

Class: Main

+ main(args: String[]): void

Class: Event

- name: String
- startDateTime: LocalDateTime
- endDateTime: LocalDateTime
- duration: Duration
- memo: Memo
- id: int
- numEvents: int
- tags: ArrayList<String>
- alerts: ArrayList<Alerts>

+ Event(name: String, start: LocalDateTime, end: LocalDateTime)
+ Event(name: String, start: LocalDateTime, end: LocalDateTime, memo: Memo)
+ addTag(tag: String): void
+ addMemo(newMemo: Memo): void
+ addAlert(newAlert: Alert): void
+ bringDownNum(): void
+ getTags(): ArrayList<String>
+ getAlerts(): ArrayList<Alert>
+ getId(): int
+ toString(): String
+ getName(): String
+ getStartDateTime(): LocalDateTime
+ getEndDateTime(): LocalDateTime
+ getDuration(): Duration
+ removeTag(content: String): void
+ getMemo(): Memo
+ editName(newName: String): void
+ editMemo(memo: Memo): void
+ editStart(start: LocalDateTime): void
+ editEnd(end: LocalDateTime): void
+ removeMemo(): void
+ removeAlert(alert: Alert): void
+ postpone(): void

Class: Alert

- name: String
- dateTime: LocalDateTime

- id: int
- numAlerts: int
+ userAlerted: boolean

+ Alert(name: String, when: LocalDateTime)
+ getId(): int
+ bringDownNum(): void
+ getLocalDate(): LocalDate
+ getLocalTime(): LocalTime
+ getLocalDateTime(): LocalDateTime
+ changeName(content: String): void
+ changeTime(time: LocalDateTime): void
+ getName(): String

Class: Memo

- contents: String
- events: ArrayList<Integer>
- id: int
- numMemos: int

+ Memo(contents: String)
+ getId(): int
+ getEvents(): ArrayList<Event>
+ toString(): String
+ bringDownNum(): void
+ addEvent(id: Integer): void
+ removeEvent(id: Integer): void
+ changeName(content: String): void

Class: Series

- name: String
- events: ArrayList<Integer>
- id: int
- numSeries: int

+ Series(name: String)
+ getId(): int
+ getName(): String
+ getEvents(): ArrayList<Integer>
+ bringDownNum(): void
+ addEvent(newEvent: Event): void

Class: AlertSeries

- name: String
- alerts: ArrayList<Alert>
- id: int
- numAlertSeries: int
- duration: Duration

+ AlertSeries(name: String, duration: Duration)
+ getId(): int
+ getName(): String
+ getAlerts(): ArrayList<Integer>
+ bringDownNum(): void
+ removeAlert(id: int): void
+ editDuration(duration: Duration): void
+ editName(name: String): void

Class: EventManager

- store: ArrayList<Event>

+ EventManager(store: ArrayList<Event>)
+ postponeIndef(id: int): void
+ duplicateEvent(id: int): Event
+ createEvent(name: String, start: LocalDateTime, end: LocalDateTime): Event
+ deleteEvent(event: Event): void
+ editEventStart(event: Event, newStart: LocalDateTime): void
+ editEventEnd(event: Event, newEnd: LocalDateTime): void
+ editName(event: Event, newName: String): void
+ addTag(event: Event, content: String): void
+ removeTag(event: Event, content: String): void
+ editTag(event: Event, oldTag: String, newTag: String): void
+ deleteAlert(event: Event, alert: Alert): void
+ addAlert(event: Event, alert: Alert): void
+ deleteMemo(event: Event): void
+ addMemo(event: Event, memo: Memo): void
+ getEvent(id: int): Event
+ getEvent(event: Event): int
+ getMemo(event: Event): Memo
+ getEventsByDate(date: LocalDateTime): ArrayList<Event>
+ getPastEvents(): ArrayList<Event>
+ getFutureEvents(): ArrayList<Event>
+ getCurrentEvents(): ArrayList<Event>
+ getEventsByMemo(memo: Memo): ArrayList<Event>
+ getEventsByTag(tag: String): ArrayList<Event>
- timetoDate(time: LocalDateTime): LocalDate

Class: AlertManager

- serialAlerts: ArrayList<ArrayList<Alert>> - store: ArrayList<Alert>
--

+ AlertManager(store: ArrayList<Alert>, serialAlerts: ArrayList<ArrayList<Alert>>) + addAlert(name: String, when: LocalDateTime): Alert + addSerialAlert(name: String, start: LocalDateTime, finish: LocalDateTime, repetition: Duration): ArrayList<Alert> + editName(alert: Alert, content: String): void + removeAlert(alert: Alert): void + editAlertTime(alert: Alert, when: LocalDateTime): void + removeSerialAlert(alert: Alert): ArrayList<Alert> + seriesofAlerts(alert: Alert): ArrayList<Alert> + editSeriesOfAlerts(alert: Alert, name: String): void + remainingAlert(int: id): ArrayList<Alert>

