

# TIME TABLE PROJECT

Assigning resource rooms & scheduling courses



---

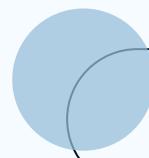
## Team 1

HyeHyun Lee Jinkyeong Kyon Jiyun Seo  
Manjong Kim Seoyeon Kim Yoonsun Jeong

# Table of Contents

<b>1. Roles / Progress</b>	역할 / 일정
<b>2. Identify Platform</b>	개발 도구 설정
<b>3. Data Analysis / migration</b>	데이터 분석 및 전처리
<b>4. Database / Table Creation</b>	데이터베이스 구축
<b>5. AI Algorithm Development</b>	AI 알고리즘 개발
<b>6. Server construction</b>	서버 구축
<b>7. Screen / Dashboard Display</b>	프론트 엔드

# 1. Roles

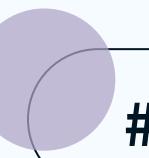


# Team 1-1

## AI Algorithm Development

Jinkyeong Kyon  
Seoyeon Kim

- AI Engine Development



# Team 1-2

## Backend Development

Manjong Kim  
Jiyun Seo  
Yoonsun Jeong

- Database/Table Creation
- Server building
- API connect



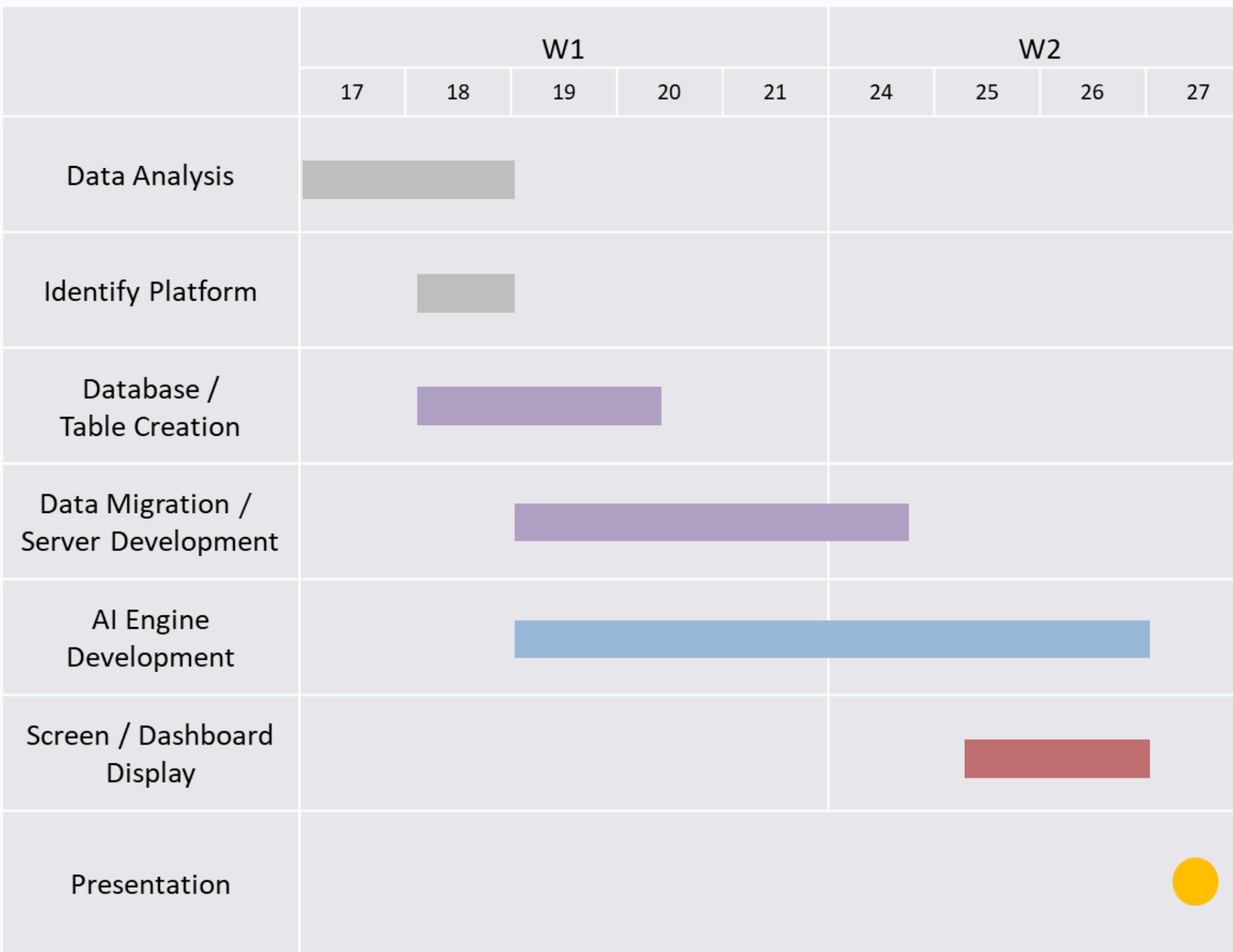
# Team 1-3

## Frontend Development

HyeHyun Lee

- Screen/Dashboard Display

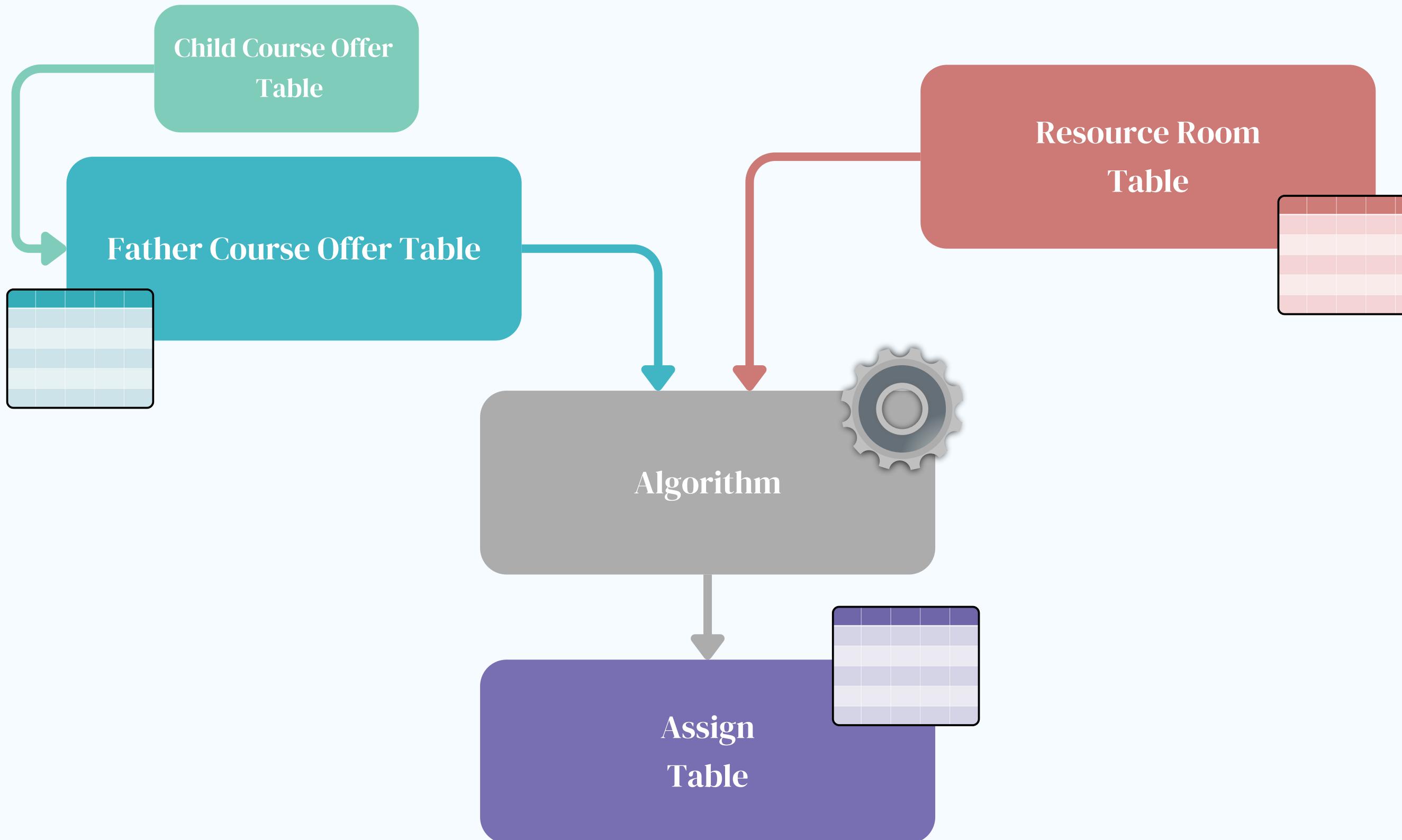
# 1. Progress



- # Team 1-1
- # Team 1-2
- # Team 1-3

# 1. Progress : Table Creation

---



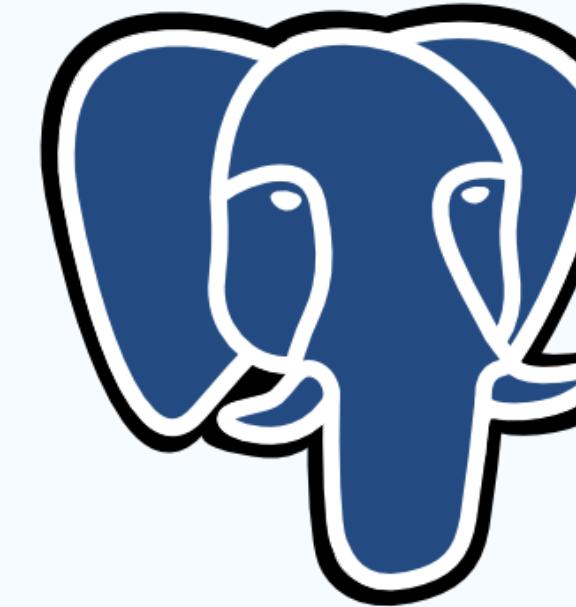
## 2. Identify Platform

Server.



**django**

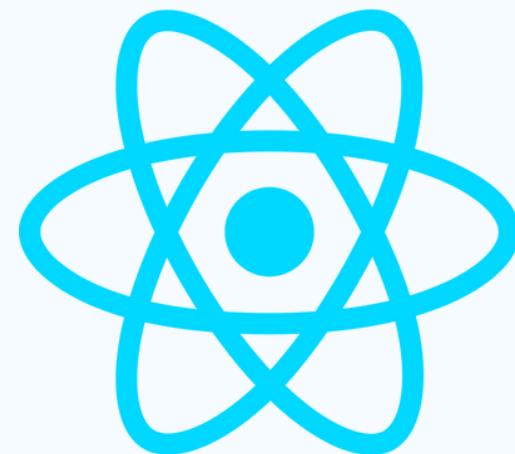
- Python-based
- Web Framework
- Suitable for Data Processing



**PostgreSQL**

- Relational Database
- Highly Compatible with Django
- Familiar with the Tool from Previous Experience

