

=====

## Mini Projects Development

=====

=> We are learning mini projects development to learn below concepts

- 1) Improving Requirement Analysis
- 2) How to ask queries on requirement
- 3) Database Design for the project (Data Modeling)
- 4) Java Classes & Methods Design
- 5) Coding & Testing
- 6) Code Integration

=====

## Mini & Major Projects Development Process

=====

- Step-1) Trainer will explain requirement & will share Requirement Document with you.
- Step-2) Students should analyze the requirement & should ask questions if any.
- Step-3) Students should work on database design (data modeling).
- Step-4) Show your database analysis to trainer for checking.
- Step-5) Trainer will explain database design (You compare your db design with Trainer db design to find any gaps).
- Step-6) Students should work on Java Classes & Methods Analysis.
- Step-7) Trainer will explain analysis for Java Classes & Methods.
- Step-8) Students should work on coding part.
- Step-9) Trainer will develop the code by involving few students (team work).
- Step-10) Code Integration.

=====

## 01-Mini Project : Training Institute Counsellors Portal

=====

- > Counsellor is responsible to collect students enquiries information.
- > Counsellor will enter student enquiry details in website (It is Open Enquiry).
- > Counsellor can check enquiries to follow up based on course/status.
- > Counsellor will update enquiry status (Enrolled/Lost).
- > Counsellor can check performance report in dashboard
  - total enquires handled
  - Open Enquiries Count
  - Enrolled Enquiries Count
  - Lost Enquiries count

=====

## Counsellors Portal Screens

=====

- 1) Registration Page
- 2) Login Page
- 3) Dashboard page
- 4) Add Enquiry
- 5) View Enquiries (with filter)
- 6) Edit and Update Enquiry

=====

## Assignment

=====

- 1) Analyse requirement & ask questions
- 2) Database Design (Data Model) (write sql queries to create required tables)
- 3) Java Classes & Methods Design

=====

## Database Design

=====

=> We need to below 2 tables

### 1) counsellors\_tbl

counsellor\_id (PK)  
name  
email (unique)  
pwd  
phno  
created\_date  
updated\_date

### 2) enquiries\_tbl

enq\_id (PK)  
student\_name  
student\_phno  
course\_name  
class\_mode  
enq\_status  
created\_date  
updated\_date  
  
counsellor\_id (FK)

=====

## Java Components

=====

- 1) Counsellor.java (Entity & DTO)
- 2) Enquiry.java (Entity & DTO)
- 3) CounsellorRepo.java (I)

- 4) EnquiryRepo.java (I)
- 5) DashboardResponse.java
- 6) ViewEnqFilterRequest.java
- 7) CounsellorService.java (I)
- 8) CounsellorServiceImpl.java
- 9) EnquiryService.java (I)
- 10) EnquiryServiceImpl.java
- 11) CounsellorController.java
- 12) EnquiryController.java

=====  
View Pages  
=====

- 1) index.html
- 2) register.html
- 3) dashboard.html
- 4) add-enq.html
- 5) view-enqs.html

=====  
Assignment  
=====

- 1) Create Springboot project with required dependencies
- 2) Create entities + Repositories + request + response + services + service impl classes.
- 3) Implement service layer methods.

=====  
Development Process  
=====

- 1) Create springboot application with below dependencies

- a) web-starter
- b) thymeleaf
- c) data-jpa
- d) mysql
- e) devtools

- 2) configure datasource properties in application.properties file

#Data Source Properties  
spring.datasource.url=jdbc:mysql://localhost:3306/jrtp27  
spring.datasource.username=root  
spring.datasource.password=root

#ORM Properties  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.show-sql=true  
spring.jpa.properties.hibernate.format\_sql=true

- 3) Create Entities and Repository interfaces.
- 4) Create DTO classes

5) Create Service interfaces & impl classes with required methods.

```
-----  
    public boolean register(Counsellor counsellor);  
    public Counsellor login(String email, String pwd);  
    public DashboardResponse getDashboardInfo(Integer counsellorId);  
-----
```

```
    public boolean addEnquiry(Enquiry enq, Integer counsellorId);  
    public List<Enquiry> getAllEnquiries(Integer counsellorId);  
    public List<Enquiry> getEnquiresWithFilter(ViewEnqsFilterRequest filterReq, Integer  
counsellorId);  
    public Enquiry getEnquiryById(Integer enqId);  
-----
```