

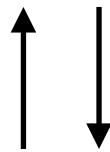
- **댕댕이의 관리자 페이지**
- **사용 언어 및 기술**
 - 언어 : Java
 - 프레임워크 : JSP / Servlet, Bootstrap
 - 데이터베이스 : Oracle, JDBC, Amazon RDS
 - 사용 API : Gson, DataTables, Full Calendar, Daum Map
- **Repository**
 - 저장소 : <https://github.com/qkrrudtjr954/DengAdmin>
- **설명**
 - DengDengE 프로젝트를 진행하면서 프로젝트를 관리하기 위한 백오피스를 구성했습니다.
 - DengDengE 웹사이트의 글 통계, 날짜 별 통계 등을 그래프로 표현하여 웹사이트의 동향을 분석할 수 있습니다.
 - amazon RDS 를 사용하여 웹서버와 DB서버의 안정적인 통신을 제공합니다.
 - ‘나쁜말 사전’ 기능을 통해 유저가 올린 글 등을 검색, 삭제가 가능하여 관리가 용이합니다.
 - 사용자와의 문의와 답변이 기능을 통해 관리자와 유저의 커뮤니케이션이 가능합니다.

Deng Admin 개발 기능 (상세)

- 메인 페이지



1. Table 이름을 변수에 담아 ajax 를 호출한다.



Controller

```
} else if(command.equals("drawChart")) {  
    AdminService adminService = AdminService.getInstance();  
  
    String table = req.getParameter("table");  
    List<GraphDayDto> days = adminService.getDateCount(table);  
  
    String json = new Gson().toJson(days);  
  
    resp.getWriter().write(json);  
}
```

2. Gson 을 사용한 Json String으로 리턴한다.

Service

```
public List<GraphDayDto> getDateCount(String table){  
    String sql = " select o.* " +  
        " from ( " +  
        "     select count(*) counts, to_char(reg_date, 'YYYY-MM-DD') days" +  
        " from "+ table +  
        " group by to_char(reg_date, 'YYYY-MM-DD') " +  
        " having to_char(reg_date, 'YYYY-MM-DD') <= ? " +  
        " ) o " +  
        " where rownum <= 10 " +  
        " order by o.days asc";  
  
    Connection conn = DBConnection.makeConnection();  
    PreparedStatement psmt = null;  
    ResultSet rs = null;  
  
    List<GraphDayDto> list = new ArrayList<>();  
  
    Calendar cal = Calendar.getInstance();  
  
    SimpleDateFormat sdf = new SimpleDateFormat("YYYY-MM-DD");  
  
    String today = sdf.format(cal.getTime());  
  
    try {  
        psmt = conn.prepareStatement(sql);  
        psmt.setString(1, today);  
  
        rs = psmt.executeQuery();  
  
        while(rs.next()) {  
            GraphDayDto dto = new GraphDayDto();  
            dto.count = rs.getInt("counts");  
            dto.date = rs.getString("days");  
  
            list.add(dto);  
        }  
    } catch (SQLException e) {  
        // TODO Auto-generated catch block  
        e.printStackTrace();  
    } finally {  
        DBClose.close(psmt, conn, rs);  
    }  
  
    return list;  
}
```

3. JDBC를 사용하여 오늘을 기준으로 최근 10일의 게시물 갯수를 리턴한다.

4. 리턴한 Gson String을 Json으로 파싱하고,
차트를 비동기로 다시 그린다.