

# Digital Away Day

---

Deloitte Away Day is a maven project to generate a day program to arrange a list of activities.

## Author

Shubham Agarwal (shubham.agarwal7076@gmail.com)

## Tech

Program uses the following technologies:

- [Java] - Code base language (version 1.8)
- [Spring] - For Dependency Injection (version 4.0)
- [Apache Maven] - Build automation tool (version 3.3.9)
- [junit] - Library for testing (version 4.12)
- [codehaus mojo] - Execute Java app plugin (version 1.6.0)

## Installation

Deloitte digital Away Day generates a day program based in an input file stored in the directory `src\main\resources\input.txt`.

These are the allowed formats for tasks:

Format	example
[task_name] [time_in_minutes]min	Docker master class 60min
[task_name] sprint	Docker master class sprint

Sprint is used for 15 minutes tasks. Find bellow an input example:

Duck Herding 60min

Archery 45min

Learning Magic Tricks 45min

Laser Clay Shooting 60min

Human Table Football 30min

Buggy Driving 30min  
Salsa & Pickles sprint  
2-wheeled Segways 45min  
Viking Axe Throwing 60min  
Giant Puzzle Dinosaurs 30min  
Giant Digital Graffiti 60min  
Cricket 2020 60min  
Wine Tasting sprint  
Arduino Bonanza 30min  
Digital Treasure Hunt 60min  
Enigma Challenge 45min  
Monti Carlo or Bust 60min  
New Zealand Haka 30min  
Time Tracker sprint  
Indiano Drizzle 45min

## Execution

Deloitte Away Day requires [Maven](#) v3+ and [Java] v1.8+ to run.

Please set JAVA\_HOME and M2\_HOME in environment variable then  
Open a terminal (CMD) and go to project location before executing below  
commands to compile and execute:

```
$ mvn clean install  
$ mvn exec:java
```

Output will be shown in the terminal. Find below an output example:

```
Feb 13, 2019 2:33:26 PM  
org.springframework.context.support.AbstractApplicationContext  
prepareRefresh  
INFO: Refreshing  
org.springframework.context.annotation.AnnotationConfigApplicationContex  
t@4783da3f: startup date [Wed Feb 13 14:33:26 IST 2019]; root of context  
hierarchy  
Deloitte Digital Away Day:  
Team 1:  
09:00 am : Duck Herding 60min  
10:00 am : Archery 45min  
10:45 am : Learning Magic Tricks 45min
```

11:30 am : Human Table Football 30min  
13:00 pm : Viking Axe Throwing 60min  
14:00 pm : Giant Digital Graffiti 60min  
15:00 pm : Cricket 2020 60min  
16:00 pm : Staff Motivation Presentation 60min

Team 2:

09:00 am : Laser Clay Shooting 60min  
10:00 am : Buggy Driving 30min  
10:30 am : Salsa & Pickles sprint  
10:45 am : 2-wheeled Segways 45min  
11:30 am : Giant Puzzle Dinosaurs 30min  
13:00 pm : Wine Tasting sprint  
13:15 pm : Arduino Bonanza 30min  
13:45 pm : Digital Treasure Hunt 60min  
14:45 pm : Enigma Challenge 45min  
15:30 pm : New Zealand Haka 30min  
16:00 pm : Staff Motivation Presentation 60min

Feb 13, 2019 2:33:26 PM

org.springframework.context.support.AbstractApplicationContext doClose

INFO: Closing

org.springframework.context.annotation.AnnotationConfigApplicationContext  
t@4783da3f: startup date [Wed Feb 13 14:33:26 IST 2019]; root of context  
hierarchy

## Design

These are the main classes used in digital awayday project:

### ***com.digital.awayday.service***

*DayProgramService: Class to store a morning ActivityBlock and an evening ExtraTimeBlock. Morning and evening start and finish date can be configured. Methods to include tasks into both blocks are included.*

*AwayDayService: Class to store a list of DayProgram. Morning and evening start and finish time, extra time for evening and number of teams can be configured. Also a method to add a list of tasks is included.*

---

***com.digital.awayday.service***

---

*ActivityBlockService : Class to handle tasks to their respective slot if there is enough free size*

*TaskDistributionService : Class to collect and loop through all tasks and try to fit into slots. Also it will calculate how many teams can be formed in given tasks*

---

***com.digital.awayday.dao***

---

*AwayDayDao: Class to load task from resource.*

---

***com.digital.awayday.model***

---

*Task: Class to store a task data (name, duration). Start time is calculated using a LocalTime.*

*ExtraTimeBlock: ActivityBlock extended class including an extratime. Available size method overwriten to manage extra time.*

---

***com.digital.awayday.file***

---

*FileUtil: Class with a method for reading a list of tasks from a file (specific format).*

---

***com.digital.awayday.exception***

---

*AwayDayException: Returned exception when a functional or execution error occurs*

---