] Reported”

Resident Education Assistant Links

http://localhost:8080/rea/{id}/delete

Input:

The “id” associated with the REA. This is included in the link for the desired REA to be deleted.

Output:

Deletes the REA with the given id. Returns string with the message:

“Deleted User: [id]”

http://localhost:8080/rea/{id}/delete-ra/{raId}

See:
http://localhost:8080/ra/{id}/delete

^^^ raId = id ^^^

http://localhost:8080/rea/{id}/set-name

Input:

The “id” associated with the REA. This is included in the link for the desired RA.

The “name” to be set for the current REA. This is given as a string in JSON formatting, as seen below.

{
“name” : “...”
}

Output:

Changes the name of the REA with the given id. Returns string with the message:

“Name Edited”

http://localhost:8080/rea/{id}/set-email

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The “email” to be set for the current REA. This is given as a string in JSON formatting, as seen below.

```
{
“email” : “...”
}
```

Output:

Changes the email of the REA with the given id. Returns string with the message:

“Email Edited”

http://localhost:8080/rea/{id}/set-gender

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The “gender” to be set for the current REA. This is given as a string in JSON formatting, as seen below.

```
{
“gender” : “...”
}
```

Output:

Changes the gender of the REA with the given id. Returns string with the message:

“Gender Edited”

http://localhost:8080/rea/{id}/set-hall

Input:

The “id” associated with the RA. This is included in the link for the desired RA.

The “hall” to be set for the current RA. This is given as a string in JSON formatting, as seen below.

{
“hall” : “...”
}

Output:

Changes the hall of the RA with the given id. Returns string with the message:

“Hall Edited”

http://localhost:8080/rea/{id}/is-ra-clocked-in/{raId}

Input:

The “id” associated with the REA. The “raId” associated with the RA. These are included in the link.

Output:

Returns all Boolean indicated whether the given RA is clocked in. Returns string of the boolean, as seen below.

“[boolean]”
}

http://localhost:8080/rea/{id}/is-ra-enabled/{raId}

Input:

The “id” associated with the REA. The “raId” associated with the RA. These are included in the link.

Output:

Returns all Boolean indicated whether the given RA account is enabled. Returns string of the boolean, as seen below.

“[boolean]”
}

http://localhost:8080/rea/{id}/enable

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

Output:

Enables the account of the REA with the given id.

http://localhost:8080/rea/{id}/enable-ra/{raId}

See:
http://localhost:8080/ra/{id}/enable

^^^ raId = id ^^^

http://localhost:8080/rea/{id}/disable

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

Output:

Disables the account of the REA with the given id.

http://localhost:8080/rea/{id}/disable-ra/{raId}

See:
http://localhost:8080/ra/{id}/disable

^^^ raId = id ^^^

http://localhost:8080/rea/{id}/set-floor

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The “floor” to be set for the current REA. This is given as a string in JSON formatting, as seen below.

{
“floor” : “...”
}

Output:

Changes the floor of the REA with the given id. Returns string with the message:

“Floor Edited”

http://localhost:8080/rea/{id}/set-timezone

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The “timezone” to be set for the current REA. This is given as a string in JSON formatting, as seen below.

```
{
“timezone” : “...”
}
```

Output:

Changes the floor of the REA with the given id. Returns string with the message:

“Timezone Edited”

http://localhost:8080/rea/{id}/set-all

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

All basic attributes to be set for the current REA. This is given as a string in JSON formatting, as seen below.

```
{
“name“ : “...”,
“email“ : “...”,
“gender“ : “...”,
“hall“ : “...”,
“enabled“ : “...”,
“floor“ : “...”
}
```

Output:

Sets basic attributes REA with the given id. Returns string with the message:

“Basic Attributes Set”

http://localhost:8080/rea/{id}/create-account

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

Minimum number of attributes to be set for the current REA. This is given as a string in JSON formatting, as seen below.

{
“name“ : “...”,
“email“ : “...”,
“enabled“ : “...”
}

Output:

Sets minimum number of attributes to create an REA account with the given id. Returns string with the message:

“Basic Attributes Set”

http://localhost:8080/rea/{id}/create-welcome-message

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The welcome message to be set for the current REA. This is given as a string in JSON formatting, as seen below.