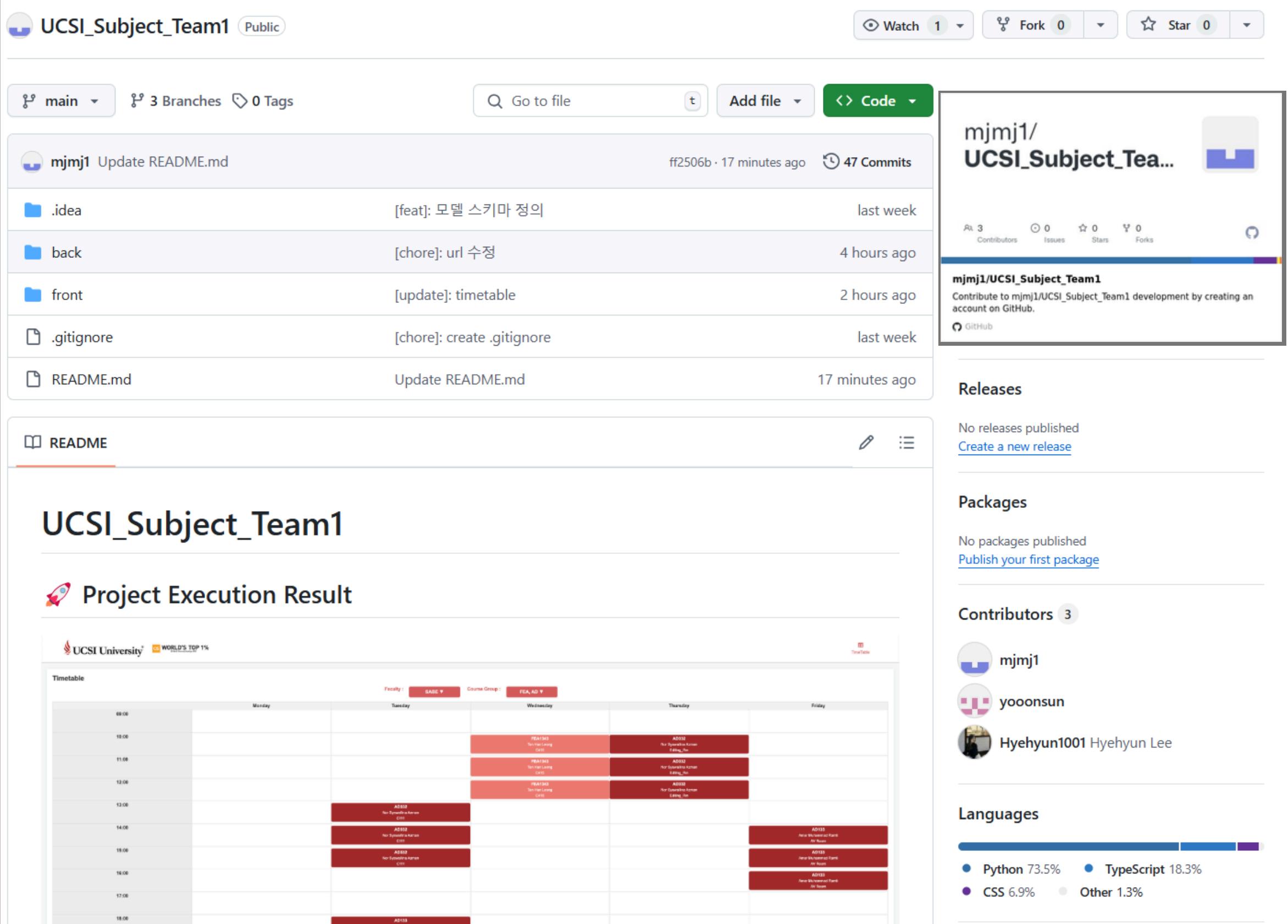
Front.



**React / Vite**

- JavaScript-based UI Library / Frontend Build Tool
- Familiar with the Tool from Previous Experience
- Optimized for modern web development with high performance

## 2. Identify Platform : Github

UCSI\_Subject\_Team1 (Public)

Watch 1 Fork 0 Star 0

main 3 Branches 0 Tags Go to file Add file Code

mjmj1 Update README.md ff2506b · 17 minutes ago 47 Commits  
[feat]: 모델 스키마 정의 last week  
.idea [chore]: url 설정 4 hours ago  
back [update]: timetable 2 hours ago  
front [chore]: create .gitignore last week  
.gitignore README.md Update README.md 17 minutes ago

README

## UCSI\_Subject\_Team1

### Project Execution Result

UCSI University WORLD'S TOP 1% Timetable

Monday	Tuesday	Wednesday	Thursday	Friday
9:00				
10:00				
11:00				
12:00				
13:00				
14:00				
15:00				
16:00				
17:00				
18:00				

mjmj1/UCSI\_Subject\_Tea...  
3 Contributors 0 Issues 0 Stars 0 Forks  
mjmj1/UCSI\_Subject\_Team1 Contribute to mjmj1/UCSI\_Subject\_Team1 development by creating an account on GitHub.

Releases No releases published Create a new release

Packages No packages published Publish your first package

Contributors 3  
mjmj1  
yooonsun  
Hyehyun1001 Hyehyun Lee

Languages Python 73.5% TypeScript 18.3%  
CSS 6.9% Other 1.3%

[https://github.com/mjmj1/UCSI\\_Subject\\_Team1](https://github.com/mjmj1/UCSI_Subject_Team1)

### 3. Data Analysis

---

## Father

- Single
- Father class

## Child

- Class belonging to Father  
(Combined to ...)

## Exception

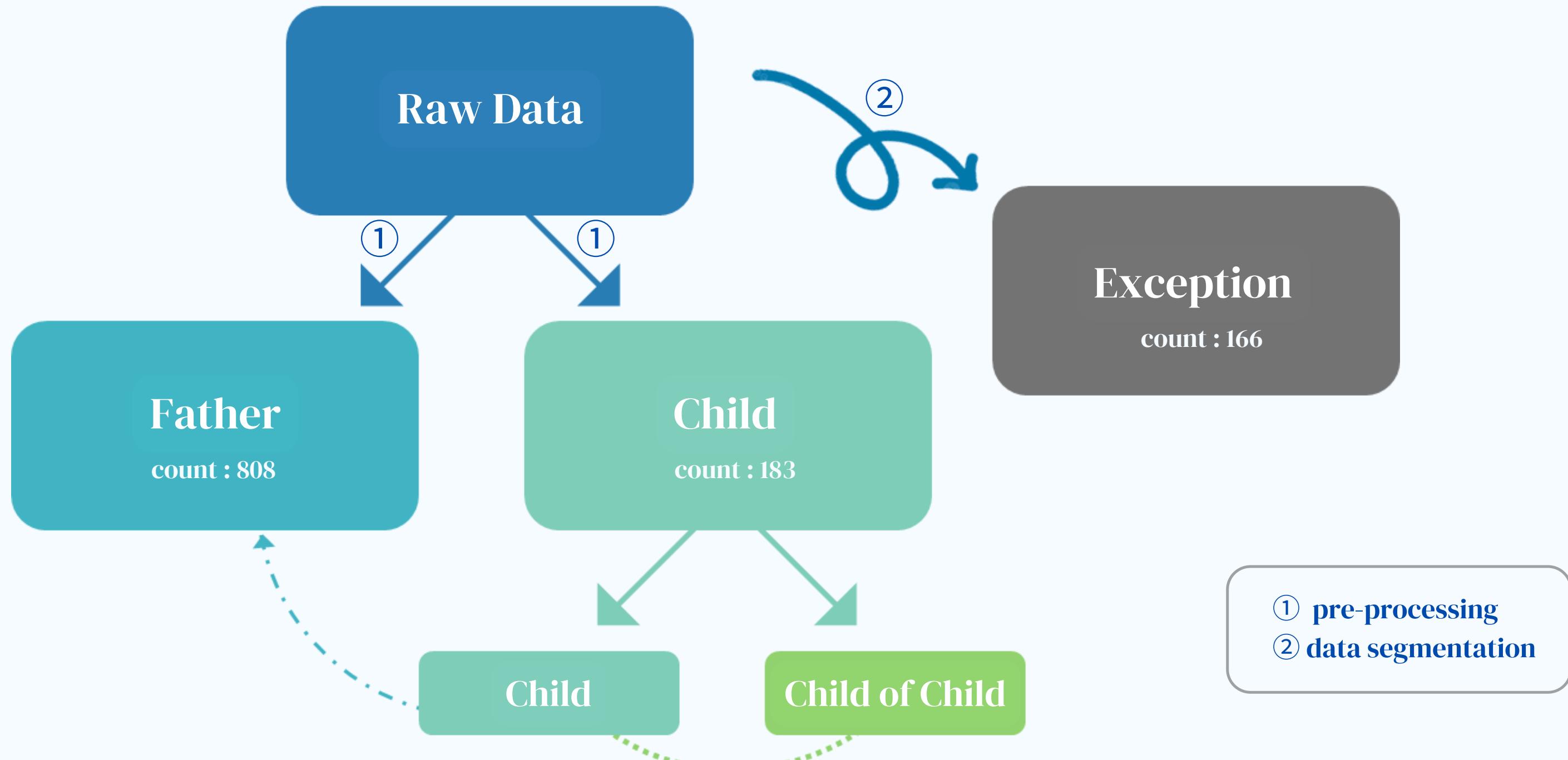
- Reference to itself
- Error Data  
(only combined to /)  
(Child data without Father)

### Why?

---

- Since [combined to (course\_code)] was categorized as a child, we separated it into a father/child table.
- Data that does not meet this criteria or cannot be utilized is classified as an exception table.

### 3. Data migration : CourseOffer



### 3. Data migration : CourseOffer

```
def check_father_code(father_df, child_df): 1개의 사용 위치 mjmj1
    for index, row in child_df.iterrows():
        if row['check_code'] in father_df['CourseCode'].values:
            father_code = father_df.loc[father_df['CourseCode'] == row['check_code'], 'CourseCode'].values[0]
            child_df.at[index, 'FatherCode'] = father_code

    return child_df

def check_child_code(child_df): 1개의 사용 위치 mjmj1
    for index, row in child_df.iterrows():
        if row['check_code'] in child_df['CourseCode'].values:
            new_father_code = child_df.loc[child_df['CourseCode'] == row['check_code'], 'FatherCode'].values[0]
            child_df.at[index, 'FatherCode'] = new_father_code

    return child_df

def check_exception_df(child_df): 1개의 사용 위치 mjmj1
    drop_list = []

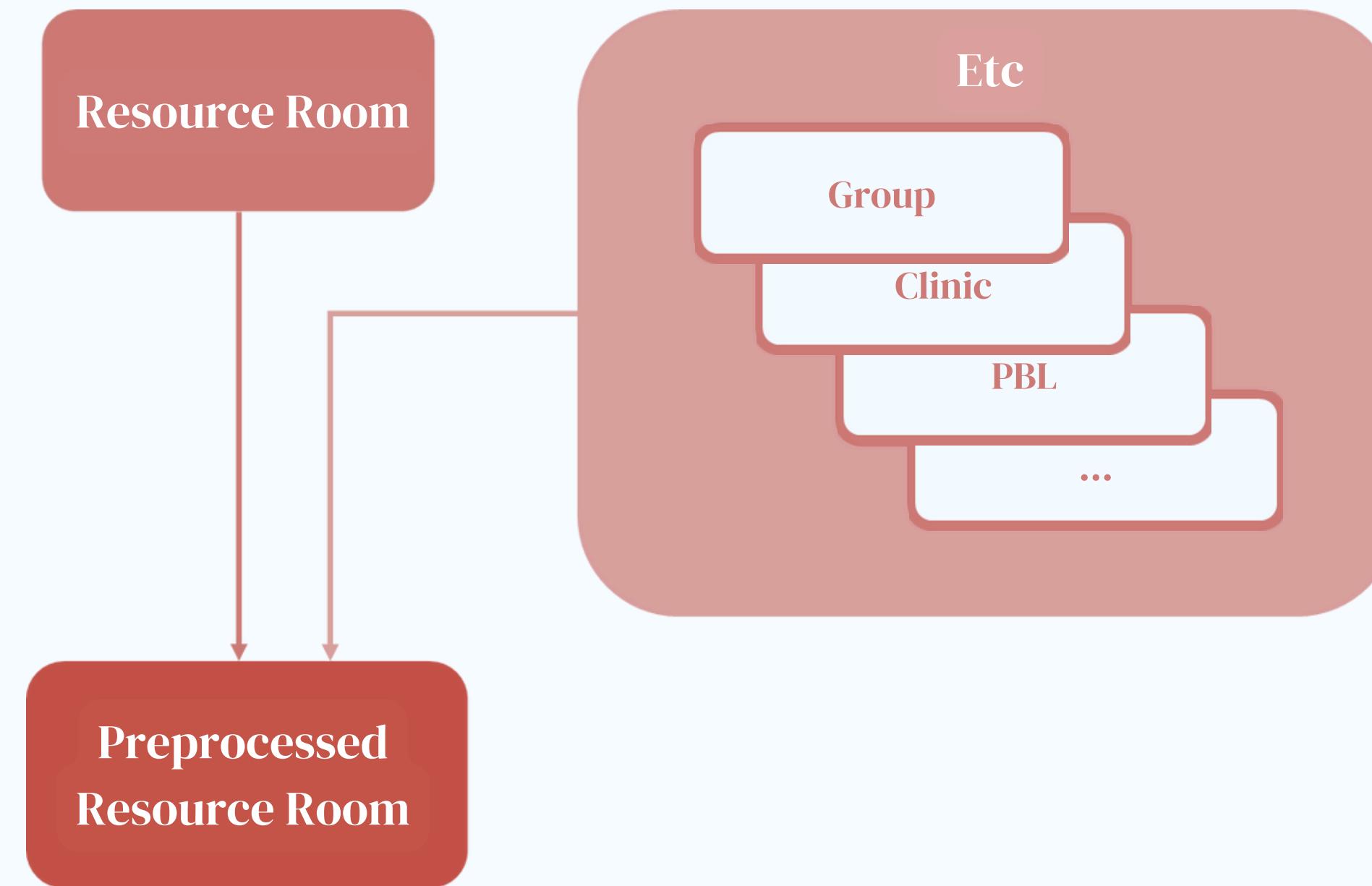
    for index, row in child_df.iterrows():
        if pd.isna(row['FatherCode']):
            add_exception_row(row)
            drop_list.append(index)

    child_df = child_df.drop(drop_list).reset_index(drop=True)

    return child_df
```

```
def make_family_tables(df): 1개의 사용 위치 mjmj1
    father_df, child_df = separate_table(df)
    child_df = check_father_code(father_df, child_df)
    child_df = check_child_code(child_df)
    child_df = check_exception_df(child_df)
    father_df = make_courseType_fatherCode(father_df)
```

