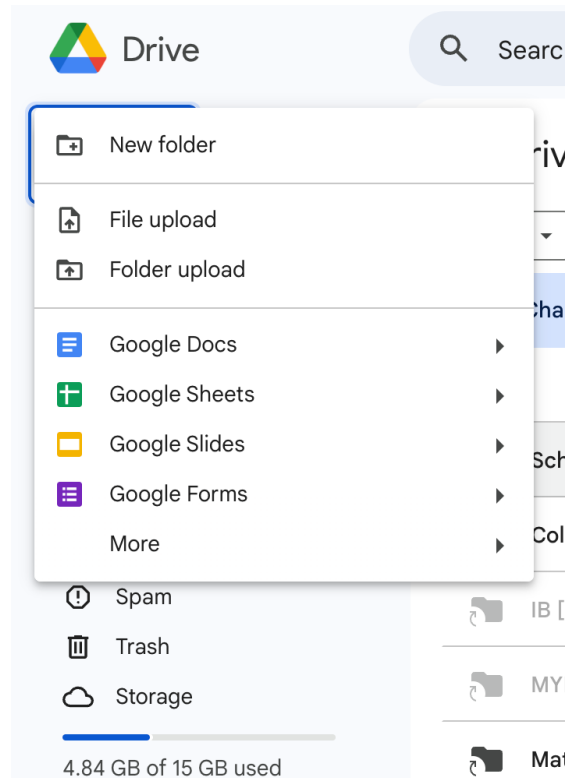


Manual

1. Create folder in GGdrive by clicking “+ New ⇒ New Folder”, e.g., Scheduling_Project











2. Put .ipynb file and .xlsx file in the same folder, e.g., put Resident Scheduling Project .ipynb and new_input_2024_adj.xlsx in Scheduling_Project. Open the .ipynb file.

My Drive > Scheduling_Project ▾



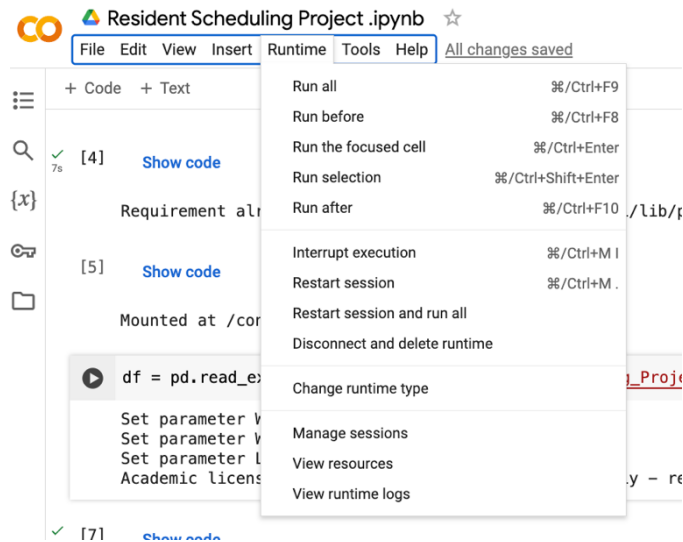
Type ▾ People ▾ Modified ▾

Name	Owner	Last modified b...	File size	
 Resident Scheduling Project .ipynb	 me	5:18 PM	65 KB	⋮
 new_input_2024_adj.xlsx	 me	11:38 AM	24 KB	    ⋮