{
“message” : “...”
}

Output:

Sets welcome message for an REA account with the given id. Returns string with the message:

“Welcome Message Created”

http://localhost:8080/rea/{id}/get-welcome-message

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

Output:

Gets welcome message for an REA account with the given id. Returns string with the message:

“[message]”

http://localhost:8080/rea/{id}/delete-welcome-message

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

Output:

Deletes welcome message for an REA account with the given id. Returns string with the message:

“Welcome Message Deleted”

http://localhost:8080/rea/{id}/add-ra/{raId}

Input:

The “id” associated with the REA. The new “raId” to be added. These are included in the link for the desired RA.

Output:

Adds the RA with the raId to the REA with the given id. Returns string with the message:

“RA Added: [raId]”

http://localhost:8080/rea/{id}/remove-ra/{raId}

Input:

The “id” associated with the REA. The new “raId” to be removed. These are included in the link for the desired RA.

Output:

Removes the RA with the raId to the REA with the given id. Returns string with the message:

“RA Removed: [raId]”

http://localhost:8080/rea/{id}/get-ras

Input:

The “id” associated with the REA. These are included in the link for the desired RA.

Output:

Gets the RAs that belong to the REA with the given id. Returns string array with the raids formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"raId" : "..."
}
```

http://localhost:8080/rea/{id}/assign-shifts

Input:

The “id” associated with the REA. This is included in the link for the desired RA.

Output:

Assigns shifts created by the REA to their RAs in the order that they are stored. Returns string with the message:

“Shifts Assigned”

http://localhost:8080/rea/{id}/assign-shifts-randomly

Input:

The “id” associated with the REA. This is included in the link for the desired RA.

Output:

Assigns shifts created by the REA to their RAs in the order that they are randomly stored as. Returns string with the message:

“Shifts Assigned”

http://localhost:8080/rea/{id}/assign-shifts-automatically

Input:

The “id” associated with the REA. This is included in the link for the desired RA.

Output:

Assigns shifts created by the REA to their RAs based on the RA’s competition of that type of shift as well as the total number of shifts they have completed. Returns string with the message:

“Shifts Assigned”

http://localhost:8080/rea/{id}/shift-complete/{raId}

Input:

The “id” associated with the REA. The “raId” associated with the RA. These are included in the link for the desired RA.

The “dutyLevel” of the event completed by the RA. This is given as a string in JSON formatting, as seen below.

```
{
“dutyLevel” : “...”
}
```

Output:

Increments the counter of the RA for the number of “dutyLevel” shifts they have completed. Returns string with the message:

“Added Complete Shift to [raId]”

http://localhost:8080/rea/{id}/ra-primary/{raId}

Input:

The “id” associated with the REA. The RA associated with the “raId”. These are included in the link for the desired RA.

Output:

Returns the number of primary shifts completed by the RA. Returns string with the message:

“Number of Primary Shifts Completed: [#]”

http://localhost:8080/rea/{id}/ra-secondary/{raId}

Input:

The “id” associated with the REA. The RA associated with the “raId”. These are included in the link for the desired RA.

Output:

Returns the number of secondary shifts completed by the RA. Returns string with the message:

“Number of Secondary Shifts Completed: [#]”

http://localhost:8080/rea/{id}/ra-tertiary/{raId}

Input:

The “id” associated with the REA. The RA associated with the “raId”. These are included in the link for the desired RA.

Output:

Returns the number of tertiary shifts completed by the RA. Returns string with the message:

“Number of Tertiary Shifts Completed: [#]”

http://localhost:8080/rea/{id}/ra-stats/{raId}

Input:

The “id” associated with the REA. The RA associated with the “raId”. These are included in the link for the desired RA.

Output:

Returns the total number of shifts completed by the RA. Returns string with the message:

“Total Shifts Completed: [#]”

http://localhost:8080/rea/{id}/ra/{raId}/get-events

See:
http://localhost:8080/ra/{id}/get-events
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/get-shifts

See:
http://localhost:8080/ra/{id}/get-shifts
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/delete-event/{eventId}

See:
http://localhost:8080/ra/{id}/delete-event{eventId}
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/delete-event-day

See:
http://localhost:8080/ra/{id}/delete-event-day
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/delete-event-week

See:
http://localhost:8080/ra/{id}/delete-event-week
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/delete-event-month

See:
http://localhost:8080/ra/{id}/delete-event-month
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/edit-event/{eventId}/edit-title

See:
http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-title
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/edit-event/{eventId}/edit-description

See:
http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-description
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/edit-event/{eventId}/edit-duty-level

See:
http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-duty-level
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/add-event/{eventId}

See:
http://localhost:8080/ra/{id}/add-event/{eventId}
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/add-shift/{eventId}

See:
http://localhost:8080/ra/{id}/add-shift/{eventId}
^^^ raId = id ^^^

http://localhost:8080/rea/{id}/ra/{raId}/view-drop-requests

Input:

The “id” associated with the REA. The “raId” for the associated RA. These are included in the link for the desired RA.

Output:

Gets events that have been requested to be dropped by the RA with the given id. Returns string array with the raids formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "..."
}
```