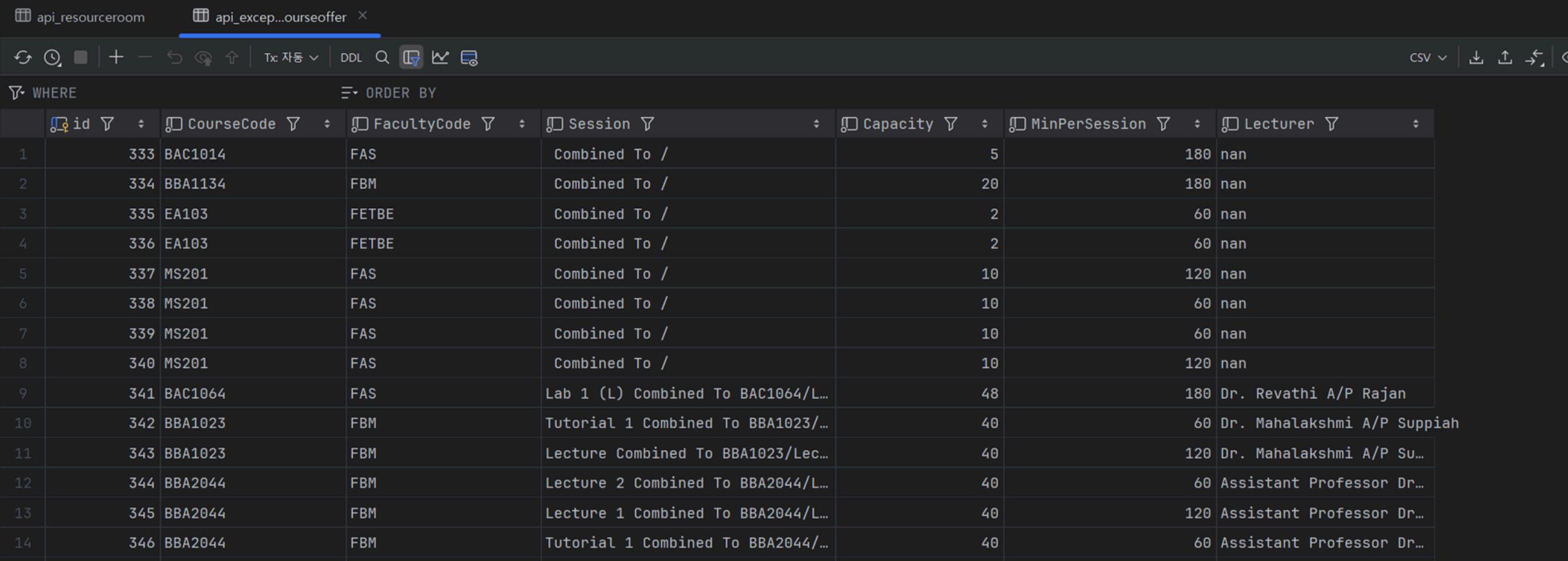
### 3. Data migration : ResourceRoom



---

Because the course code doesn't clearly indicate the lecture's content,  
all rooms with a special purpose are categorized as "etc."

# 4. Database / Table Creation



	<input type="checkbox"/> id	<input type="checkbox"/> CourseCode	<input type="checkbox"/> FacultyCode	<input type="checkbox"/> Session	<input type="checkbox"/> Capacity	<input type="checkbox"/> MinPerSession	<input type="checkbox"/> Lecturer
1		333	BAC1014	FAS	Combined To /	5	180 nan
2		334	BBA1134	FBM	Combined To /	20	180 nan
3		335	EA103	FETBE	Combined To /	2	60 nan
4		336	EA103	FETBE	Combined To /	2	60 nan
5		337	MS201	FAS	Combined To /	10	120 nan
6		338	MS201	FAS	Combined To /	10	60 nan
7		339	MS201	FAS	Combined To /	10	60 nan
8		340	MS201	FAS	Combined To /	10	120 nan
9		341	BAC1064	FAS	Lab 1 (L) Combined To BAC1064/L...	48	180 Dr. Revathi A/P Rajan
10		342	BBA1023	FBM	Tutorial 1 Combined To BBA1023/...	40	60 Dr. Mahalakshmi A/P Suppiah
11		343	BBA1023	FBM	Lecture Combined To BBA1023/Lec...	40	120 Dr. Mahalakshmi A/P Su...
12		344	BBA2044	FBM	Lecture 2 Combined To BBA2044/L...	40	60 Assistant Professor Dr...
13		345	BBA2044	FBM	Lecture 1 Combined To BBA2044/L...	40	120 Assistant Professor Dr...
14		346	BBA2044	FBM	Tutorial 1 Combined To BBA2044/...	40	60 Assistant Professor Dr...
15		347	BBA2143	FBM	Lecture Combined To BBA2143/Lec...		
16		348	BBA2143	FBM	Tutorial 1 Combined To BBA2143/...		
17		349	BBA2163	FBM	Lecture Combined To BBA2163/Lec...		
18		350	BBA2163	FBM	Tutorial 2 Combined To BBA2163/...		
19		351	BBA2163	FBM	Tutorial 1 Combined To BBA2163/...		
20		352	BBA2204	FBM	Lecture 1 Combined To BBA2204/L...		
21		353	BBA2204	FBM	Tutorial 1 Combined To BBA2204/...		
22		354	BBA2204	FBM	Lecture 2 Combined To BBA2204/L...		
23		355	BBA2293	FBM	Tutorial 2 Combined To BBA2293/...		
24		356	BBA2293	FBM	Lecture Combined To BBA2293/Lec...		
25		357	BBA2413	FBM	Lecture Combined To BBA2413/Lec...		
26		358	BBA2413	FBM	Tutorial 1 Combined To BBA2413/...		

```
class ExceptionCourseOffer(models.Model): 2개의 사용 위치 mjmj1
    CourseCode = models.CharField(max_length=20)
    FacultyCode = models.CharField(max_length=20)
    Session = models.CharField(max_length=255)
    Capacity = models.IntegerField()
    MinPerSession = models.IntegerField()
    Lecturer = models.CharField(max_length=255)

    def __str__(self): mjmj1
        return f"{self.CourseCode}"
```

# 4. Database / Table Creation

api\_fathercourseoffer

WHERE ORDER BY

	id	CourseCode	FacultyCode	Session	Capacity	MinPerSession	Lecturer	CourseType
1	1617	AB443	SABE	Lecture	25	180	Nurul Anida Mohamad	Le
2	1618	AB516	SABE	Lecture Father C...	35	450	Lim Ker Chwing	Le
3	1619	AB533	SABE	Lecture Father C...	30	180	Asst. Prof. IDr Baizu...	Le
4	1620	AB618	SABE	Lecture 2	20	240	Nurul Anida Mohamad	Le
5	1621	AB618	SABE	Lecture 1	20	210	Nurul Anida Mohamad	Le
6	1622	AB623	SABE	Lecture	48	180	Ar. Tee Khay Mee	Le
7	1623	AB633	SABE	Lecture	10	180	Tan Han Leong	Le
8	1624	AD133	SABE	Lecture 1 (S)	10	180	Amar Muhammad Ramli	Le
9	1625	AD133	SABE	Lecture 2 (S)	10	180	Amar Muhammad Ramli	Le
10	1626	AD332	SABE	Lecture 1 (S)	10	180	Nor Syawallina Azman	Le
11	1627	AD332	SABE	Lecture 2 (S)	10	180	Nor Syawallina Azman	Le
12	1628	AI663	SABE	Lecture 1 (S)	10	180	Nur Rafida Hamzah	Le
13	1629	AI663	SABE	Lecture 2 (S)	10	180	Nur Rafida Hamzah	Le
14	1630	AR423	SABE	Lecture	30	180	Dr. Wong Leong Yee	Le
15	1631	AR433	SABE	Lecture				
16	1632	AR453	SABE	Lecture				
17	1633	AR533	SABE	Lecture Father C...				
18	1634	AR724	SABE	Lecture Father C...				
19	1635	AR853	SABE	Lecture Father C...				
20	1636	AR873	SABE	Lecture Father C...				
21	1637	BAA2023	FAS	Lecture 1 (S) Fa...				
22	1638	BAA2023	FAS	Lecture 4 (S) Fa...				
23	1639	BAA2023	FAS	Lecture 3 (S) Fa...				
24	1640	BAA2023	FAS	Lecture 2 (S) Fa...				
25	1641	BAB1044	FAS	Lecture 2 (S) Fa...				
26	1642	BAB1044	FAS	Lecture 1 (S) Fa...				

