ARTIFICIAL INTELLIGENCE

INT - 404

PROJECT:- INTELLIGENT TIME-TABLE PREPARATION

Submitted By:-

NAME	REGISTRATION	ROLL NO.
	NO.	
P Smithin Reddy	11807366	A17/K18GT
Srijan Singh	11807017	A19/K18GT
Maddala Narmada	11807192	A20/K18GT

Submitted to:- Amandeep Kaur Mam



Format Of The Project Report

- 1. Format Of Report
- 2. Abstract
- 3. Introduction
- 4. Objectives
- 5. Methodology
- 6. Results
- 7. Conclusion
- 8. Future Scope
- 9. GitHub Link

1. Abstract:

This project defines and says that the total fundamental basics we used in the implementation of this application. And we approach successfully and creates a time-table according to needs we provide. The overview and outlook of this project makes the student to develop a creative planning task in the further. By modifying this

application we can also able to management and create the different types of time-tables according to department. And here we can highlight the process to approach of the project.

2. Introduction:

For any university or college or institution to run successfully the main theme should be a good time-table management and make a reliable.

Here time-table management evolves an approach in which involves the subjects list and duration of each hour. The university can provide their related subjects in the list and hour after that it going to give a time-table. And the best example we can discuss is the LPU here all streams are well- planned and time-table is provided in every field respective like classes and examinations, scrutiny. With the help of these management we can minimize the work to the humans.

3. Objectives:

The objective we can achieve with this project is the output comes from this one a timetable in which there is an subjectslist and duration hour. With the help of providing a max hour to each subject we can distinguish each subjects and provided with specific hour. And also it is very helpful. By mentioning the student id and faculty id it can give us a time-table.

4. Methodology:

In the time-table preparation we have written a programme in such a way that that the user will elated and can able to have the time-table in a perfect manner.

Here first we have to provide the total days in a week and subjects in the list in which we want and duration of hour to each class is provided.

And here we are using the csv file and pathlib in the code in which we used to get the project.

And here we will show the Excel file which represents in an effective way and we show in that days in excel file and duration of each hour of the subject.

And we are using an inbuilt package of sys in python of the command line argument.

The main thing of any project is allocation that means a perfect allocation of each subject to that related hour and well planned task.

Coding in python helps us a well mannered readable structure. And AI is suitable for implementing the any project code.

And here the main part which is def to define the definition and here we are giving the maximum hour to the subject.

And for saving the time-table we can create and save into csv file.

And by defining by the variables like subjects list and start hour and next hour we can start the code.

5. Results:

```
Implementation of code:
import pathlib
import csv
# Let's start by defines variables
subjects list = []
start hour = 8 # school start at 8.am
next_hour = 9 # 1rst next hour is 9.am
school days = [
  'monday',
  'tuesday'
time_slot_list = [] # get list of time slot
subject_per_slot = {}
MAX HOUR PER SUBJECT = 6 # use capital letter because it's a
constant variable
subject_hour_count = {}
def fill in subjects list():
  """Ask user subjects and fill in subjects list"""
```

```
enter another subject = True
  while enter another subject:
    subject = input('Type another subject: ')
    subject = subject.capitalize()
    if not subject in subjects list:
      subjects list.append(subject)
      subject hour count[subject] = MAX HOUR PER SUBJECT
    else:
      print(f'You\'ve already type {subject} in list.')
    question = input('Enter another subject (type "n" to exit)?')
    if question.lower() == 'n':
      enter_another_subject = False
def fill_out_subjects_list():
  """Ask user subjects and fill in subjects list"""
  subjects = input('Type all subjects you want add in subjects list\
```

```
and separate them by comma: ') # we collect all subjects
  the_subjects = subjects.replace(', ', ',') # remove space after comma
  # Split all subjects in order to put them into a list
  the subjects = the subjects.split(',')
  for subject in the subjects:
    subject = subject.capitalize()
    if not subject in subjects list:
      subjects list.append(subject)
      subject hour count[subject] = MAX HOUR PER SUBJECT
def ask_hour():
  """Ask hour to user"""
  print(f'Subjects list: {subjects_list}')
  print(f'Planning time: {start hour}h-{next hour}h')
```

user_answer = input('What\'s subject do you want put here? ')

```
def fill in timetable():
  """Display an hour & ask user which subject he want to put there"""
  global start hour
  global next_hour
  for day in school days:
    # Reset start and next hour
    the hour = {}
    time = 0
    start hour = 8 # we suppose that school start at 8.am
    next hour = 9
    print('\n----')
    print(f'{day.capitalize()} timetable')
    print('----\n')
    while time < 4: # Suppose we've 4hours course/day (you can
```

return user_answer

change it)

```
hour format = f'{start hour}h-{next hour}h' # format time slot
# it's represent 8 hours/per day for school
if time == 2: # if it's a midday (12.am), make a break
  # Add a break in timetable with 'Break time' as inscription
  subject per slot[hour format] = ['Break time']
  # Add hour format while making sure we avoid duplicate
  if not hour format in time slot list:
    time slot list.append('hour format')
else:
  chosen subject = ask hour().capitalize()
  print(f'start hour: {start hour}')
  print(f'next hour: {next hour}')
  # Check that subject chosen by user is in subjects list
  while not chosen_subject in subjects_list:
    print(f'{chosen subject} is not in subjects list.')
    print('Choose another subject.')
    chosen subject = ask hour().capitalize()
  # Add hour format while making sure we avoid duplicate
```