http://localhost:8080/rea/{id}/ra/{raId}/deny-drop/{eventId}

Input:

The “id” associated with the REA. The “raId” for the associated RA. The “eventId” to be denied. These are included in the link for the desired RA.

Output:

Returns the shift drop denied by the RA. Returns string with the message:

“Drop Denied for Event: [eventId]”

http://localhost:8080/rea/{id}/ra/{raId}/delete-shift/{eventId}

Input:

The “id” associated with the REA. The “raId” for the associated RA. The “eventId” to be deleted. These are included in the link for the desired RA.

Output:

Deletes the event with “eventId” and returns all the shifts of the RA with the given id. Returns string array with the shifts formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/rea/{id}/create-shift/{eventId}

Input:

The “id” associated with the REA. The “eventId” to be created. This is included in the link for the desired REA.

The new shift to be added for the current REA. This is given as a string in JSON formatting, as seen below.

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

Output:

Outputs all the current shifts belonging to the REA with the given id. Returns string with the message:

For String[0], ..., String[n] =>

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
```

```
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

http://localhost:8080/rea/{id}/get-events

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

Output:

Returns all the events of the REA with the given id. Returns string array with the events formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/rea/{id}/get-shifts

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

Output:

Returns all the shifts of the REA with the given id. Returns string array with the shifts formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/rea/{id}/get-meetings

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

Output:

Returns all the meetings of the REA with the given id. Returns string array with the shifts formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/rea/{id}/get-availability/{eventId}

Input:

The “id” associated with the REA. The event associated with the “eventId”. These are included in the link for the desired RA.

Output:

Returns the number of spots left to fill the shift created by the REA. Returns string with the message:

“[#]”

http://localhost:8080/rea/{id}/create-event/{eventId}

Input:

The “id” associated with the REA. The “eventId” to be created. This is included in the link for the desired REA.

The new event to be added for the current REA. This is given as a string in JSON formatting, as seen below.

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

Output:

Outputs all the current events belonging to the REA with the given id. Returns string with the message:

For String[0], ..., String[n] =>

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
```

```
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

http://localhost:8080/rea/{id}/delete-event/{eventId}

Input:

The “id” associated with the REA as well as the “eventId” of the event to be deleted. These are included in the link for the desired REA.

Output:

Deletes event with the given eventId in the REA with the given id. Returns string with the message:

“Deleted: [eventId]”

http://localhost:8080/rea/{id}/delete-event-day

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The starting “day” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The starting “month” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The “timezone” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

```
{
"day" : "..."
"month" : "..."
"timezone" : "..."
}
```

Output:

Deletes events on starting day in the REA with the given id. Returns string with the message:

“Deleted Day”

http://localhost:8080/rea/{id}/delete-event-week

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The starting “day” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The starting “month” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The “year” of the starting day events to be deleted. This is given as a string in JSON formatting, as seen below.

The “timezone” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

```
{
"day" : "..."
"month" : "..."
"year" : "..."
"timezone" : "..."
}
```

Output:

Deletes all events up until a week after the starting day in the REA with the given id. Returns string with the message:

“Deleted Week”

http://localhost:8080/rea/{id}/delete-event-month

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The starting “month” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The “year” of the starting day events to be deleted. This is given as a string in JSON formatting, as seen below.