## \*Course Type

- 'Le' : Lecture
- 'Tu' : Tutorial
- 'Gr' : Group
- 'Cl' : Clinic
- 'La' : Lab
- 'PB' : PBL
- 'Ki' : Kitchen
- 'Dr' : Drawing

```
class FatherCourseOffer(models.Model): 9개의 사용 위치 mjmj1
    id = models.AutoField(primary_key=True)
    CourseCode = models.CharField(max_length=20)
    FacultyCode = models.CharField(max_length=20)
    Session = models.CharField(max_length=255)
    Capacity = models.IntegerField()
    MinPerSession = models.IntegerField()
    Lecturer = models.CharField(max_length=255)
    CourseType = models.CharField(max_length=10)

    def __str__(self): mjmj1
        return f'{self.CourseCode} ({self.Session})'
```

# 4. Database / Table Creation

api\_childcourseoffer

WHERE ORDER BY

	id	FatherCode	CourseCode	Capacity	Session
1	367	BEI1083	AB243	5	Lecture Combined To BEI10...
2	368	BEI2017	AB316	5	Lecture Combined To BEI20...
3	369	BEI2033	AB343	5	Lecture Combined To BEI20...
4	370	BEI2067	AB416	10	Lecture 1 Combined To BEI...
5	371	BEI2067	AB416	10	Lecture 2 Combined To BEI...
6	372	BEI2074	AB423	5	Lecture Combined To BEI20...
7	373	BEA2053	AB543	20	Lecture Combined To BEA20...
8	374	BEI2053	AB803	15	Lecture Combined To BEI20...
9	375	MEA2018	AM531	5	Lecture 2 Combined To MEA...
10	376	MEA2018	AM531	5	Lecture 1 Combined To MEA...
11	377	MEA2018	AM541	10	Lecture 2 Combined To MEA...
12	378	MEA2018	AM541	10	Lecture 1 Combined To MEA...
13	379	MEA2074	AM542	5	Lecture Combined To MEA20...
14	380	BEA1063	AR253	10	Lecture Combined To BEA10...
15	381	BEA2018	AR315	5	Lecture Combined To BEA20...
16	382	BEA2023	AR323	10	Lecture Combined To BEA20...
17	383	BEA2033	AR333	5	Lecture Combined To BEA20...
18	384	BEA2053	AR343	5	Lecture Combined To BEA20...
19	385	BEA2068	AR415	10	Lecture Combined To BEA20...
20	386	BEA2068	AR415	10	Lecture Combined To BEA20...
21	387	BEA2083	AR443	7	Lecture Combined To BEA20...
22	388	BEA2073	AR463	10	Lecture Combined To BEA20...
23	389	BEA3018	AR518	10	Lecture Combined To BEA30...
24	390	BEA3018	AR518	10	Lecture Combined To BEA30...
25	391	BEA3034	AR524	10	Lecture Combined To BEA30...
26	392	BEA3023	AR543	30	Lecture Combined To BEA30...
27	393	BEA3023	AB616	10	Lecture Combined To BEA30...

```
class ChildCourseOffer(models.Model): 5개의 사용 위치  ↗ mjmj1
    id = models.AutoField(primary_key=True)
    FatherCode = models.CharField(max_length=20)
    CourseCode = models.CharField(max_length=20)
    Capacity = models.IntegerField()

    def __str__(self): ↗ mjmj1
        return f'{self.FatherCode} - {self.CourseCode} ({self.Capacity})'
```

# 4. Database / Table Creation

api\_resourceroom

WHERE ORDER BY

			Description	Capacity	Lecture	Tutorial	Lab	ETC
1	1688	C209	C209 - [C209]	102	Y	Y	N	N
2	1689	C403	C403 - [C403]	50	Y	Y	N	N
3	1690	C405	C405 - [C405]	135	Y	Y	N	N
4	1691	C401	C401 - [C401]	40	Y	Y	N	N
5	1692	C407	C407 - [C407]	100	Y	Y	N	N
6	1693	MPH - GBS	MPH - [MULTIPURPOSE HALL]	100	Y	Y	Y	N
7	1694	C121 - iMus	C121 - iMus	100	Y	Y	N	Y
8	1695	C123 - iMus	C123 - iMus	100	Y	Y	N	Y
9	1696	C127	C127 - Chamber Room	12	Y	Y	N	N
10	1697	C Lab A	C Lab A - [COMPUTER LAB A]	47	N	N	Y	N
11	1698	C Lab B	C Lab B - [COMPUTER LAB B]	40	N	N	Y	N
12	1699	C Lab C	C Lab C - [COMPUTER LAB C]	80	N	N	Y	N
13	1700	C301	C301 - [Auditorium 1]	350	Y	Y	N	N
14	1701	C410	C410 - [C410]	10	N	Y	Y	N
15	1702	C412	C412 - [C412]	0	Y	Y	N	N
16	1703	C404	C404 - [C404]	20	Y	Y	Y	N
17	1704	C402	C402 - [C402]	10	N	Y	Y	N
18	1705	C311 Physics Lab	C311 - [C311] Physics Lab	60	N	N	Y	N
19	1706	C117 - iMus	C117 - Keyboard Lab	11	Y	Y	N	Y
20	1707	C119	C119 - [C119] Percussion Room					
21	1708	G08-23	G08-23 - [Speech Lab]					
22	1709	C107	c107 - [c107]					
23	1710	CG01	CG01 - [CG01]					
24	1711	CG02	CG02 - [CG02]					
25	1712	CG03	CG03 - [CG03]					
26	1713	CG04	CG04 - [CG04]					
27	1714	CG08	CG08 - [CG08]					
28	1715	C112	C112 - [C112] Teaching Studio					
29	1716	C114	C114 - [C114] Teaching Studio					
30	1717	C115 - iMus	C115 - iMus					
31	1718	C201	C201 - [C201]					
32	1719	C202	C202 - [C202]					
33	1720	C204	C204 - [C204]					
34	1721	C205	C205 - [C205]					
35	1722	C206	C206 - [C206]					

241 행

```
class ResourceRoom(models.Model): 5개의 사용 위치 mjmj1
    id = models.AutoField(primary_key=True)
    ResourceCode = models.CharField(max_length=100)
    Description = models.CharField(max_length=255)
    Capacity = models.IntegerField()
    Lecture = models.CharField(max_length=1)
    Tutorial = models.CharField(max_length=1)
    Lab = models.CharField(max_length=1)
    ETC = models.CharField(max_length=1)

    def __str__(self): mjmj1
        return f"{self.ResourceCode} ({self.Capacity})"
```

# 4. Database / Table Creation

api\_resourceroom api\_assignable

WHERE ORDER BY

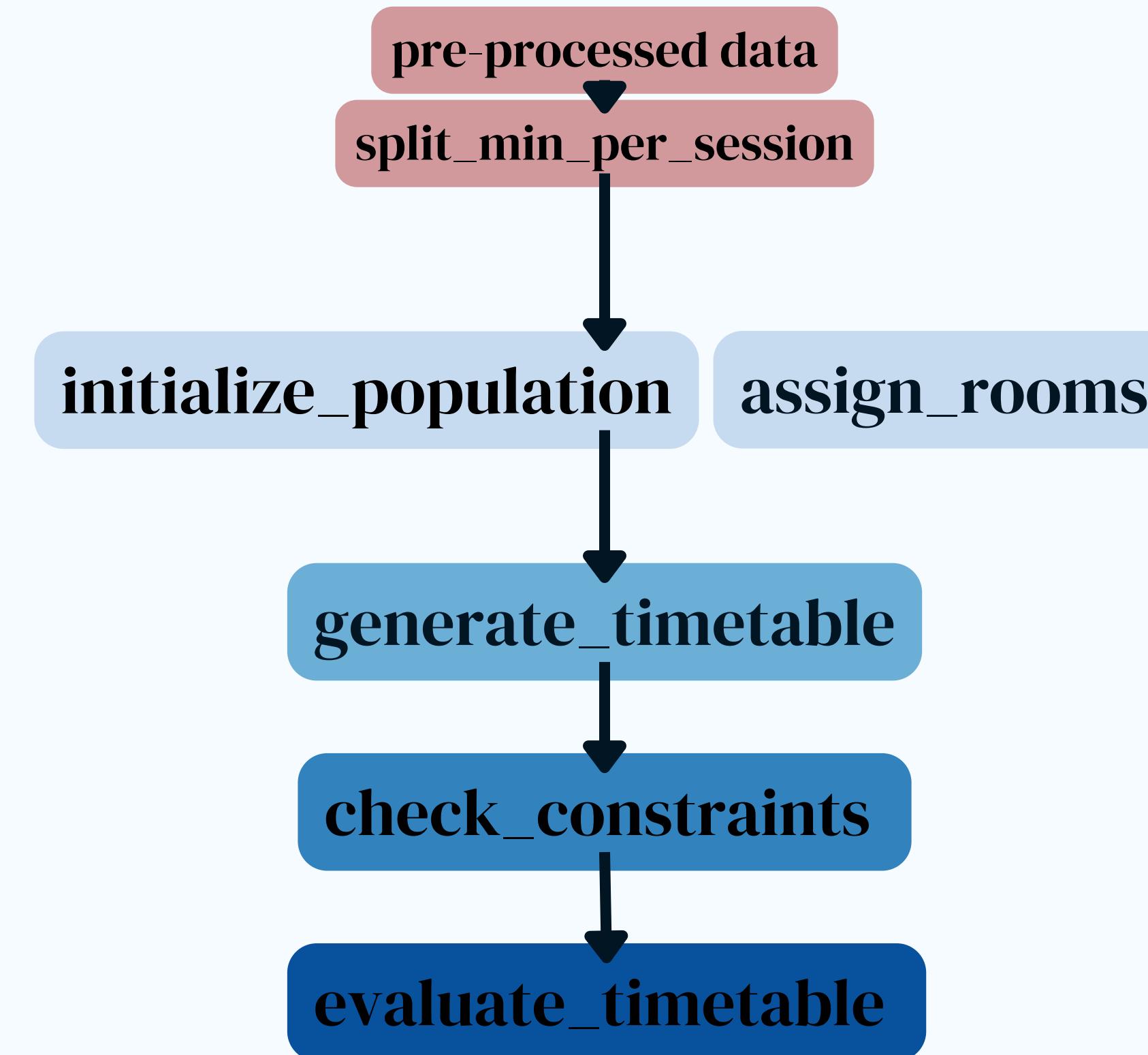
	CourseCode	ResourceCode	FacultyCode	Session	Lecturer	DayOfWeek	RoomCapacity	CourseCapacity	StartTime
1	AB443	G08-12	SABE	Lecture	Nurul Anida Mohamad	Thursday	25	25	10:00 - 11:00
2	AB516	C403	SABE	Lecture Father Class	Lim Ker Chwing	Monday	50	50	08:00 - 09:00
3	AB516	C107	SABE	Lecture Father Class	Lim Ker Chwing	Thursday	50	50	14:00 - 15:00
4	AB516	C201	SABE	Lecture Father Class	Lim Ker Chwing	Tuesday	50	50	10:00 - 11:00
5	AB533	C401	SABE	Lecture Father Class	Asst. Prof. IDR Baizura Hanim Bt Bidin	Friday	40	40	14:00 - 15:00
6	AB618	C404	SABE	Lecture 2	Nurul Anida Mohamad	Tuesday	20	20	10:00 - 11:00
7	AB618	G08-21	SABE	Lecture 2	Nurul Anida Mohamad	Tuesday	20	20	14:00 - 15:00
8	AB618	CA2-01	SABE	Lecture 1	Nurul Anida Mohamad	Monday	20	20	08:00 - 09:00
9	AB618	CA2-02	SABE	Lecture 1	Nurul Anida Mohamad	Friday	20	20	14:00 - 15:00
10	AB623	CA1-01	SABE	Lecture	Ar. Tee Khay Mee	Thursday	48	48	10:00 - 11:00
11	AB633	C315	SABE	Lecture	Tan Han Leong	Tuesday	10	10	10:00 - 11:00
12	AD133	AV_Room	SABE	Lecture 1 (S)	Amar Muhammad Ramli	Wednesday	10	10	10:00 - 11:00
13	AD133	Prodn_Rm	SABE	Lecture 2 (S)	Amar Muhammad Ramli	Monday	10	10	10:00 - 11:00
14	AD332	Editing_Rm	SABE	Lecture 1 (S)	Nor Syawallina Azman	Friday	10	10	10:00 - 11:00
15	AD332	C111	SABE	Lecture 2 (S)	Nor Syawallina Azman	Tuesday	10	10	10:00 - 11:00
16	AI663	Food_Media_A	SABE	Lecture 1 (S)	Nur Rafida Hamzah	Tuesday	10	10	10:00 - 11:00
17	AI663	Food_Media_B	SABE	Lecture 2 (S)	Nur Rafida Hamzah	Friday	10	10	10:00 - 11:00
18	AR423	C3M07	SABE	Lecture	Dr. Wong Leong Yee	Thursday	30	30	10:00 - 11:00
19	AR433	FASMeetingRoom1	SABE	Lecture	Ts. Dr. Khairul Zah				
20	AR453	C119	SABE	Lecture	Nurhayati Zulkepli				
21	AR533	E01-02	SABE	Lecture Father Class	Ar. Norizan Daud				
22	AR724	C602	SABE	Lecture Father Class	Bakhtiar Amir				
23	AR853	GBS_Room4	SABE	Lecture Father Class	James Lim Chin Wan				
24	AR873	B501	SABE	Lecture Father Class	Cheah Ee Von				
25	BAA2023	CA-102	FAS	Lecture 1 (S) Father Class	Asst. Prof. Dr. C				
26	BAA2023	C208	FAS	Lecture 4 (S) Father Class	Asst. Prof. Dr. C				
27	BAA2023	C407	FAS	Lecture 3 (S) Father Class	Asst. Prof. Dr. C				
28	BAA2023	MPH - GBS	FAS	Lecture 2 (S) Father Class	Asst. Prof. Dr. C				
29	BAB1044	C405	FAS	Lecture 2 (S) Father Class	Dr. Lau Yin Yin				
30	BAB1044	E02-06	FAS	Lecture 1 (S) Father Class	Dr. Lau Yin Yin				
31	BAB1044	GG06	FAS	Lecture 3 (S) Father Class	Dr. Lau Yin Yin				
32	BAB1053	G08-23	FAS	Lecture 4 (S)	Assoc. Prof. Dr.				
33	BAB1053	G08-11	FAS	Lecture 2 (S)	Assoc. Prof. Dr.				
34	BAB1053	GTR	FAS	Lecture 1 (S)					
35	BAB1053	Group_Therapy_Room	FAS	Lecture 3 (S)					

```
class AssignTable(models.Model): 7개의 사용 위치 mjmj1
    id = models.AutoField(primary_key=True)
    CourseCode = models.CharField(max_length=20)
    ResourceCode = models.CharField(max_length=100)
    FacultyCode = models.CharField(max_length=20)
    Session = models.CharField(max_length=255)
    Lecturer = models.CharField(max_length=255)
    CombineBy = models.CharField(max_length=50)
    DayOfWeek = models.CharField(max_length=10)
    RoomCapacity = models.IntegerField()
    CourseCapacity = models.IntegerField()
    StartTime = models.IntegerField()
    MinPerSession = models.IntegerField()

    def __str__(self): mjmj1
        return f'{self.ResourceCode} ({self.CourseCode})'
```

