

## 1. Horas Comprometidas por Professor

Para calcular a quantidade total de horas que cada professor está alocado em aulas, foi necessário consultar algumas tabelas para conectar o professor à sua agenda de aulas e somar a duração de cada aula.

A consulta abaixo junta as tabelas **PROFESSOR**, **SUBJECT**, **CLASS** e **CLASS\_SCHEDULE**. Ela calcula a diferença entre o horário de término e início de cada aula, soma esses valores para cada professor e exibe o total de horas.

```
SELECT
  p.id AS professor_id,
  p.name AS nome_professor,
SUM(
  TIME_TO_SEC(TIMEDIFF(cs.end_time, cs.start_time))
) AS total_horas_comprometidas
FROM
  PROFESSOR p
JOIN
  PROFESSOR_SUBJECT ps ON p.id = ps.professor_id
JOIN
  SUBJECT s ON ps.subject_id = s.id
JOIN
  CLASS c ON s.id = c.subject_id
JOIN
  CLASS_SCHEDULE cs ON c.id = cs.class_id
GROUP BY
  p.id, p.name
ORDER BY
  total_horas_comprometidas DESC;
```

## 2. Lista de Salas com Horários Livres e Ocupados

Para gerar uma lista completa com os horários de todas as salas, mostrando quais estão livres ou ocupados, a melhor abordagem é usar SQL para buscar os dados de ocupação e uma linguagem de programação como o Python para formatar e apresentar a agenda completa.

## Passo 1. Consulta SQL para buscar todas as aulas agendadas

Executei uma consulta para obter uma lista de todas as salas, seus horários ocupados e qual matéria está sendo lecionada.

```
SELECT
    b.name AS nome_predio,
    r.name AS nome_sala,
    cs.day_of_week AS dia_da_semana,
    cs.start_time AS horario_inicio,
    cs.end_time AS horario_fim,
    s.name AS materia,
    c.code AS codigo_turma
FROM
    ROOM r
JOIN
    BUILDING b ON r.building_id = b.id
LEFT JOIN
    CLASS_SCHEDULE cs ON r.id = cs.room_id
LEFT JOIN
    CLASS c ON cs.class_id = c.id
LEFT JOIN
    SUBJECT s ON c.subject_id = s.id
ORDER BY
    nome_predio, nome_sala, dia_da_semana, horario_inicio;
```

## Passo 2: Lógica de Programação para Exibir a Grade de Horários

A partir dos dados da consulta SQL acima, escrevi um script em pseudocódigo em Python que vai fazer o seguinte:

1. **Definir os Períodos de Aula:** Criei uma lista com todos os possíveis horários da escola (ex: 08:00-09:00, 09:00-10:00, etc.) para cada dia da semana.
2. **Mapear Salas e Horários:** Criei uma estrutura de dado para armazenar o status ("Livre" ou "Ocupado") de cada horário, para cada sala. Inicializando todos como "Livre".
3. **Processar os Resultados do SQL:** Percorri os resultados da consulta. Para cada aula encontrada, marquei o horário correspondente na sua estrutura de dados como "Ocupado", adicionando a informação da matéria.
4. **Imprimir a Grade Final:** Apresentei os dados de forma organizada, mostrando cada sala e o status de cada um de seus horários ao longo da semana.

```
# Suponha que 'resultados_sql' é a lista de dados da consulta anterior
```

```
# 1. Definir a estrutura da agenda
```

```
dias_da_semana = ["Segunda-feira", "Terça-feira", "Quarta-feira", "Quinta-feira",  
"Sexta-feira"]
```

```
horarios_aula = ["08:00-09:00", "09:00-10:00", "10:00-11:00", "11:00-12:00"] # etc.
```

```
agenda_salas = {} # Ex: { "Sala 101": { "Segunda-feira": { "08:00-09:00": "Livre" } } }
```

```
# 2. Inicializar a agenda com todos os horários livres
```

```
# (código para preencher a 'agenda_salas' com "Livre")
```

```
# 3. Preencher a agenda com os horários ocupados
```

```
for aula in resultados_sql:
```

```
    sala = aula['nome_sala']
```

```
    dia = aula['dia_da_semana']
```

```
    inicio = aula['horario_inicio']
```

```
    fim = aula['horario_fim']
```

```
    materia = aula['materia']
```

```
# 4. Exibir a agenda final
```

```
for sala, agenda_dia in agenda_salas.items():
```

```
    print(f"\n--- Agenda para a Sala: {sala} ---")
```

```
    for dia, horarios in agenda_dia.items():
```

```
        print(f"    {dia}:")
```

```
        for horario, status in horarios.items():
```

```
            print(f"            {horario}: {status}")
```