The “timezone” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

```
{
"month" : "..."
"year" : "..."
"timezone" : "..."
}
```

Output:

Deletes all events for the entire month day in the REA with the given id. Returns string with the message:

“Deleted Month”

http://localhost:8080/rea/{id}/find-ra-in-hall

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The new “hall” to be searched. This is given as a string in JSON formatting, as seen below.

```
{
"hall" : "..."
}
```

Output:

Returns an array of the RA ids that match the given hall.
This is given as an array of strings each in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"raId" : "..."
}
```

http://localhost:8080/rea/{id}/find-rea-in-hall

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The new “hall” to be searched. This is given as a string in JSON formatting, as seen below.

```
{
"hall" : "..."
}
```

Output:

Returns an array of the REA ids that match the given hall.
This is given as an array of strings each in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"reaId" : "..."
}
```

http://localhost:8080/rea/{id}/find-rec-in-hall

Input:

The “id” associated with the REA. This is included in the link for the desired REA.

The new “hall” to be searched. This is given as a string in JSON formatting, as seen below.

```
{
"hall" : "..."
}
```

Output:

Returns an array of the REC ids that match the given hall.
This is given as an array of strings each in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"recId" : "..."
}
```

http://localhost:8080/rea/{id}/report-ra/{raId}

Input:

The “id” associated with the REA. The new “raId” to be reported. These are included in the link for the desired REA.

Output:

Reports the RA with the raId from the REA with the given id. Returns string with the message:

“RA Id: [raId] Reported”

http://localhost:8080/rea/{id}/false-report-ra/{raId}

Input:

The “id” associated with the REA. The new “raId” to be reported. These are included in the link for the desired REA.

Output:

Nullifies the report for the RA with the raId from the REA with the given id. Returns string with the message:

“[raId] Report Nullified”

http://localhost:8080/rea/{id}/clear-reports-ra/{raId}

Input:

The “id” associated with the REA. The new “raId” to be reported. These are included in the link for the desired REA.

Output:

Clear all reports for the RA with the raId from the REA with the given id. Returns string with the message:

“[raId] Reports Reset”

http://localhost:8080/rea/{id}/report-rea/{reaId}

Input:

The “id” associated with the REA. The new “reaId” to be reported. These are included in the link for the desired REA.

Output:

Reports the REA with the reaId from the REA with the given id. Returns string with the message:

“REA Id: [reaId] Reported”

http://localhost:8080/rea/{id}/report-rec/{recId}

Input:

The “id” associated with the REA. The new “recId” to be reported. These are included in the link for the desired REA.

Output:

Reports the REC with the recId from the REA with the given id. Returns string with the message:

“REC Id: [recId] Reported”

Resident Education Coordinator Links

http://localhost:8080/rec/{id}/delete

Input:

The “id” associated with the REC. This is included in the link for the desired REC to be deleted.

Output:

Deletes the REC with the given id. Returns string with the message:

“Deleted User: [id]”

http://localhost:8080/rec/{id}/delete-rea/{reaId}

See:
http://localhost:8080/rea/{id}/delete

^^^ reaId = id ^^^

http://localhost:8080/rec/{id}/delete-ra/{raId}

See:
http://localhost:8080/ra/{id}/delete

^^^ raId = id ^^^

http://localhost:8080/rec/{id}/set-name

Input:

The “id” associated with the REC. This is included in the link for the desired RA.

The “name” to be set for the current REC. This is given as a string in JSON formatting, as seen below.

```
{
“name” : “...”
}
```

Output:

Changes the name of the REC with the given id. Returns string with the message:

“Name Edited”

http://localhost:8080/rec/{id}/set-email

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The “email” to be set for the current REC. This is given as a string in JSON formatting, as seen below.

```
{
“email” : “...”
}
```

Output:

Changes the email of the REC with the given id. Returns string with the message:

“Email Edited”

http://localhost:8080/rec/{id}/set-gender

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The “gender” to be set for the current REC. This is given as a string in JSON formatting, as seen below.

```
{
“gender” : “...”
}
```

Output:

Changes the gender of the REC with the given id. Returns string with the message:

“Gender Edited”

http://localhost:8080/rec/{id}/set-hall

Input:

The “id” associated with the REC. This is included in the link for the desired RA.

The “hall” to be set for the current RA. This is given as a string in JSON formatting, as seen below.

