

# Deng Admin 개발 기능 (상세)

## - 메인 페이지



1. 페이지를 로드할 때, ajax 를 호출한다.

### Controller

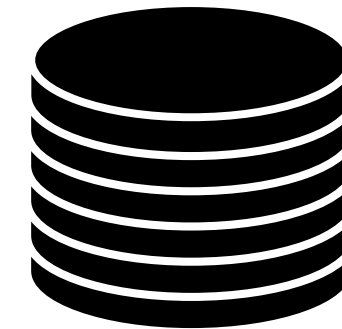
```
} else if(command.equals("drowInfo")) {  
    AdminService adminService = AdminService.getInstance();  
    HashMap<String, Integer> tablesCount = adminService.getTablesCount();  
    String json = new Gson().toJson(tablesCount);  
    resp.getWriter().write(json);  
}
```

2. Gson 을 사용한 Json String으로 리턴한다.

### Service

```
public HashMap<String, Integer> getTablesCount() {  
    HashMap<String, Integer> returnMap = new HashMap<>();  
    returnMap.put("animalCount", adminDao.getAnimalCount());  
    returnMap.put("afterCount", adminDao.getAfterCount());  
    returnMap.put("userCount", adminDao.getUserCount());  
    returnMap.put("commuCount", adminDao.getCommuCount());  
    returnMap.put("completeCount", adminDao.getCompleteCount());  
    return returnMap;  
}
```

3. Service 에서 각 테이블별 count 수를 리턴 받아 HashMap 으로 저장한뒤 리턴한다.



DB

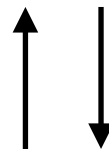
DAO

# 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.