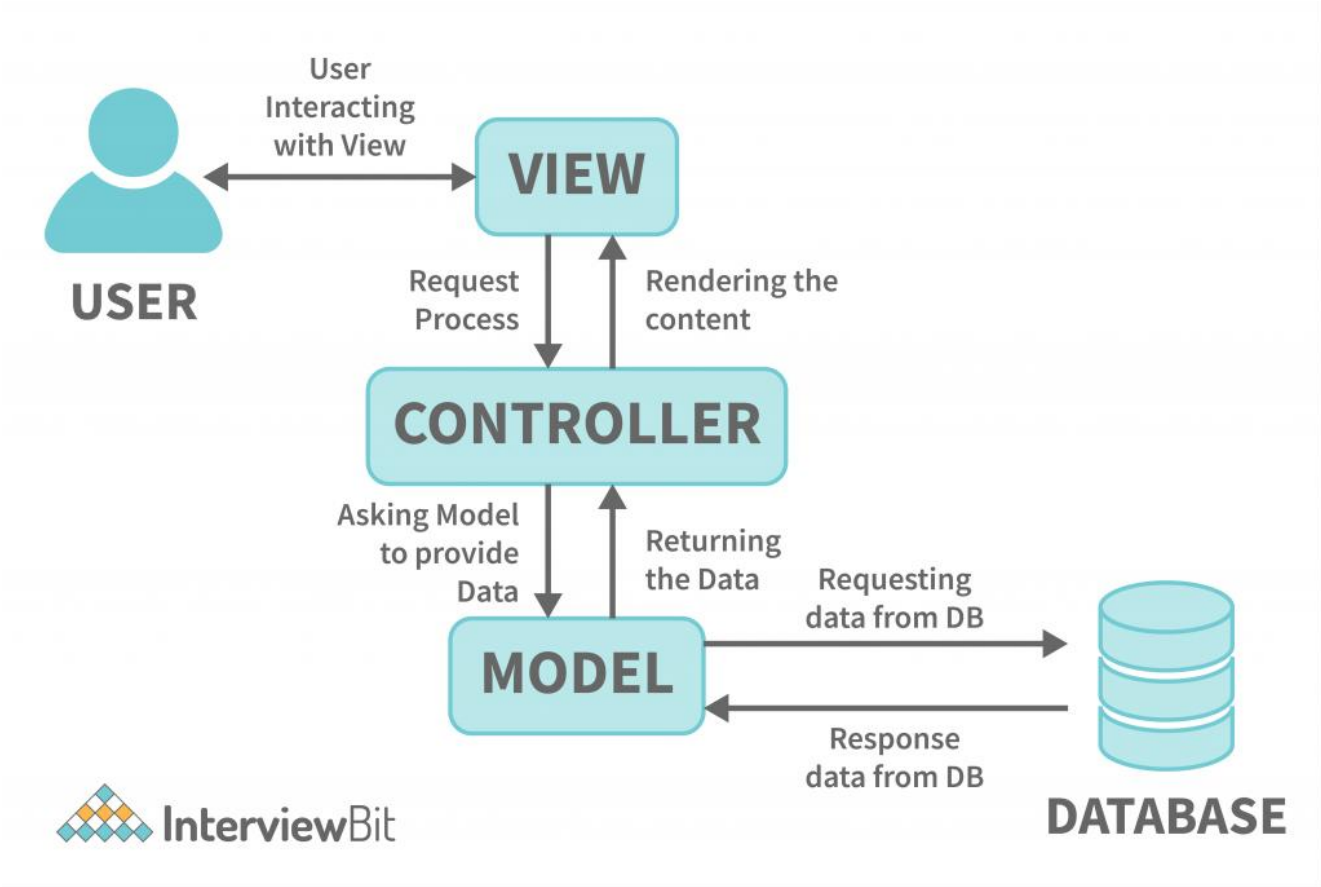
Prac 8) Write a program for implementing concept of MVC Architecture.

The MVC design pattern is a software architecture pattern that separates an application into three main components: Model, View, and Controller, making it easier to manage and maintain the codebase. It also allows for the reusability of components and promotes a more modular approach to software development.

What is the MVC Design Pattern?

The Model View Controller (MVC) design pattern specifies that an application consists of a data model, presentation information, and control information. The pattern requires that each of these be separated into different objects.

- The MVC pattern separates the concerns of an application into three distinct components, each responsible for a specific aspect of the application's functionality.
- This separation of concerns makes the application easier to maintain and extend, as changes to one component do not require changes to the other components.



1. Model

The Model component in the MVC (Model-View-Controller) design pattern demonstrates the data and business logic of an application. It is responsible for managing the application's data, processing business rules, and responding to requests for information from other components, such as the View and the Controller.

2. View

Displays the data from the Model to the user and sends user inputs to the Controller. It is passive and does not directly interact with the Model. Instead, it receives data from the Model and sends user inputs to the Controller for processing.