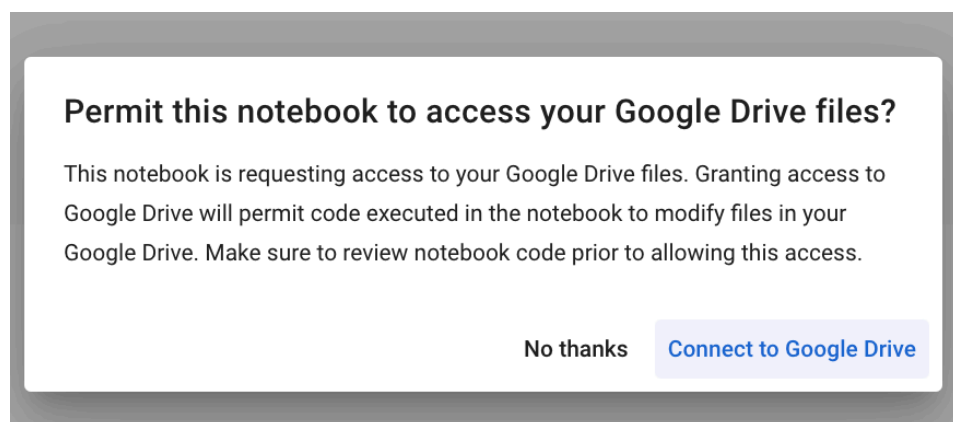
- Before running the file, you need to set directory of .xlsx file by adjusting this below command line in .ipynb file. For example, if GGdrive folder is Scheduling_Project and excel file name is new_input_2024_adj.xlsx. You have to set directory like this:

```
df =  
pd.read_excel("/content/drive/MyDrive/Scheduling_Project/new_input_2024_adj.xlsx", sheet_name = None)
```

- From menu bar, click “Runtime ⇒ Run all”



- After running, the below window will show for the first time of run. Click “Connect to GG Drive” and sign into your GG Drive



6. It will take sometimes to run and if it can find the result, it will show “optimal solution found” in notebook as well as popping the result file in download as shown below:

Explored 1 nodes (81885 simplex iterations) in 243.62 seconds (110.35 work units)
Thread count was 2 (of 2 available processors)

Solution count 1: 49

Optimal solution found (tolerance 1.00e-04)

Best objective 4.900000000000e+01, best bound 4.900000000000e+01, gap 0.0000%

colab.research.google.com/drive/1qN5sTIXsL0vqbyD1F2BKujf8CRzU_q6#scrollTo=1Qy5xSP1X7Hx

Resident Scheduling Project.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Code Text

```
23421 -5.324114e+20 9.040000e+22 0.000000e+00 355
32837 4.899798e+01 8.000000e+00 0.000000e+00 405
34137 4.899984e+01 4.169811e+01 0.000000e+00 455
35237 4.883073e+01 8.391667e+01 0.000000e+00 515
36307 4.874763e+01 1.646267e+02 0.000000e+00 555
37437 4.868443e+01 4.279900e+01 0.000000e+00 605
38097 4.866667e+01 0.000000e+00 0.000000e+00 655
38097 4.866667e+01 0.000000e+00 0.000000e+00 655
```

Root relaxation: objective 4.866667e+01, 38097 iterations, 39.91 seconds (19.01 work units)
Total elapsed time = 65.56s (DegenMoves)
Total elapsed time = 79.00s (DegenMoves)
Total elapsed time = 86.11s (DegenMoves)
Total elapsed time = 99.24s (DegenMoves)
Total elapsed time = 106.67s (DegenMoves)

Nodes		Current Node		Objective Bounds		Work	
Expl	Unexpl	Obj	Depth IntInf	Incumbent	BestBd	Gap	It/Node Time
0	0	48.66667	0 2873	-	48.66667	-	129s
0	0	49.00000	0 418	-	49.00000	-	239s
H	0	0	0	49.0000000	49.00000	0.00%	243s
0	0	49.00000	0 418	49.00000	49.00000	0.00%	243s

Cutting planes:
Gomory: 1
MIR: 8
Zero half: 245
RLT: 124
Relax-and-lift: 1

Explored 1 nodes (81885 simplex iterations) in 243.62 seconds (110.35 work units)
Thread count was 2 (of 2 available processors)

Solution count 1: 49

Optimal solution found (tolerance 1.00e-04)
Best objective 4.900000000000e+01, best bound 4.900000000000e+01, gap 0.0000%

Recent Download History

- schedule_result.xlsx 32.4 KB • 27 minutes ago
- Result_Vacation_Plan (20).xlsx 8.3 KB • 1 hour ago
- Result_Schedule (26).xlsx 9.7 KB • 1 hour ago
- Vacation_Table (26).xlsx 25.7 KB • 1 hour ago
- resident_scheduling_project_py 45.7 KB • 2 hours ago
- Result_Vacation_Plan (19).xlsx 8.3 KB • 4 hours ago
- Result_Schedule (25).xlsx 9.7 KB • 4 hours ago
- Vacation_Table (25).xlsx 25.7 KB • 4 hours ago
- Result_Vacation_Plan (18).xlsx

