

## FREE SLOT AUTOMATION

B Prashanthi, Assistant Professor

D. Vamsi Krishna

Mahatma Gandhi Institute of Technology.

### Abstract

Greater number of the colleges generally maintain soft copies of the lecturer's time tables either in the form of excel documents or any centralized database maintained by the college. When stored in these formats, the data is pretty much unorganized and it's hard to analyze this data and extract information from it. Most of the colleges face a hardship in identifying the free times of their staff in the event of conduct of examinations or scheduling meetings. In the current context, this work is being manually done which is not productive and manual work is always error prone. In this paper, we are presenting an application that automates this task by taking the time tables or the schedules of the staff and gives the output that consists of information regarding the free slots of the staff in the requested time.

### Introduction

Identification of free slots of their respective staff is considered as a base work for many activities conducted by colleges and organizations. The greater the number of staff, the tedious it becomes to identify the free slots of each and every staff and analyze it. Considering the current scenario of most of the colleges, they get the work done manually by taking the time tables of each staff individually and identify their free times. Once this information is obtained, the same can be used as a source for other tasks such as automatic invigilation duty assignment, scheduling meetings when most of the staff are free, etc. In majority of the cases the time tables of the staff are either saved in form of excel spreadsheets, which is totally raw data that generally has no structure and is unorganized or could be stored in relational databases in form of

tables which are organized and easy to analyze and maintain. In either of the cases, though the data is present, extracting the insights from it manually is still tedious and there is a need and scope for automating this task. The proposed application is an automation, which takes the input of each of the staff and builds a model to identify the free slots of every staff. Also the timings could also be provided so that the exact list of staff available in the required time slot can be identified. Once this is done, the data can be exported to spreadsheets and then be used as input data for other automations such as automatic exam invigilation assignment.

### Scope

The application provides a complete automation that can be used by any organization or college to reduce their overhead in manually extracting information from the data they have. This application only requires the inputs of the schedules of staff members and is capable of organizing the data by itself and extracting the required information automatically. There is no requirement of high end hardware support or network connections to run this software.

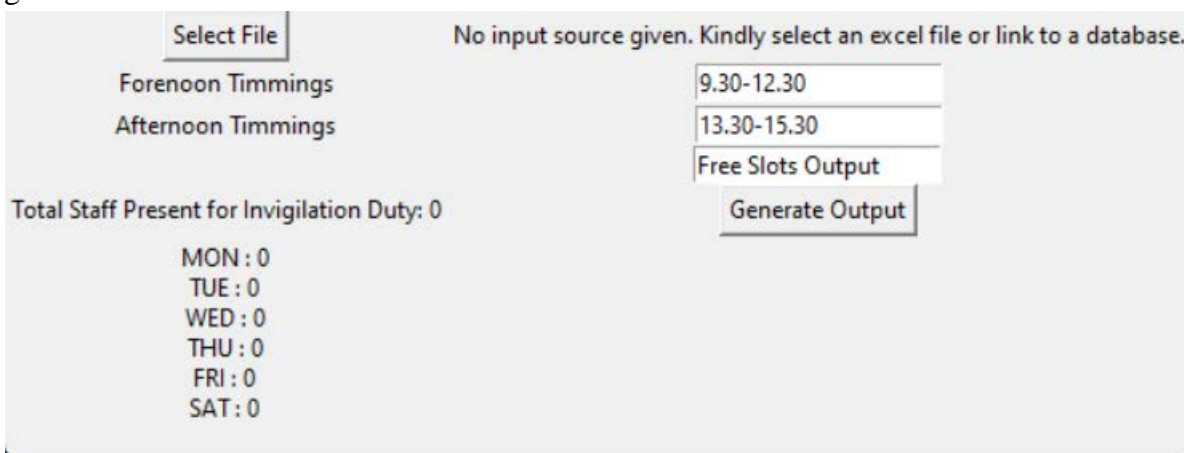
### Problem Statement

To conduct an event in any college or organization, identifying the free timings of their staff in an important task either to assign them their respective duties by considering their current schedule or to manage their work schedules by analyzing their work load. Consider a situation where a college has to assign invigilators for exam duties for a week. For this the management has to gather data of all the available staff and their schedules. Then they have to identify whether each staff is free for a particular exam and then do the same for rest of the staff which is practically a

tedious and time consuming job. As the staff count, number of exams increases the time and work to be done grows exponentially. But since the information extracted from it has its own importance, most of them can't afford to skip it. Even if all the work is done, there still are many chances that the work done is faulty. Since schedules keep changing all the time, when again the same information is needed, all the work has to be done again. Considering all the above aspects, automating all these tasks could reduce the time and effort in doing the same work.

