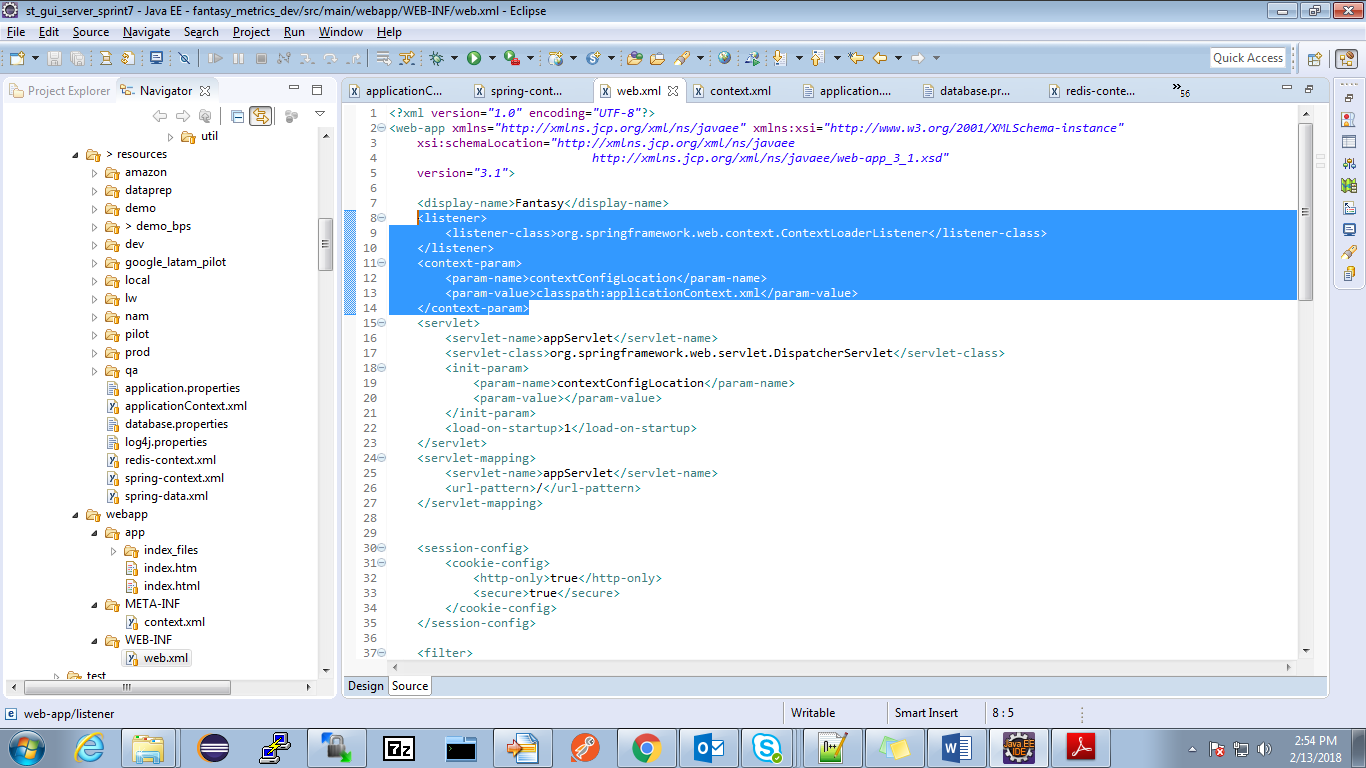
Grunt: The JavaScript Task Runner -

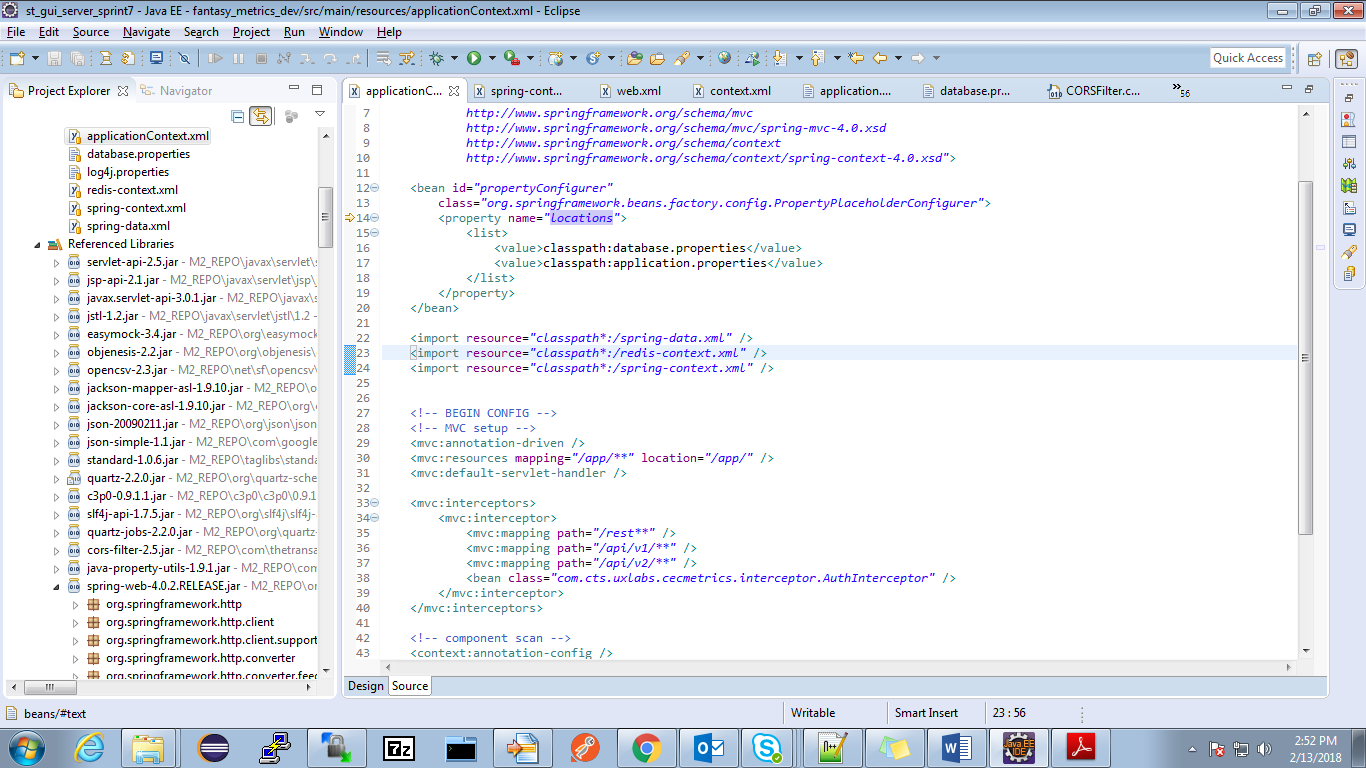
Grunt is a JavaScript task runner, a tool used to automatically perform frequent tasks such as minification, compilation, unit testing, and linting. It uses a command-line interface to run custom tasks defined in a file.

In web.xml – To load applicationContext.xml



In applicationContext.xml –

Beans for property files -



Also loaded spring-data.xml, redis-context.xml and spring-context.xml

@Target({ElementType.***FIELD***, ElementType.***METHOD***, ElementType.***PARAMETER***, ElementType.***ANNOTATION\_TYPE***})

@Retention(RetentionPolicy.***RUNTIME***)

@Documented

**public** **@interface** Value {

Usage of factory -

1. private static final Logger LOGGER = LoggerFactory.getLogger(Top5Bottom5KPIStandingsNotificationJob.class);

2. In LeagueController.java -

@Autowired

**private** LeagueTypeServiceFactory leagueTypeServiceFactory;

leagueTypeServiceFactory.getLeagueTypeService(leagueType); //Call to get a leagueTypeService object

**private** Map<String, LeagueTypeService> leagueTypeServiceMap;

/\*\*

\* **@param** leagueType

\* **@return**

\*/

**public** LeagueTypeService getLeagueTypeService(**final** Integer leagueType) {

String name = LeagueType.*getName*(leagueType);

name = name + "Service";

***LOG***.info(name);

LeagueTypeService leagueTypeService = leagueTypeServiceMap.get(name);

***LOG***.info("LeagueTypeService :- " + leagueTypeService.getClass());

**return** leagueTypeService;//Gives actual object

}

//LeagueType enum

**public** **enum** LeagueType {

***POINTS***("PointsLeague", 1), ***NEVER\_ENDING\_POINTS***("NeverEndingPointsLeague", 3);

**private** String name;

**private** Integer type;

**private** LeagueType(**final** String name, **final** Integer type) {

**this**.type = type;

**this**.name = name;

}

**public** **static** String getName(**final** Integer type) {

**if** (type == 1) {

**return** ***POINTS***.name;

} **else** {

**return** ***NEVER\_ENDING\_POINTS***.name;

}

}

**public** Integer getType() {

**return** **this**.type;

}

@Override

**public** String toString() {

**return** **this**.name;

}

}