Full Download History

7. On the other hand, if the input cause infeasibility, it will show below. Anyway, if the algorithm run for too long (over 30 mins), it is safe to say that the input is infeasible

Root relaxation: infeasible, 828258 iterations, 3637.55 seconds (2032.75 work units)

Nodes		Current Node		Objective Bounds		Work	
Expl	Unexpl	Obj	Depth IntInf	Incumbent	BestBd	Gap	It/Node Time
0	0	infeasible	0	-	infeasible	-	3668s

Explored 1 nodes (828258 simplex iterations) in 3668.79 seconds (2042.66 work units)
Thread count was 2 (of 2 available processors)

Solution count 0

Model is infeasible

Best objective -, best bound -, gap -

Model is infeasible

8. To try fixing infeasibility, you have to adjust input file located in the folder. After adjusting, please remember to make sure what you update is saved (ctrl + s). Then, close input file , reopen again, go back to notebook. Then, click “Runtime ⇒ Restart session and run all ”

Examples :

1. This input cause infeasibility. Say I want to adjust value in the red block

new_input_2024_adj .XLSX

File Edit View Insert Format Data Tools Help

Q Menus 100% 123 Default... 10 B I

	A	B	C	D	E	F	G	H	I	J	K
1		1	1 or 2	2	2 or 3	3	3 or 4	4	4 or 5	5	
2	Trauma	3	0	0	2(-1)	0	0				
3	EGS	2	0	2	0		0	1	0	1	
4	SICU	2	0	0	2(-1)	0	0				
5	CRS	2	0	1(-1)	0				0	1	
6	MIS	2	0	0	0	1(-1)	0		0	1	
7	SO	1	0				0	1	0	1	
8	Brendo	1	0	1(-1)	0	0	1(-1)	0	0		
9	Peds	2	0		0	0	2	0	0		
10	Thor	1	0								
11	Txp	1	0		0	0	0				
12	Holy				0	1(-1)	0				
13	NFA	1(-1)	0								
14	NFB	1(-1)	0								
15	NFC	1(-1)	0	0	1(-1)	0	0				
16	NFS						0	1	0		
17	Lancaster						0	1	0		
18	Elective	0	0	0	0	0	0	0	0		
19	CGOH		0	0	0						
20	Vasc	0	0	0	0						
21	Wound	0	0	0	0						
22	-	0	0	0	0	0	0				

2. I adjust to 2(-2). Then, I save using ctrl +s \Rightarrow close input file \Rightarrow reopen input file

[illegible]

3. Go back to notebook and click “Runtime ⇒ Restart session and run all ”

The screenshot shows a Jupyter Notebook titled "Resident Scheduling Project.ipynb". The "Runtime" menu is open, displaying various options. The option "Restart session and run all" is highlighted. The notebook contains two code cells. Cell [1] has a title "# @title" and code "%pip install gurobipy". Cell [2] has a title "# @title" and code for importing libraries (gurobipy, pandas, numpy, ipywidgets, IPython.display, time, math, collections, google.colab) and setting environment variables (WLSACCESSID, WLSSECRET, LICENSEID) before starting the environment.

```
[1] # @title
%pip install gurobipy
Requirement already satisfied: gurobipy in /lib/python3.10/dist-packages (11.0.1)

[2] # @title
import gurobipy
from gurobipy import *
import pandas as pd
import numpy as np
import ipywidgets as widgets
from IPython.display import clear_output
import time, math
import collections
from google.colab import files
#Importing Excel data file
from google.colab import files
#print('Select Input Data File')
#uploaded = files.upload()
from os import sched_getparam

drive.mount('/content/drive')
# @title
env = gp.Env(empty=True)
env.setParam('WLSACCESSID', '4d7cd1b7-3dfe-4e87-80c7-2ada56221071')
env.setParam('WLSSECRET', 'b307457c-470d-4e18-b9c6-30f10d32ed35')
env.setParam('LICENSEID', 2378987)
env.start()
```