```
{
“hall” : “...”
}
```

Output:

Changes the hall of the RA with the given id. Returns string with the message:

“Hall Edited”

http://localhost:8080/rec/{id}/is-ra-clocked-in/{raId}

Input:

The “id” associated with the REC. The “raId” associated with the RA. These are included in the link.

Output:

Returns all Boolean indicated whether the given RA is clocked in. Returns string of the boolean, as seen below.

“[boolean]”
}

http://localhost:8080/rec/{id}/is-ra-enabled/{raId}

Input:

The “id” associated with the REC. The “raId” associated with the RA. These are included in the link.

Output:

Returns all Boolean indicated whether the given RA account is enabled. Returns string of the boolean, as seen below.

“[boolean]”
}

http://localhost:8080/rec/{id}/enable

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

Output:

Enables the account of the REC with the given id.

http://localhost:8080/rec/{id}/enable-ra/{raId}

See:
http://localhost:8080/ra/{id}/enable

^^^ raId = id ^^^

http://localhost:8080/rec/{id}/disable

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

Output:

Disables the account of the REC with the given id.

http://localhost:8080/rec/{id}/disable-ra/{raId}

See:
http://localhost:8080/ra/{id}/disable

^^^ raId = id ^^^

http://localhost:8080/rec/{id}/set-floor

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The “floor” to be set for the current REC. This is given as a string in JSON formatting, as seen below.

{
“floor” : “...”
}

Output:

Changes the floor of the REC with the given id. Returns string with the message:

“Floor Edited”

http://localhost:8080/rec/{id}/set-timezone

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The “timezone” to be set for the current REC. This is given as a string in JSON formatting, as seen below.

```
{
“timezone” : “...”
}
```

Output:

Changes the floor of the REC with the given id. Returns string with the message:

“Timezone Edited”

http://localhost:8080/rec/{id}/set-all

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

All basic attributes to be set for the current REC. This is given as a string in JSON formatting, as seen below.

```
{
“name“ : “...”,
“email“ : “...”,
“gender“ : “...”,
“hall“ : “...”,
“enabled“ : “...”,
“floor“ : “...”
}
```

Output:

Sets basic attributes REC with the given id. Returns string with the message:

“Basic Attributes Set”

http://localhost:8080/rec/{id}/create-account

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

Minimum number of attributes to be set for the current REC. This is given as a string in JSON formatting, as seen below.

```
{
“name“ : “...”,
“email“ : “...”,
“enabled“ : “...”
}
```

Output:

Sets minimum number of attributes to create an REC account with the given id. Returns string with the message:

“Basic Attributes Set”

http://localhost:8080/rec/{id}/create-welcome-message

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The welcome message to be set for the current REC. This is given as a string in JSON formatting, as seen below.

```
{
“message” : “...”
}
```

Output:

Sets welcome message for an REC account with the given id. Returns string with the message:

“Welcome Message Created”

http://localhost:8080/rec/{id}/get-welcome-message

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

Output:

Gets welcome message for an REC account with the given id. Returns string with the message:

“[message]”

http://localhost:8080/rec/{id}/delete-welcome-message

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

Output:

Deletes welcome message for an REC account with the given id. Returns string with the message:

“Welcome Message Deleted”

http://localhost:8080/rec/{id}/add-ra/{raId}

Input:

The “id” associated with the REC. The new “raId” to be added. These are included in the link for the desired RA.

Output:

Adds the RA with the raId to the REC with the given id. Returns string with the message:

“RA Added: [raId]”

http://localhost:8080/rec/{id}/add-rea/{reaId}

Input:

The “id” associated with the REC. The new “reaId” to be added. These are included in the link for the desired REC.

Output:

Adds the REA with the reaId to the REC with the given id. Returns string with the message:

“REA Added: [reaId]”

http://localhost:8080/rec/{id}/remove-ra/{raId}

Input:

The “id” associated with the REC. The new “raId” to be removed. These are included in the link for the desired RA.

Output:

Removes the RA with the raId to the REC with the given id. Returns string with the message:

“RA Removed: [raId]”

http://localhost:8080/rec/{id}/remove-rea/{reaId}

Input:

The “id” associated with the REC. The new “reaId” to be removed. These are included in the link for the desired REC.

Output:

Removes the REA with the reaId to the REC with the given id. Returns string with the message:

“REA Removed: [reaId]”

http://localhost:8080/rec/{id}/get-ras

Input:

The “id” associated with the REC. These are included in the link for the desired RA.

Output:

Gets the RAs that belong to the REC with the given id. Returns string array with the raids formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"raId" : "..."
}
```

http://localhost:8080/rec/{id}/get-reas

Input:

The “id” associated with the REC. These are included in the link for the desired REC.

Output:

Gets the REAs that belong to the REC with the given id. Returns string array with the raids formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"reaId" : "..."
}
```

http://localhost:8080/rec/{id}/assign-shifts

Input:

The “id” associated with the REC. This is included in the link for the desired RA.

Output:

Assigns shifts created by the REC to their RAs in the order that they are stored. Returns string with the message:

“Shifts Assigned”

http://localhost:8080/rec/{id}/assign-shifts-randomly

Input:

The “id” associated with the REC. This is included in the link for the desired RA.

Output:

Assigns shifts created by the REC to their RAs in the order that they are randomly stored as. Returns string with the message:

“Shifts Assigned”

http://localhost:8080/rec/{id}/assign-shifts-automatically

Input:

The “id” associated with the REC. This is included in the link for the desired RA.

Output:

Assigns shifts created by the REC to their RAs based on the RA’s competition of that type of shift as well as the total number of shifts they have completed. Returns string with the message:

“Shifts Assigned”

http://localhost:8080/rec/{id}/shift-complete/{raId}

Input:

The “id” associated with the REC. The “raId” associated with the RA. These are included in the link for the desired RA.

The “dutyLevel” of the event completed by the RA. This is given as a string in JSON formatting, as seen below.

```
{
“dutyLevel” : “...”
}
```

Output:

Increments the counter of the RA for the number of “dutyLevel” shifts they have completed. Returns string with the message:

“Added Complete Shift to [raId]”

http://localhost:8080/rec/{id}/ra-primary/{raId}

Input:

The “id” associated with the REC. The RA associated with the “raId”. These are included in the link for the desired RA.

Output:

Returns the number of primary shifts completed by the RA. Returns string with the message:

“Number of Primary Shifts Completed: [#]”

http://localhost:8080/rec/{id}/ra-secondary/{raId}

Input:

The “id” associated with the REC. The RA associated with the “raId”. These are included in the link for the desired RA.

Output:

Returns the number of secondary shifts completed by the RA. Returns string with the message:

“Number of Secondary Shifts Completed: [#]”

http://localhost:8080/rec/{id}/ra-tertiary/{raId}

Input:

The “id” associated with the REC. The RA associated with the “raId”. These are included in the link for the desired RA.

Output:

Returns the number of tertiary shifts completed by the RA. Returns string with the message:

“Number of Tertiary Shifts Completed: [#]”

http://localhost:8080/rec/{id}/ra-stats/{raId}

Input:

The “id” associated with the REC. The RA associated with the “raId”. These are included in the link for the desired RA.

Output:

Returns the total number of shifts completed by the RA. Returns string with the message:

“Total Shifts Completed: [#]”

http://localhost:8080/rec/{id}/ra/{raId}/get-events

See:
http://localhost:8080/ra/{id}/get-events
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/get-shifts

See:
http://localhost:8080/ra/{id}/get-shifts
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/delete-event/{eventId}

See:
http://localhost:8080/ra/{id}/delete-event{eventId}
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/delete-event-day

See:
http://localhost:8080/ra/{id}/delete-event-day
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/delete-event-week

See:
http://localhost:8080/ra/{id}/delete-event-week
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/delete-event-month

See:
http://localhost:8080/ra/{id}/delete-event-month
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/edit-event/{eventId}/edit-title

See:
http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-title
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/edit-event/{eventId}/edit-description

See:
http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-description
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/edit-event/{eventId}/edit-duty-level

See:
http://localhost:8080/ra/{id}/edit-event/{eventId}/edit-duty-level
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/add-event/{eventId}

See:
http://localhost:8080/ra/{id}/add-event/{eventId}
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/add-shift/{eventId}

See:
http://localhost:8080/ra/{id}/add-shift/{eventId}
^^^ raId = id ^^^

http://localhost:8080/rec/{id}/ra/{raId}/view-drop-requests

Input:

The “id” associated with the REC. The “raId” for the associated RA. These are included in the link for the desired RA.

Output:

Gets events that have been requested to be dropped by the RA with the given id. Returns string array with the raids formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "..."
}
```

http://localhost:8080/rec/{id}/ra/{raId}/deny-drop/{eventId}

Input:

