

## Task Introduction

The goal of this task is to create an HR management module.

HR management system is what is used by companies to manage & organize human resources, like tracking the employee's attendance, leaves, expenses, ...etc.

In this task we will be focusing on the **attendance part only**.

It is required to use **Python & Relational Database** in the assessment, and it is ok to use any web framework like **Flask** or **Django**.

## Task Description

Design and implement an API web service that covers the following requirements:

1. As a new employee, I can register to the system & start using its functionality.
2. Only signed-in employees can log their attendance.
3. As an employee, I can log my new attendance and list the previous attendance records.
  - a. Each employee lists attendance records related to himself only.
  - b. Employee can't log attendance for another employee.
4. Each attendance record shall have check-in & check-out times.
5. Attendance validations:
  - a. Check-out can't be before check-in.
    - b.1 you can't log your check-out before logging the check-in first.
    - b.2 check-out time must be always bigger than check-in time.
  - b. User can't create attendance record for the same interval. *"No attendance interval overlapping is allowed."*

**Bonus:**

1. Allow multiple check-ins & check-outs for the same attendance record.
2. For each attendance record late arrival & early leave shall be indicated if the employee arrives late or left early. *"For simplicity working hours will be hardcoded from 9 am to 5 pm."*
3. If the employee worked more than the working hours, total overtime hours shall be computed and linked to the day worked.
  - a. For normal working days "From Sunday to Thursday", overtime will be considered when an employee exceeds 8 working hours.
  - b. For weekends "Friday & Saturday", all working hours will be considered as overtime.
  - c. For simplicity, "Friday & Saturday" will be hardcoded as weekend days.
4. Employee with admin role, can list all the attendance records for all employees.
5. Add unit tests.

## Task Deliverables

1. Public GitHub repo containing your code.
2. Sample proper data to be used for testing.
3. README.md containing
  - a. Database models used in the solution.
  - b. Dependencies & configurations needed to run the code.
  - c. How to run the code.
4. it is expected to deliver APIs that allow completing the requirements mentioned above.