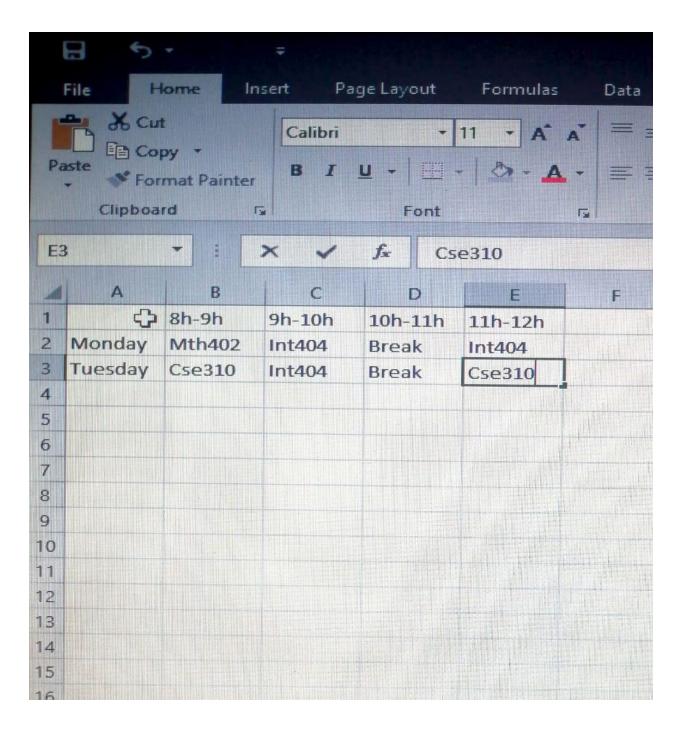
```
if not hour format in time slot list:
           time slot list.append(hour format)
           subject per slot[hour format] = [chosen subject]
        else:
           subject per slot[hour format] += [chosen subject]
        # Check that chosen subject max hours didn't reached
        for subject, max hour in subject hour count.items():
           if chosen subject == subject:
             # remove one hour on subject max hour
             subject hour count[chosen subject] = max hour - 1
      # go to next hour
      start hour += 1
     next hour += 1
      time += 1
fill out subjects list()
fill_in_timetable()
print(f'Subject per slot: {subject per slot}')
timetable_path = pathlib.Path.cwd() / 'timetable.csv'
# Now, let's write process to save timetable into a csv file
```

```
with open(timetable path, 'w') as timetable file:
  timetable writing = csv.writer(timetable file)
  # Write headers into csv file
  csv_headers = ['Hours']
  csv headers.extend(school days)
  timetable_writing.writerow(csv_headers)
  # Write content into csv file
  for time slot, concerned subjects in subject per slot.items():
    time line = [time slot]
    concerned subjects list = []
    if concerned subjects == ['Break time']:
      for x in range(0, len(school days)):
        concerned subjects list.append('Break time')
    else:
      concerned_subjects_list = concerned_subjects
    final line = time line + concerned subjects list
    timetable writing.writerow(final line)
  print('Your timetable is ready')
```

```
print('Your timetable is ready')
 Type all subjects you want add in subjects listand separate them by comma: INT404, CSE310, MTH402
Monday timetable
Subjects list: ['Int404', 'Cse310', 'Mth402', '']
Planning time: 8h-9h
What's subject do you want put here? MTH402
start_hour: 8
next hour: 9
Subjects list: ['Int404', 'Cse310', 'Mth402', '']
Planning time: 9h-10h
What's subject do you want put here? INT404
start_hour: 9
next_hour: 10
Subjects list: ['Int404', 'Cse310', 'Mth402', '']
Planning time: 11h-12h
What's subject do you want put here? INT404
start hour: 11
```

```
Subjects list: ['Int404', 'Cse310', 'Mth402', '']
Planning time: 8h-9h
What's subject do you want put here? CSE310
start hour: 8
next hour: 9
Subjects list: ['Int404', 'Cse310', 'Mth402', '']
Planning time: 9h-10h
What's subject do you want put here? INT404
start_hour: 9
next hour: 10
Subjects list: ['Int404', 'Cse310', 'Mth402', '']
Planning time: 11h-12h
What's subject do you want put here? CSE310
start_hour: 11
next hour: 12
Subject per slot: {'8h-9h': ['Mth402', 'Cse310'], '9h-10h': ['Int404', 'Int404'], '10h-11h': ['Break
time'], '11h-12h': ['Int404', 'Cse310']}
Your timetable is ready
```

The picture which was shown above is the output in which after compiling it was going to ask the user type the name of the subjects and duration of each other after providing the details it will show the time-table.



And here is the csv file in which the data will be saved and shown in an effective way.

6. Future Scope:

As every project had done will be atleast have some effect on future. With help of this project one can able to prepare and manage their own time-table. As preparing the time-table manually is not an easy task when there is strength in thousands but, with the help of these application it is possible and reduce strain and minimize human work.

7. GitHub Link:

- https://github.com/smithinr/intelligent-timetable/blob/master/intelligent%20time%20table
- https://www.github.com/srijansingh/ai-assignment
- https://github.com/MaddalaNarmada/Al project1