The “id” associated with the REC. The “raId” for the associated RA. The “eventId” to be denied. These are included in the link for the desired RA.

Output:

Returns the shift drop denied by the RA. Returns string with the message:

“Drop Denied for Event: [eventId]”

http://localhost:8080/rec/{id}/ra/{raId}/delete-shift/{eventId}

Input:

The “id” associated with the REC. The “raId” for the associated RA. The “eventId” to be deleted. These are included in the link for the desired RA.

Output:

Deletes the event with “eventId” and returns all the shifts of the RA with the given id. Returns string array with the shifts formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/rec/{id}/create-shift/{eventId}

Input:

The “id” associated with the REC. The “eventId” to be created. This is included in the link for the desired REC.

The new shift to be added for the current REC. This is given as a string in JSON formatting, as seen below.

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

Output:

Outputs all the current shifts belonging to the REC with the given id. Returns string with the message:

For String[0], ..., String[n] =>

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
```

```
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

http://localhost:8080/rec/{id}/get-events

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

Output:

Returns all the events of the REC with the given id. Returns string array with the events formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/rec/{id}/get-shifts

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

Output:

Returns all the shifts of the REC with the given id. Returns string array with the shifts formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/rec/{id}/get-meetings

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

Output:

Returns all the meetings of the REC with the given id. Returns string array with the shifts formatted as string in JSON, as seen below.

For String[0], ..., String[n] =>

```
{
"eventId" : "...",
"title" : "...",
"description" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "..."
}
```

http://localhost:8080/rec/{id}/get-availability/{eventId}

Input:

The “id” associated with the REC. The event associated with the “eventId”. These are included in the link for the desired RA.

Output:

Returns the number of spots left to fill the shift created by the REC. Returns string with the message:

“[#]”

http://localhost:8080/rec/{id}/create-event/{eventId}

Input:

The “id” associated with the REC. The “eventId” to be created. This is included in the link for the desired REC.

The new event to be added for the current REC. This is given as a string in JSON formatting, as seen below.

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

Output:

Outputs all the current events belonging to the REC with the given id. Returns string with the message:

For String[0], ..., String[n] =>

```
{
"title" : "...",
"startHour" : "...",
"startMinute" : "...",
"startMonth" : "...",
"startDay" : "...",
"startYear" : "...",
"startTimezone" : "...",
"endHour" : "...",
```

```
"endMinute" : "...",
"endMonth" : "...",
"endDay" : "...",
"endYear" : "...",
"endTimezone" : "...",
"dutyLevel" : "...",
"availability" : "..."
}
```

http://localhost:8080/rec/{id}/delete-event/{eventId}

Input:

The “id” associated with the REC as well as the “eventId” of the event to be deleted. These are included in the link for the desired REC.

Output:

Deletes event with the given eventId in the REC with the given id. Returns string with the message:

“Deleted: [eventId]”

http://localhost:8080/rec/{id}/delete-event-day

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The starting “day” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The starting “month” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The “timezone” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

```
{
"day" : "..."
"month" : "..."
"timezone" : "..."
}
```

Output:

Deletes events on starting day in the REC with the given id. Returns string with the message:

“Deleted Day”

http://localhost:8080/rec/{id}/delete-event-week

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The starting “day” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The starting “month” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The “year” of the starting day events to be deleted. This is given as a string in JSON formatting, as seen below.

The “timezone” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

```
{
"day" : "..."
"month" : "..."
"year" : "..."
"timezone" : "..."
}
```

Output:

Deletes all events up until a week after the starting day in the REC with the given id. Returns string with the message:

“Deleted Week”

http://localhost:8080/rec/{id}/delete-event-month

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The starting “month” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

The “year” of the starting day events to be deleted. This is given as a string in JSON formatting, as seen below.