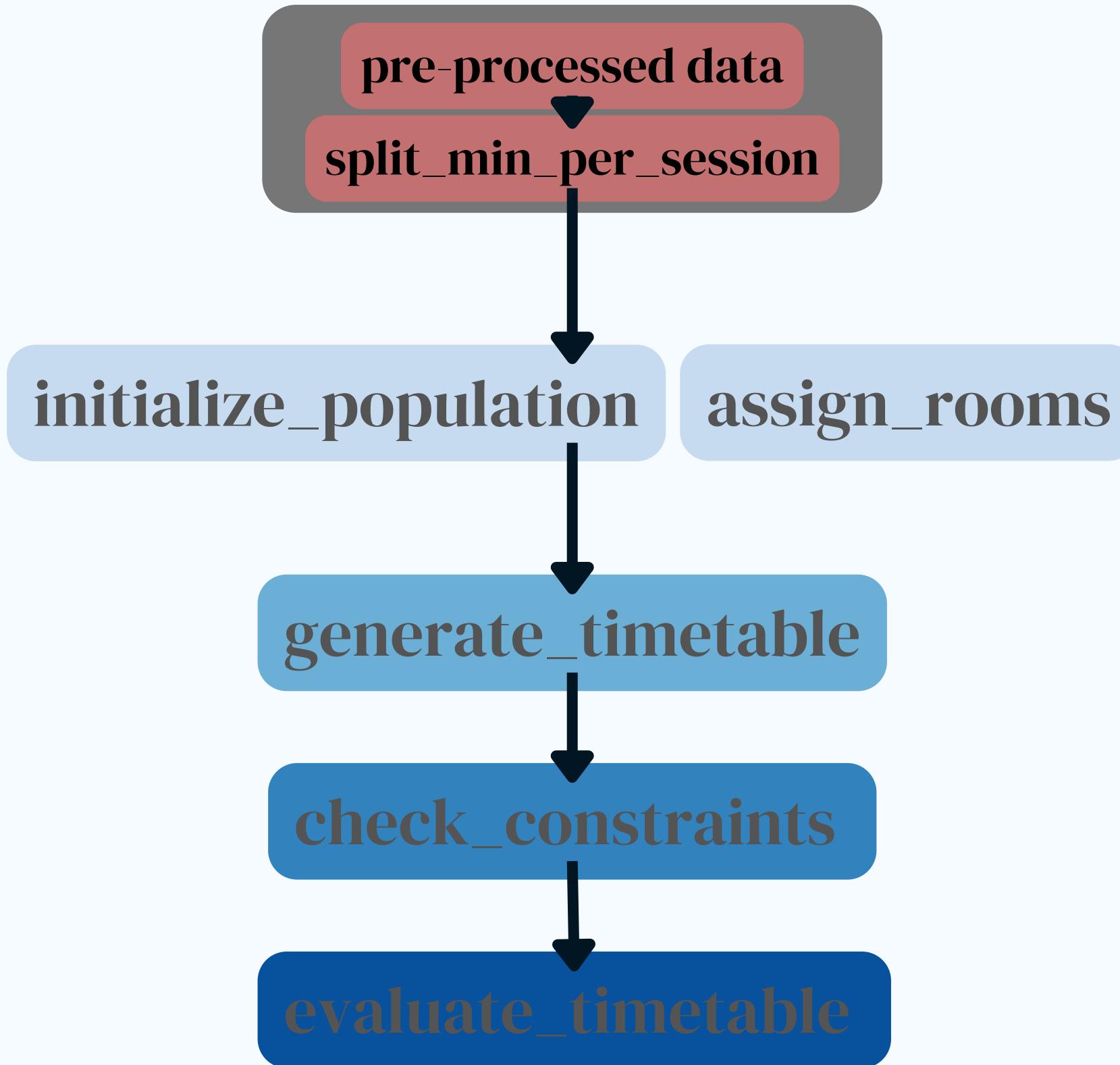
## 5. AI Algorithm Development

---



## 5. AI Algorithm Development

---



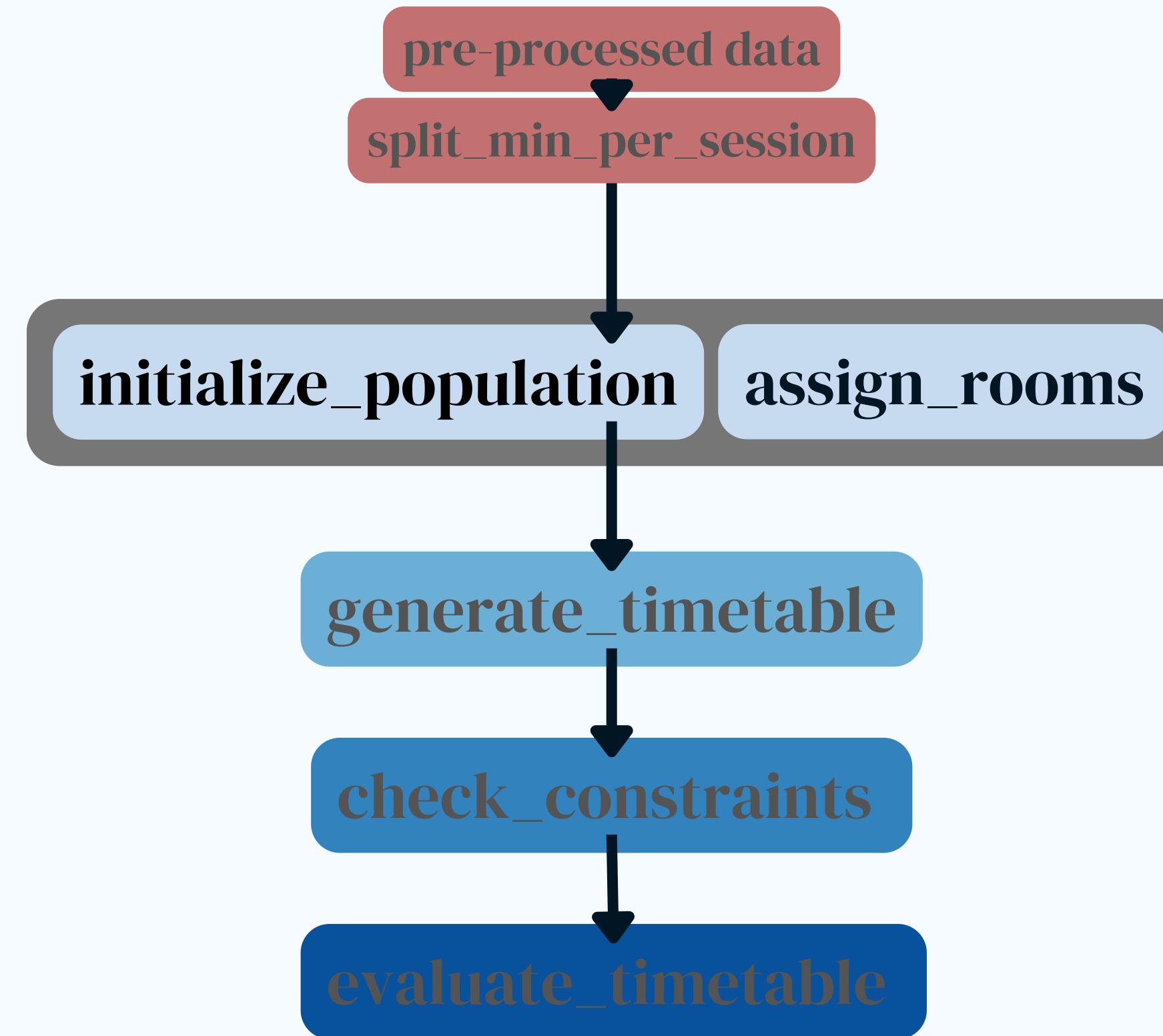
### Considerations

Courses are scheduled in blocks of up to 3 hours.

$\text{MinPerSession} \geq 180$

Split “Split\_ID” by 3 hours

# 5. AI Algorithm Development



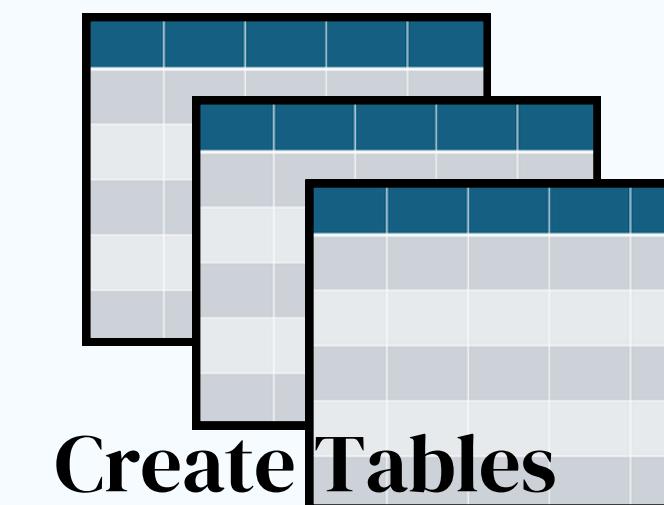
## Considerations

Course must hold 8 am ~ 10 pm.