3. Controller

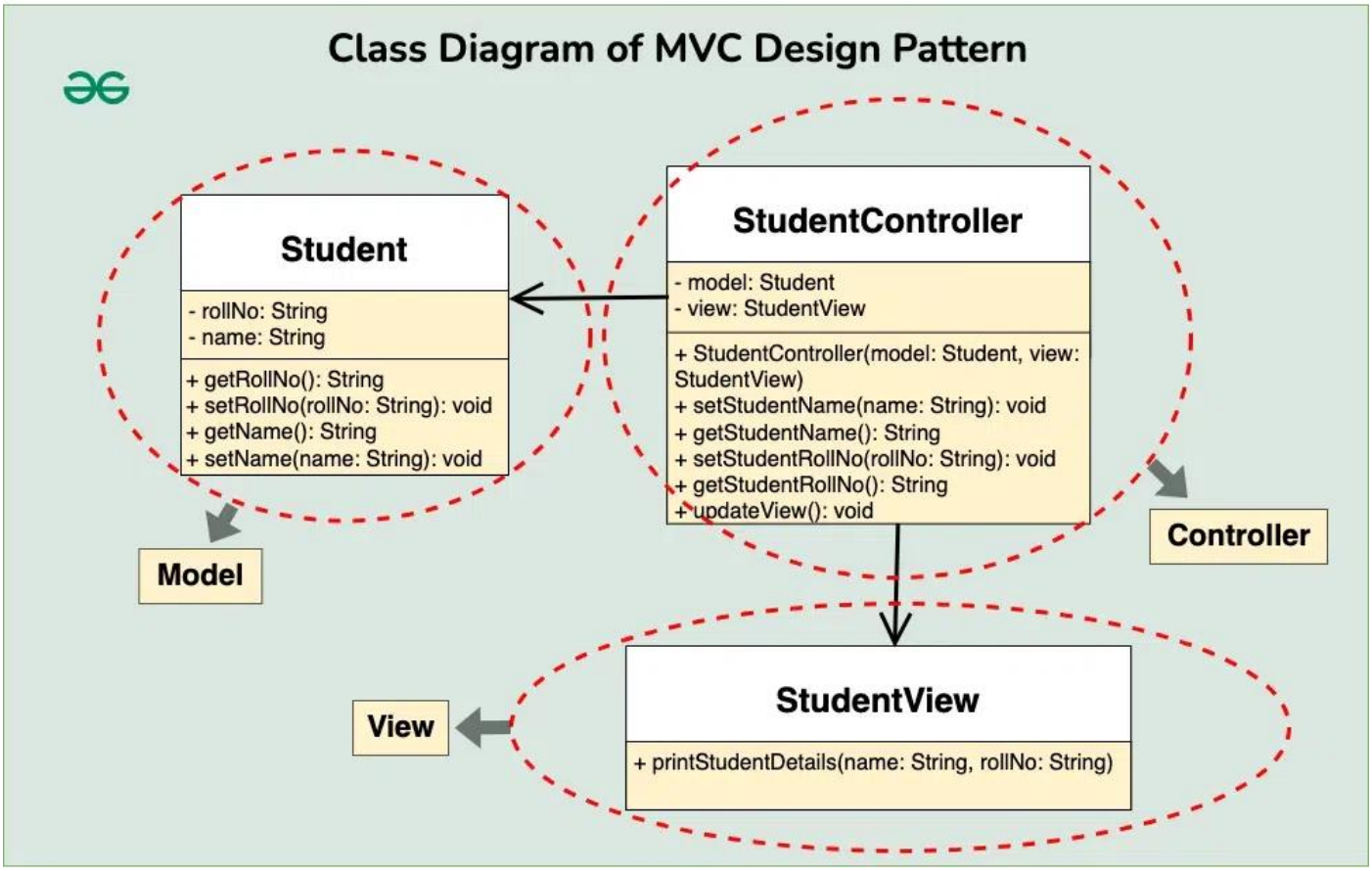
Controller acts as an intermediary between the Model and the View. It handles user input and updates the Model accordingly and updates the View to reflect changes in the Model. It contains application logic, such as input validation and data transformation.

Communication between the Components

1. **User Interaction with View:** The user interacts with the View, such as clicking a button or entering text into a form.
2. **View Receives User Input**: The View receives the user input and forwards it to the Controller.
3. **Controller Processes User Input**: The Controller receives the user input from the View. It interprets the input, performs any necessary operations (such as updating the Model), and decides how to respond.
4. **Controller Updates Model:** The Controller updates the Model based on the user input or application logic.
5. **Model Notifies View of Changes**: If the Model changes, it notifies the View.
6. **View Requests Data from Model:** The View requests data from the Model to update its display.
7. **Controller Updates View**: The Controller updates the View based on the changes in the Model or in response to user input.

8. **View Renders Updated UI**: The View renders the updated UI based on the changes made by the Controller.

Example of the MVC Design Pattern



Project Directory Structure

Quizapp
| --- Src/Main
|--------- Java
| |----------Model
| | |--------- Question.java
| | |--------- QuestionDAO.java
| |----------Controller
| | |-------- Quizservlet.java
Webapp
|--------- meta-INF
|--------- web-INF
|--------- lib
|--------- web.xml
|---------(All JSP pages i.e View Pages)