

Data structure of adjust brightness in schedule

task_dto.jar contains data structure of adjust brightness in schedule, definition as following:

```
package com.xixun.contract.model;
```

```
//task of adjust brightness in schedule
```

```
public class TaskToSetBrightness {
```

```
    public String createDate; //date of creating the task
```

```
    public String createBy; //creator
```

```
    public String name; //task name
```

```
    public List<TaskItemToSetBrightness> items; //task items
```

```
    public int defaultBrightness; //default brightness, for non-scheduled time
```

```
}
```

```
//scheudled items
```

```
public class TaskItemToSetBrightness {
```

```
    public int brightness; //brightness value for scheduled time
```

```
    public List<Schedule> schedules; //scheduled time
```

```
}
```

```
public class Schedule {
```

```
    public DateType dateType; //if this item value equals to DateType.Range,
```

means scheduled time range from startDate to endDate. If this item value equals to DateType.All, means the date range : permanent

public String **startDate**; *//for example: "2015-10-27", this value will be valid when dateType equals to DateType.Range*

public String **endDate**; *//example : "2015-11-18", this value will be valid when dateType equals to DateType.Range*

public TimeType **timeType**; *//if this item value equals to TimeType.Range, means this scheduled item' s time range :everyday tartTime to endTime. If this item value equals to TimeType.All, means time range : whole day*

public String **startTime**; *//example : "07:30", this value will be valid when timeType equals to TimeType.Range*

public String **endTime**; *//example: "21:40", this value will be valid when timeType equals to TimeType.Range*

public FilterType **filterType**; *//if this item value is FilterType.None, means not filter scheduled items. If this item value is FilterType.Week, means filter according to week. If this item value is FilterType.Month, means filter according to month.*

public List<Integer> **weekFilter**; *// sunday==0 example : [1,6,0] means filter according to Monday, Saturday and Sunday., this value will be valid when filterType equals to FilterType.Week*

public List<Integer> **monthFilter**; *// example: [1,13,28] means filter according to 1th, 13th, 28th date per month, this value will be valid when filterType equals to FilterType.Month*

```
}
```

```
public enum DateType{
```

```
    Range,
```

```
    All
```

```
}
```

```
public enum TimeType {
```

```
    Range,
```

```
    All
```

```
}
```

```
public enum FilterType {
```

```
    None,
```

```
    Week,
```

```
    Month
```

```
}
```