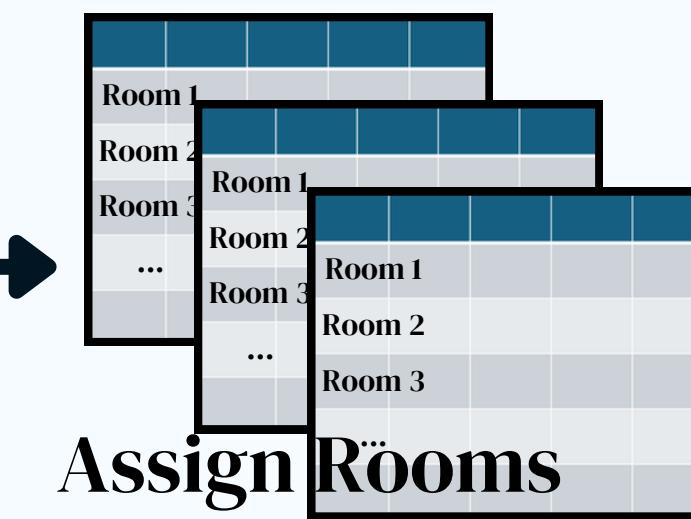
Time Conflict

Course Type

capacity

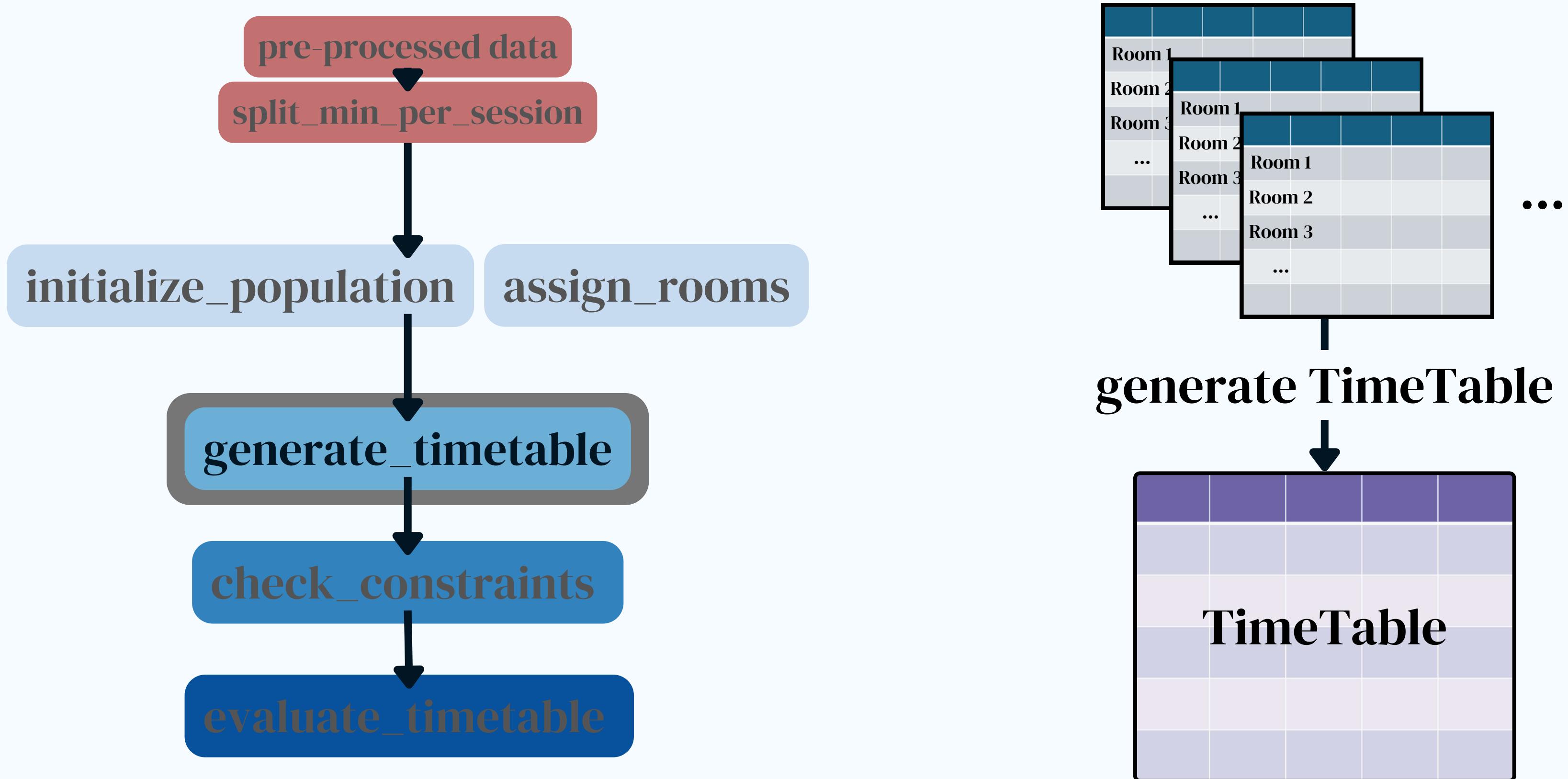


...



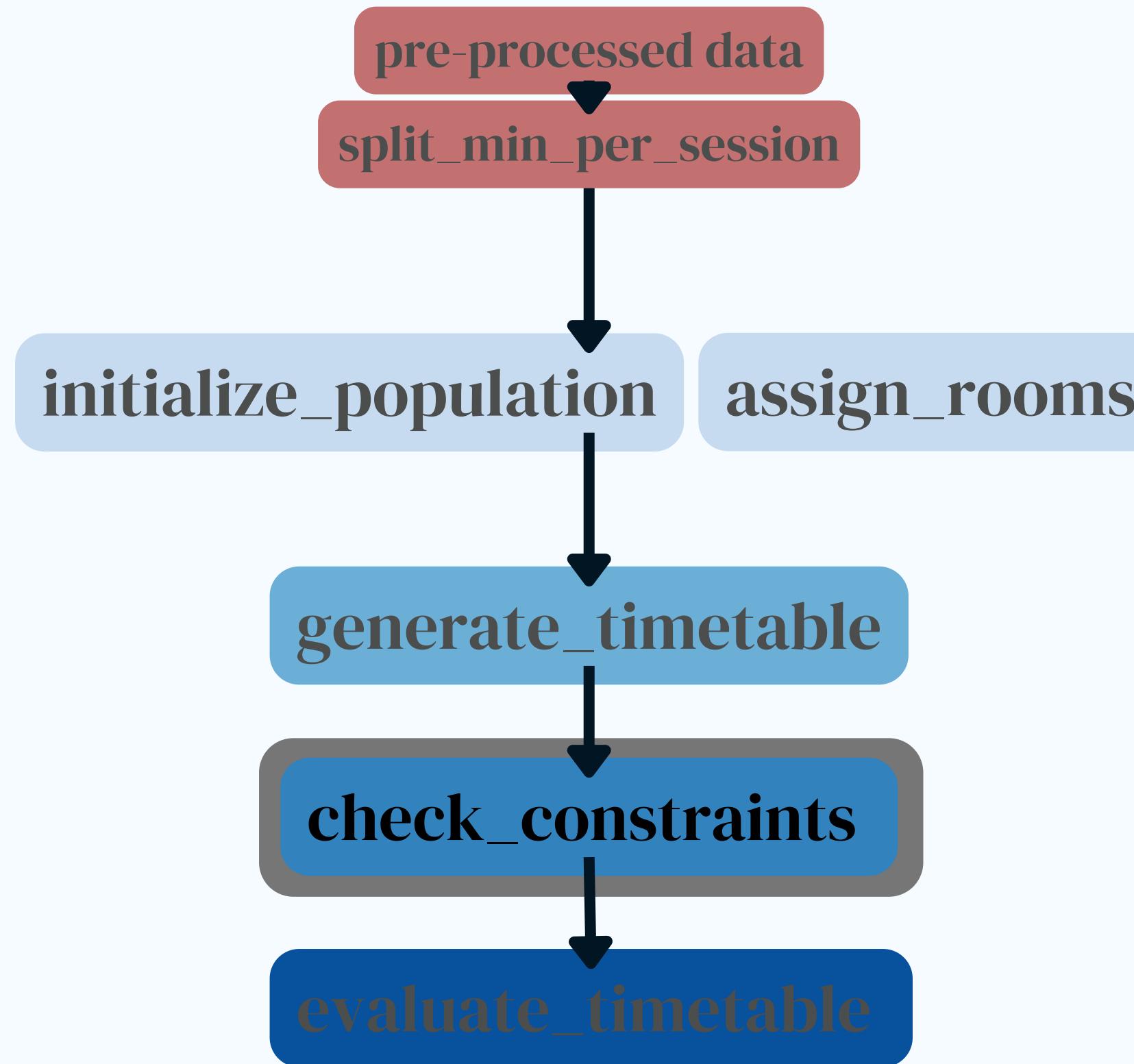
...