Class: MemoManager

- store: ArrayList<Memo>

+ MemoManager(store: ArrayList<Memo>) + addMemo(id: int, content: String): Memo + removeMemo(content: String, id: int): void - contains(content: String): boolean - getContent(content: String): Memo + removeEventMemo(memo: Memo, id: int): void + deleteMemo(content: String): void + editName(memo: Memo, content: String): void + memoEvent(memo: Memo): ArrayList<Integer>
--

Class: SeriesManager

- store: ArrayList<Series> - eventManager: EventManager
--

+ SeriesManager(eventManager: EventManager, store: ArrayList<Series>) + createSerialEvent(startStart: LocalDateTime, startEnd: LocalDateTime, repetition: Duration, absoluteEnd: LocalDateTime, name: String): Series + deleteSerialEvent(series: Series): void + editName(series: Series, name: String): void + seriesGetter(event: Event): Series

Class: OverallManager

- memoManager: MemoManager - alertManager: AlertManager - seriesManager: SeriesManager - eventManager: EventManager

Class: UserManager
- jsonArray: JSONArray - <u>FILE_PATH</u> : String
+ UserManager() + login(username: String, password: String): int + createNewUser(username: String

Class: CalendarManager
- userMg: UserManager - overMg: OverallManager - dataMg: CalendarDataFacade - userCalendarNum: int
+ getNumCalendars() + login() + sendEventToOtherUser() + login_send_event() + createEvent() + createNewUser() + deleteUserID() + getEvents() + getEventByID() + deleteEvents() + getEventsByDate() + getAlerts() + logout()

Class: CalendarDataFacade
- events: ArrayList<Event> - memos: ArrayList<Memo> - alerts: ArrayList<Alert> - series: ArrayList<Series> - alertSeries: ArrayList<AlertSeries> - loader: CalendarDataLoader

<ul style="list-style-type: none">- saver: CalendarDataSaver- jfile: JSONObject- usern: String
--

<ul style="list-style-type: none">+ CalendarDataFacade()+ getEvents(): ArrayList<Event>+ getMemos(): ArrayList<Memo>+ getAlerts(): ArrayList<Alert>+ getSeries(): ArrayList<Series>+ getAlertSeries(): ArrayList<AlertSeries>+ timetoDate(time: LocalDateTime): LocalDate+ login(username: String): void+ addNewUser(username: String): void+ logout(): void+ addEventToUser(event: Event, username: String): void
--

Class: CalendarDataLoader

<ul style="list-style-type: none">- <u>makeLocalDateTime(time: JSONObject): LocalDateTime</u>+ <u>loadData(toBeLoaded: ArrayList<JSONArray>, variables: ArrayList<ArrayList>): void</u>- <u>loadMemos(jsonmemos: JSONArray, memos: ArrayList<Memo>): void</u>+ <u>loadEvents(jsonevents: JSONArray, events: ArrayList<Event>, memos: ArrayList<Memo>, alerts: ArrayList<Alert>): void</u>- <u>loadSeries(jsonseries: JSONArray, series: ArrayList<Series>, events: ArrayList<Event>): void</u>- <u>loadAlerts(jsonalerts: JSONArray, alerts: ArrayList<Alerts>): void</u>- <u>loadAlertSeries(jsonalertSeries: JSONArray, alertSeries: ArrayList<AlertSeries>): void</u>
--

Class: CalendarDataSaver

<ul style="list-style-type: none">+ saveData(events: ArrayList<Event>, memos: ArrayList<Memo>, alerts: ArrayList<Alert>, series: ArrayList<Series>, alertSeries: ArrayList<AlertSeries>, obj: JSONObject, username: String): void- timeToJSON(time: LocalDateTime): JSONObject+ saveEvents(events: ArrayList<Event>): JSONArray- saveMemos(memos: ArrayList<Memo>): JSONArray- saveAlertSeries(alertSe: ArrayList<AlertSeries>): JSONArray- saveAlerts(alerts: ArrayList<Array>): JSONArray- saveSeries(series: ArrayList<Series>): JSONArray

Class: UserInterface

- sc: Scanner
- calendarManager: CalendarManager
- currUser: String

+ start(): void
+ viewEvents(): void
+ viewEventsByDay(): void
+ deleteEvents(): void
+ viewEvent(): void
+ createEvents(): void
+ viewAlerts(): void
+ login(): void
+ createAccount(): void
+ deleteAccount(): void