**Proposed Solution**

Considering the requirements and inputs to be given to automate the free slot identification



**Input Format**

The input is the format of the schedule to be provided for the automation. The format of the input could be structured or unstructured depending on the availability of data. Structured data is the database table format that consists of schedule of each staff in each table. The column headings are timings and row labels are the week days. The unstructured data can be an excel spreadsheet with no specified format, but it is recommended that the provided format is to be followed to have faster results. There are some constraints that should be adhered such as there should be no merged cells, no blank columns or rows between the data, empty cells represent a free schedule and the time range should be separated with a '-' in the column headings.

of staff, an application has been designed such that it is capable to take the data source, the timings at which the free slots have to be found, and the output file details. The data source can be of the type excel, that consists of schedules of all the staff in each sheet or directly point it to a database that consists schedules of each of the staff in separate tables. Following this, the user has an option to give 2 timings in which the free slots have to be found out. Finally when the required data is given, the user can finally use the generate output to get the desired free slot information in the required format.

**Output Specification**

On providing the input data the user can generate the required data, which by default will be in a excel spreadsheet. The spreadsheet consists of columns named with name of the week day followed forenoon or afternoon for each day of the week. The output is shown as per the timings given as an input by the user in the application that is the free slot in forenoon and afternoon as per the given timings.

DAY	09.15-10.15	10.15-11.15	11.15-12.15	12.15-12.45	12.45-13.45	13.45-14.45	14.45-15.45	15.45-16.45
MON			JCSE2 CD	Lunch	Major Projects	Major Projects	Major Projects	Major Projects
TUE				Lunch		JCSE2	JCSE2	
WED	Major Projects	Major Projects	Major Projects	Lunch				
THU	JCSE2 CD	JCSE2 CD		Lunch				
FRI	Mini Project with Seminar 1 M.Tech Sem-2	Mini Project with Seminar 1 M.Tech Sem-3	Mini Project with Seminar 1 M.Tech Sem-4	Lunch	Mini Project with Seminar 1 M.Tech Sem-2			
SAT				Lunch	Major Projects	Major Projects	Major Projects	

All the slots for which the staff is available is marked as Available and the rest are marked as NA (Not Available).

Name	Monday		Tuesday		Wednesday		Thursday		Friday		Saturday	
	Monday Forenoon	Monday Afternoon	Tuesday Forenoon	Tuesday Afternoon	Wednesday Forenoon	Wednesday Afternoon	Thursday Forenoon	Thursday Afternoon	Friday Forenoon	Friday Afternoon	Saturday Forenoon	Saturday Afternoon
Staff 1	NA	NA	NA	NA	NA	Available	NA	Available	NA	NA	NA	NA
Staff 2	NA	NA	NA	Available	NA	NA	NA	Available	NA	NA	NA	NA
Staff 3	NA	Available	NA	NA	NA	Available	NA	Available	NA	NA	NA	NA
Staff 4	NA	NA	NA	NA	NA	Available	NA	Available	NA	Available	NA	NA
Staff 5	NA	NA	NA	NA	NA	Available	NA	Available	NA	NA	NA	NA
Staff 6	NA	NA	NA	NA	NA	NA	NA	Available	NA	Available	NA	NA

Select File      D://MGIT Files//4-1//Slot Automation//Time Tables.xlsx file is selected.

Forenoon Timmings      9.30-12.30

Afternoon Timmings      13.30-15.30

Free Slots Output

Generate Output

Total Staff Present for Invigilation Duty: 6

MON : 1  
TUE : 1  
WED : 4  
THU : 6  
FRI : 2  
SAT : 0

### Conclusion

The proposed solution reduces manpower, workload and the automation greatly reduces time taken to generate data form given schedules. This automation targets educational institutions, and benefits them a lot by reducing the complexity involved in generating data insights from the schedules. The proposed methodology uses quite efficient computing that is highly scalable. This can automate the manual work involved in getting the free slots from pre-defined schedules.