## 5. AI Algorithm Development



## 5. AI Algorithm Development

---



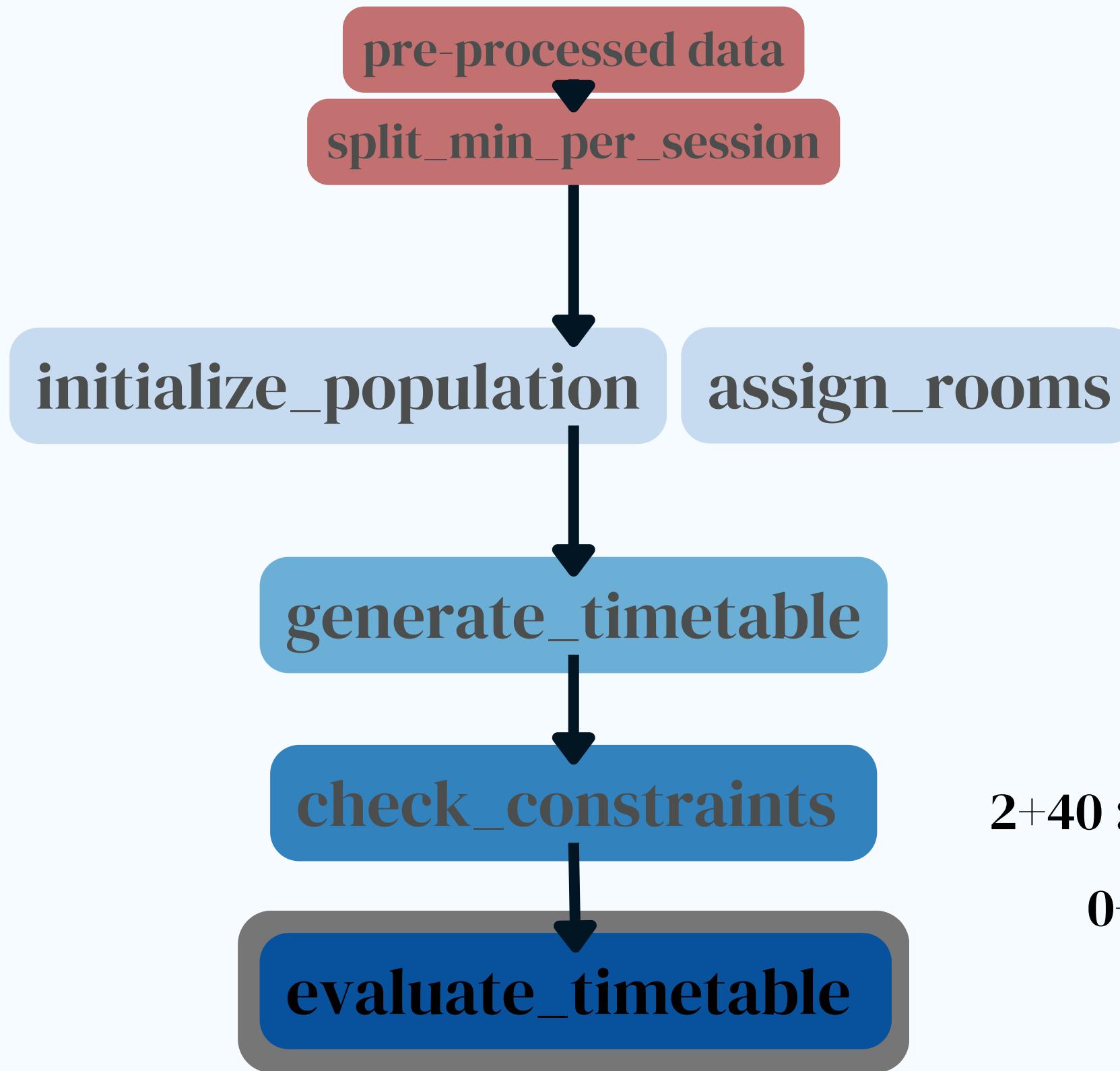
✓ Classroom duplication

✓ Lecturer duplication

✓ Overcapacity in classrooms

✓ Lecture Time Conflict

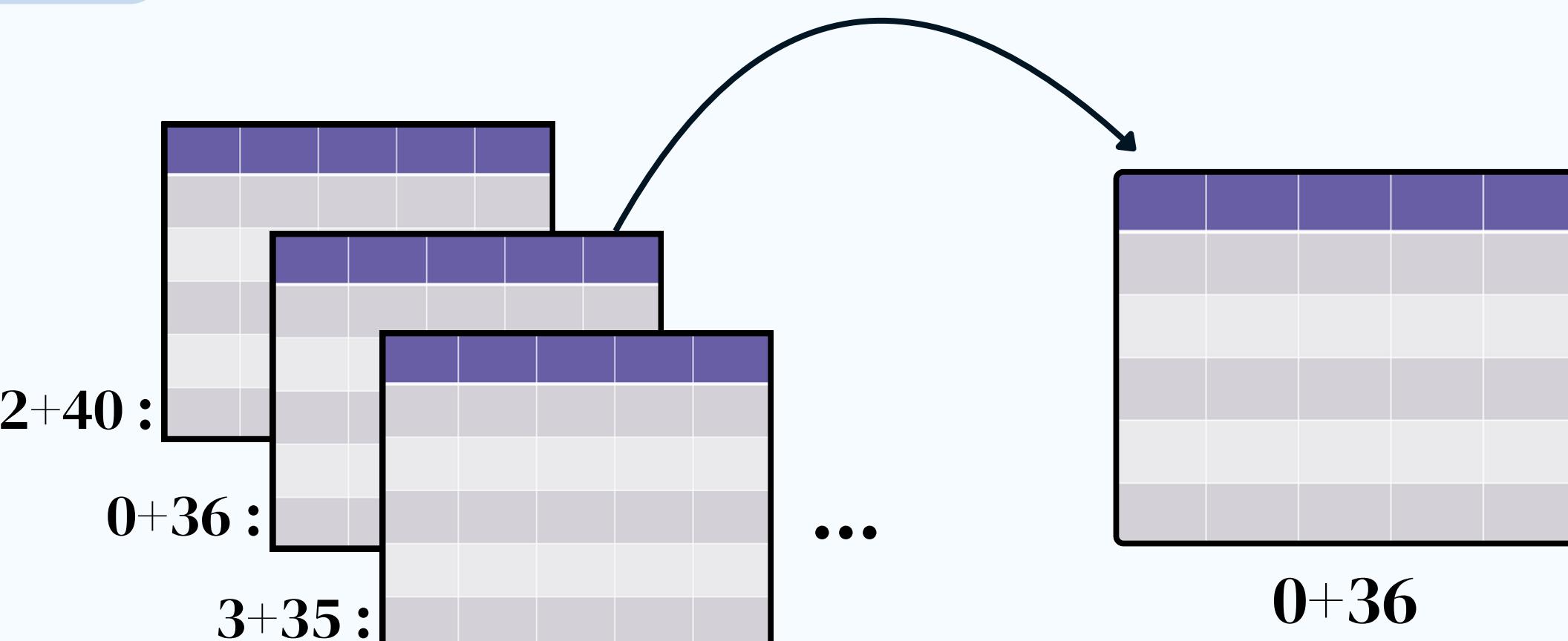
## 5. AI Algorithm Development



**Score :**

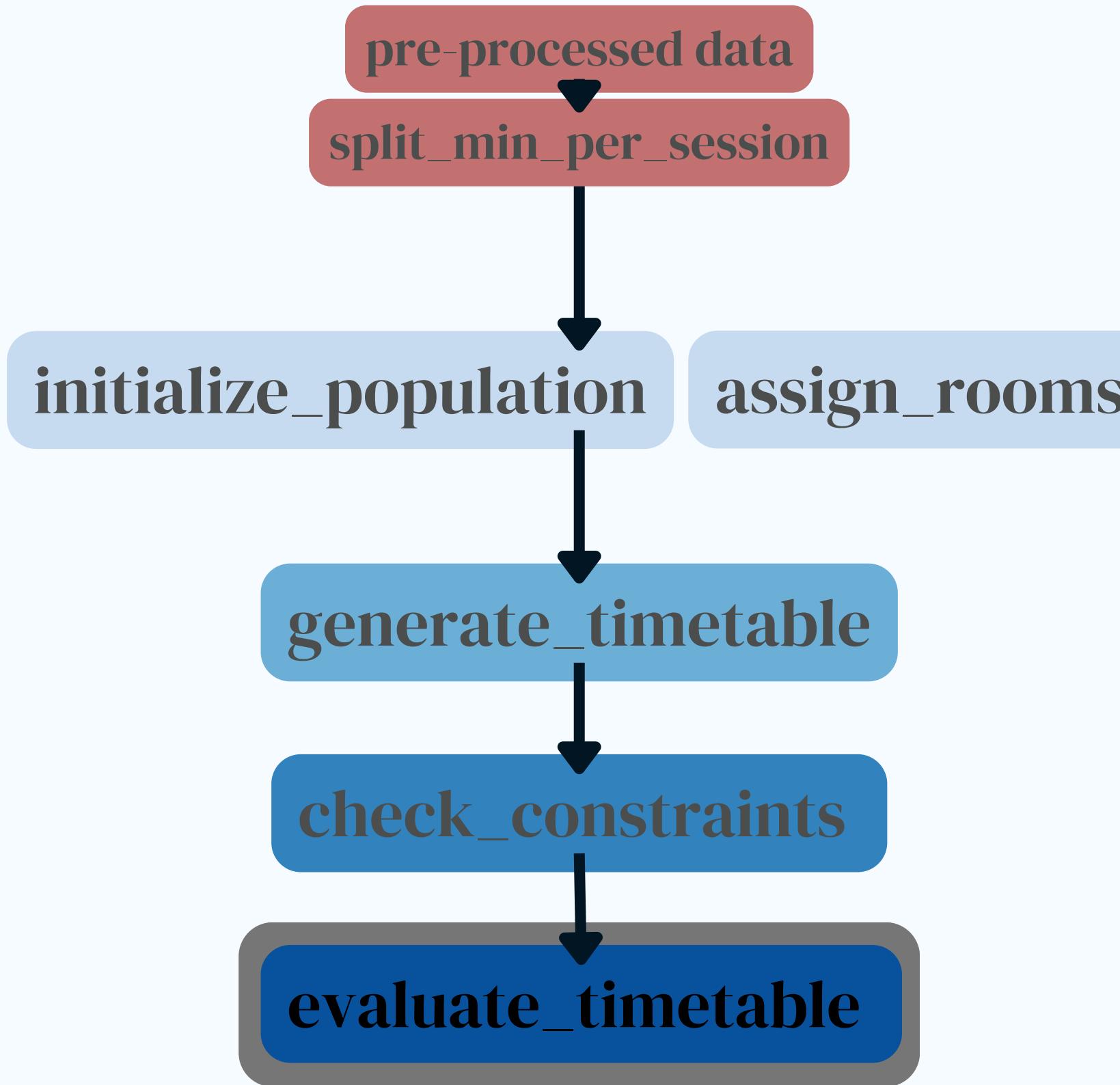
Count of  
constraint violations +

Count of unused  
resource rooms



## 5. AI Algorithm Development

---



**<Allocation>**

**Number of assigned classes**  
Total number of classes

**<Resource Utilization>**

**Number of used classrooms**  
Total number of classrooms

**<Type Matching>**

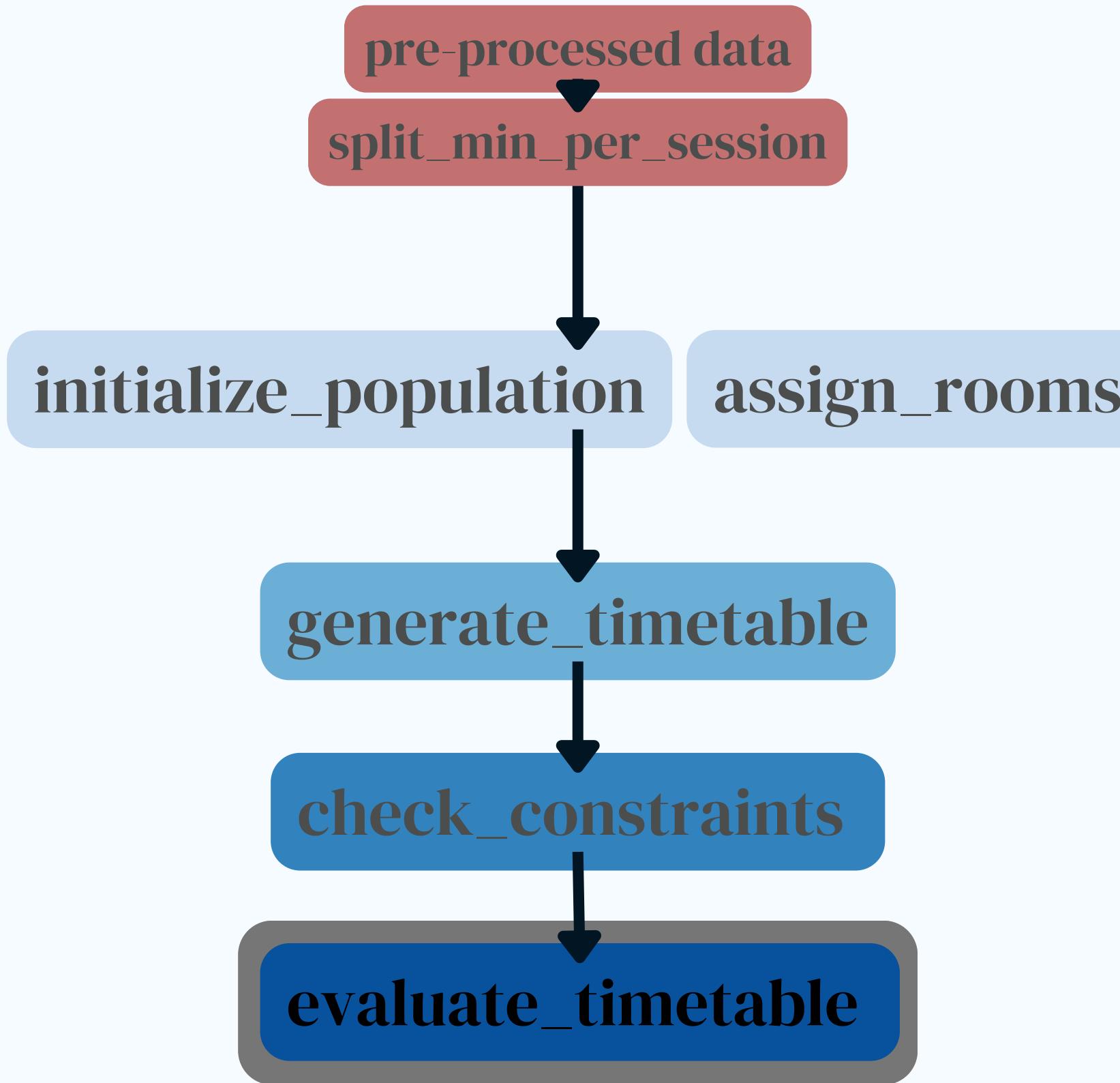
: Percentage of classes  
where lecture type matches classroom type