The “timezone” of the event to be deleted. This is given as a string in JSON formatting, as seen below.

```
{
"month" : "..."
"year" : "..."
"timezone" : "..."
}
```

Output:

Deletes all events for the entire month day in the REC with the given id. Returns string with the message:

“Deleted Month”

http://localhost:8080/rec/{id}/find-ra-in-hall

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The new “hall” to be searched. This is given as a string in JSON formatting, as seen below.

```
{
"hall" : "..."
}
```

Output:

Returns an array of the RA ids that match the given hall.
This is given as an array of strings each in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"raId" : "..."
}
```

http://localhost:8080/rec/{id}/find-rea-in-hall

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The new “hall” to be searched. This is given as a string in JSON formatting, as seen below.

```
{
"hall" : "..."
}
```

Output:

Returns an array of the REA ids that match the given hall.
This is given as an array of strings each in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"reaId" : "..."
}
```

http://localhost:8080/rec/{id}/find-rec-in-hall

Input:

The “id” associated with the REC. This is included in the link for the desired REC.

The new “hall” to be searched. This is given as a string in JSON formatting, as seen below.

```
{
"hall" : "..."
}
```

Output:

Returns an array of the REC ids that match the given hall.
This is given as an array of strings each in JSON formatting, as seen below.

For String[0], ..., String[n] =>

```
{
"recId" : "..."
}
```

http://localhost:8080/rec/{id}/report-ra/{raId}

Input:

The “id” associated with the REC. The new “raId” to be reported. These are included in the link for the desired REC.

Output:

Reports the RA with the raId from the REC with the given id. Returns string with the message:

“RA Id: [raId] Reported”

http://localhost:8080/rec/{id}/false-report-ra/{raId}

Input:

The “id” associated with the REC. The new “raId” to be reported. These are included in the link for the desired REC.

Output:

Nullifies the report for the RA with the raId from the REC with the given id. Returns string with the message:

“[raId] Report Nullified”

http://localhost:8080/rec/{id}/clear-reports-ra/{raId}

Input:

The “id” associated with the REC. The new “raId” to be reported. These are included in the link for the desired REC.

Output:

Clear all reports for the RA with the raId from the REC with the given id. Returns string with the message:

“[raId] Reports Reset”

http://localhost:8080/rec/{id}/report-rea/{reaId}

Input:

The “id” associated with the REC. The new “reaId” to be reported. These are included in the link for the desired REC.

Output:

Reports the REA with the reaId from the REC with the given id. Returns string with the message:

“REA Id: [reaId] Reported”

http://localhost:8080/rec/{id}/false-report-rea/{reaId}

Input:

The “id” associated with the REC. The new “reaId” to be reported. These are included in the link for the desired REC.

Output:

Nullifies the report for the REA with the reaId from the REC with the given id. Returns string with the message:

“[reaId] Report Nullified”

http://localhost:8080/rec/{id}/clear-reports-rea/{reaId}

Input:

The “id” associated with the REC. The new “reaId” to be reported. These are included in the link for the desired REC.

Output:

Clear all reports for the REA with the reaId from the REC with the given id. Returns string with the message:

“[reaId] Reports Reset”

http://localhost:8080/rec/{id}/report-rec/{recId}

Input:

The “id” associated with the REC. The new “recId” to be reported. These are included in the link for the desired REC.

Output:

Reports the REC with the recId from the REC with the given id. Returns string with the message:

“REC Id: [recId] Reported”

http://localhost:8080/rec/{id}/false-report-rec/{recId}

Input:

The “id” associated with the REC. The new “recId” to be reported. These are included in the link for the desired REC.

Output:

Nullifies the report for the REC with the recId from the REC with the given id. Returns string with the message:

“[recId] Report Nullified”

http://localhost:8080/rec/{id}/clear-reports-rec/{recId}

Input:

The “id” associated with the REC. The new “recId” to be reported. These are included in the link for the desired REC.

Output:

Clear all reports for the REC with the recId from the REC with the given id. Returns string with the message:

“[recId] Reports Reset”