**<Classroom Fit>**

: Evaluation of how well  
classroom size matches class enrollment

## 5. AI Algorithm Development

---



Final  
TimeTable

Allocation: 100.00%

Resource utilization: 84.65%

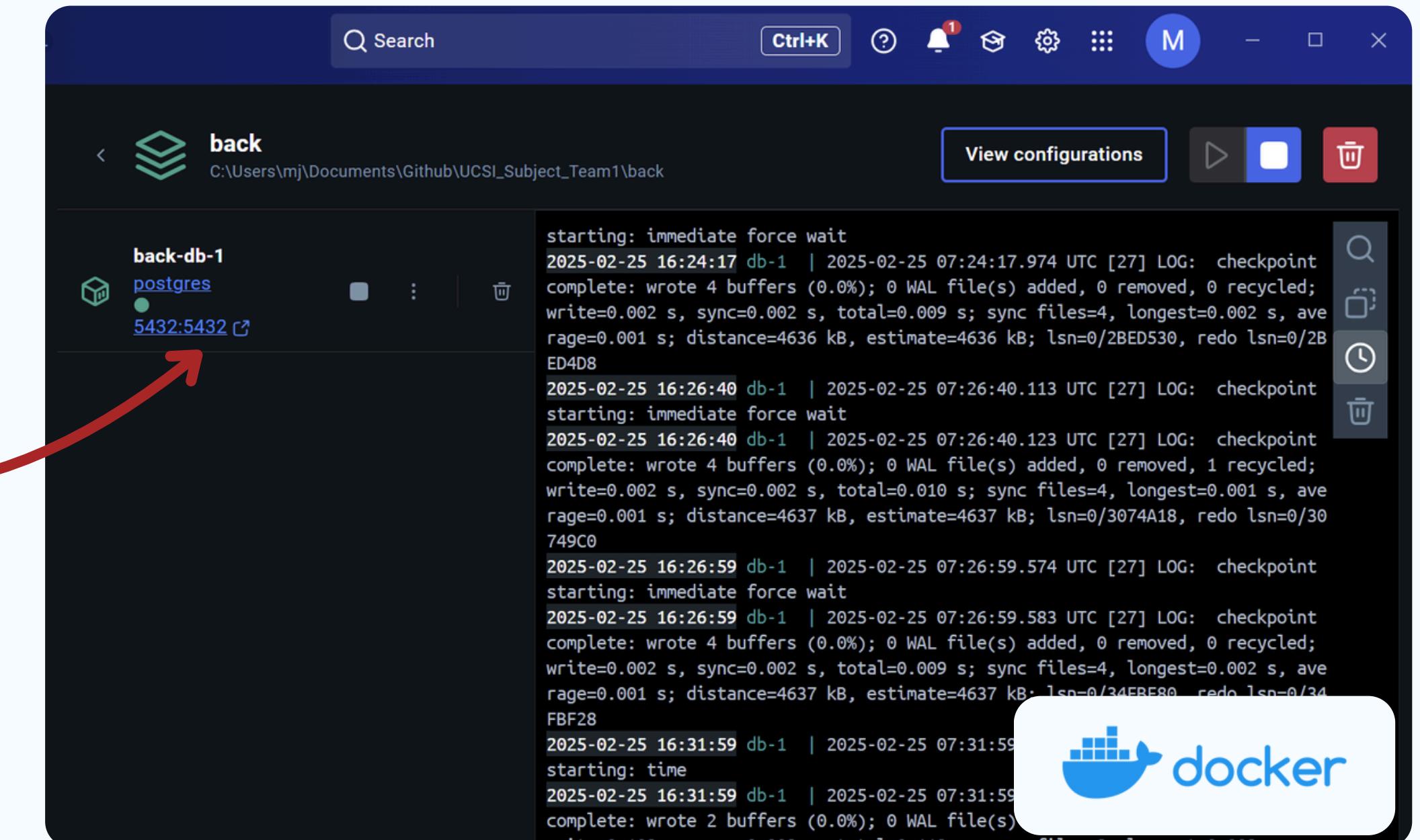
Type matching: 95.74%

Classroom fit: 90.91%

# 6. Server construction : PostgreSQL DB

```
version: '3.8'
services:
  db:
    image: postgres
    environment:
      POSTGRES_DB: project_db
      POSTGRES_USER: postgres
      POSTGRES_PASSWORD: 123456
    ports:
      - "5432:5432"
    volumes:
      - pgdata:/var/lib/postgresql/data
volumes:
  pgdata:
```

docker-compose.yml



# 6. Server construction

```
urlpatterns = [
    path("table/data/course/import/", views.make_course, name="make_course"),
    path("table/data/resource/import/", views.make_resource, name="make_resource"),
    path("table/assign/create/", views.make_assign_table, name="make_assign_table"),
    path("table/make/", views.make, name="make"),
    path("table/view/", views.get_assign_table, name="get_assign_table"),
    path("courseoffer/view/", views.get_father_course_offer, name="get_father_course_offer"),
]
```

```
def get_assign_table(request): 1개의 사용 위치 🔍 mjmj1
    code = request.GET.get('facultyCode', None)

    if code is None:
        return JsonResponse("")

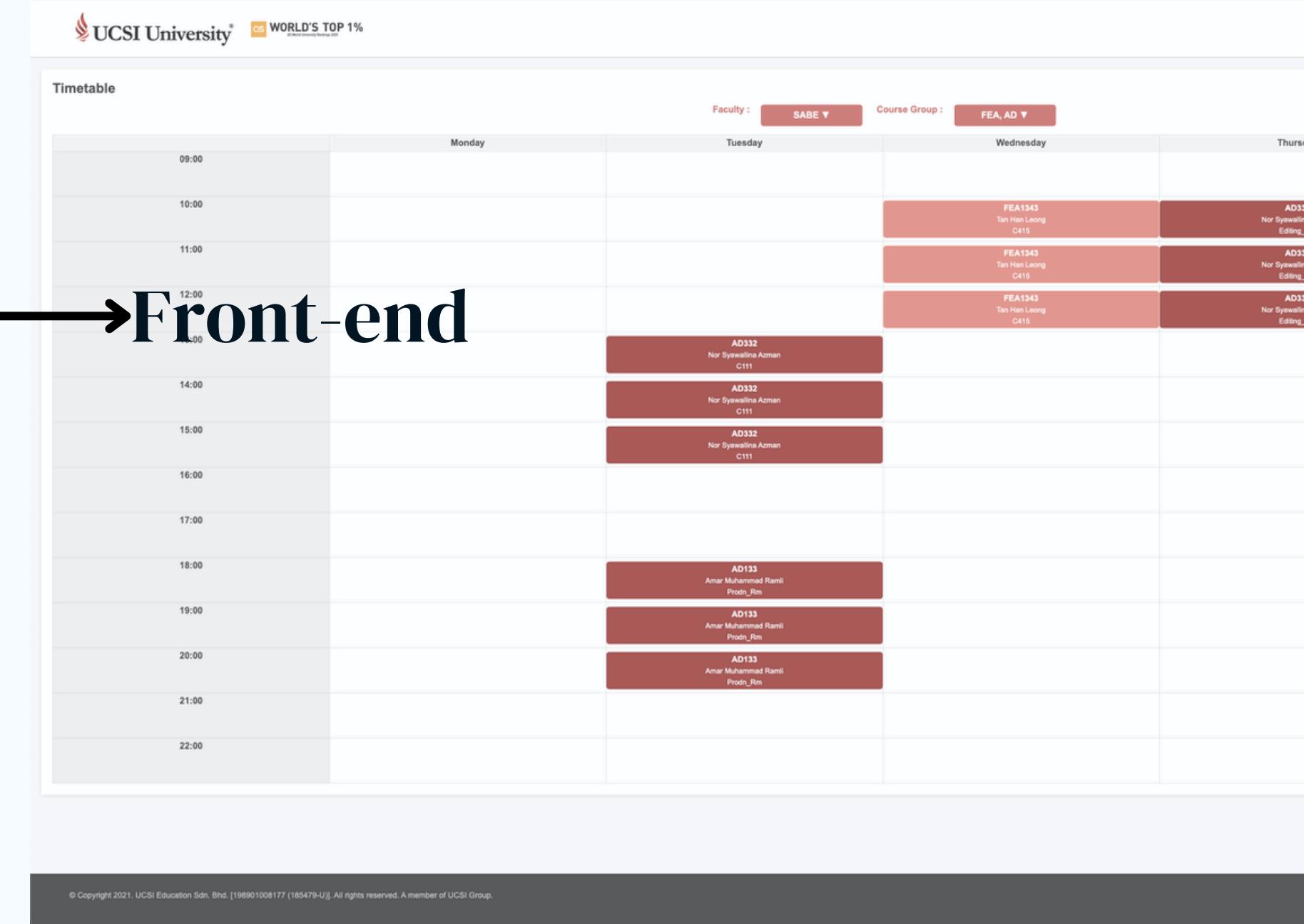
    filtered_data = AssignTable.objects.filter(FacultyCode=code)
    return JsonResponse(list(filtered_data.values()), safe=False)
```

# 6. Server construction

`http://localhost:8000/api/table/view/?facultyCode="..."`

request

```
{  
    "id": 1617,  
    "CourseCode": "AB443",  
    "FacultyCode": "SABE",  
    "Session": "Lecture",  
    "Capacity": 25,  
    "MinPerSession": 180,  
    "Lecturer": "Nurul Anida Mohamad",  
    "CourseType": "Le"  
},  
{  
    "id": 1618,  
    "CourseCode": "AB516",  
    "FacultyCode": "SABE",  
    "Session": "Lecture Father Class",  
    "Capacity": 35,  
    "MinPerSession": 450,  
    "Lecturer": "Lim Ker Chwing",  
    "CourseType": "Le"  
},  
{  
    "id": 1619,  
    "CourseCode": "AB533",  
    "FacultyCode": "SABE",  
    "Session": "Lecture Father Class",  
    "Capacity": 30,  
    "MinPerSession": 180,  
    "Lecturer": "Asst. Prof. IDr Baizura Hanim Bt Bidin",  
    "CourseType": "Le"  
},
```



The screenshot shows a university timetable for UCSI University. The header includes the university logo and 'WORLD'S TOP 1%' badge. The main section is titled 'Timetable' and shows a grid of classes from Monday to Thursday. The grid is organized by faculty (SABE) and course group (FEA, AD). Each class entry includes the course code, lecturer name, and room number. The timetable is displayed in a 24-hour format from 09:00 to 22:00.

		Monday	Tuesday	Wednesday	Thursday
	09:00				
	10:00				
	11:00				
	12:00				
	13:00				
	14:00				
	15:00				
	16:00				
	17:00				
	18:00				
	19:00				
	20:00				
	21:00				
	22:00				

© Copyright 2021, UCSI Education Sdn. Bhd. [198901008177 (185479-U)]. All rights reserved. A member of UCSI Group.

→Front-end

# 7. Screen / Dashboard Display

**Faculty :** FAS ▾

**Course Group :** Select Course ▾

	Tuesday	Wednesday	Thursday	Friday
BAA				
BAB				
BAC				
BAF				
BAN				



**UCSI University** WORLD'S TOP 1%

**Timetable**

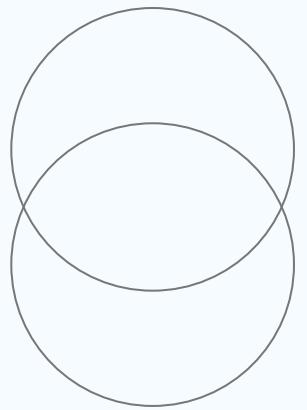
**Faculty :** SABE ▾    **Course Group :** FEA, AD ▾

	Monday	Tuesday	Wednesday	Thursday	Friday
09:00					
10:00			FEA1343 Tan Han Leong C415	AD332 Nor Syewallina Azman Editing_Rm	
11:00			FEA1343 Tan Han Leong C415	AD332 Nor Syewallina Azman Editing_Rm	
12:00			FEA1343 Tan Han Leong C415	AD332 Nor Syewallina Azman Editing_Rm	
13:00		AD332 Nor Syewallina Azman C111			
14:00		AD332 Nor Syewallina Azman C111			AD133 Amar Muhammad Ramli AV Room
15:00		AD332 Nor Syewallina Azman C111			AD133 Amar Muhammad Ramli AV Room
16:00					AD133 Amar Muhammad Ramli AV Room
17:00					
18:00		AD133 Amar Muhammad Ramli ProdIn_Rm			
19:00		AD133 Amar Muhammad Ramli ProdIn_Rm			
20:00		AD133 Amar Muhammad Ramli ProdIn_Rm			
21:00					
22:00					

© Copyright 2021, UCSI Education Sdn. Bhd. [198901008177 (185479-U)]. All rights reserved. A member of UCSI Group.



# Demonstration.



# Thank You

---

Team 1