User Registration

To implement secure user registration with password encoding, we need to create a proper controller-service architecture. This will handle the HTTP requests and save user data to the database.

👉 Creating the UserController

- Create a controller class called UserController
- Annotate it with @RestController to handle HTTP requests
- Define a registration endpoint with @PostMapping("register")
- Accept user data in JSON format using @RequestBody
- Inject the UserService into the UserController using @Autowired
- Call the service method from the controller to process the registration

Example:

```
package com.telusko.springsecdemo.controller;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RestController;
import com.telusko.springsecdemo.model.User;
import com.telusko.springsecdemo.service.UserService;

@RestController
public class UserController {
    @Autowired
    private UserService service;

    @PostMapping("register")
    public User register(@RequestBody User user) {
      return service.saveUser(user);
    }
}
```

👉 Setting Up the UserService

- Create a service class called UserService
- Annotate it with @Service to mark it as a Spring service component
- Implement a saveUser() method that takes a User object and returns the saved User
- Inject the UserRepo repository to interact with the database

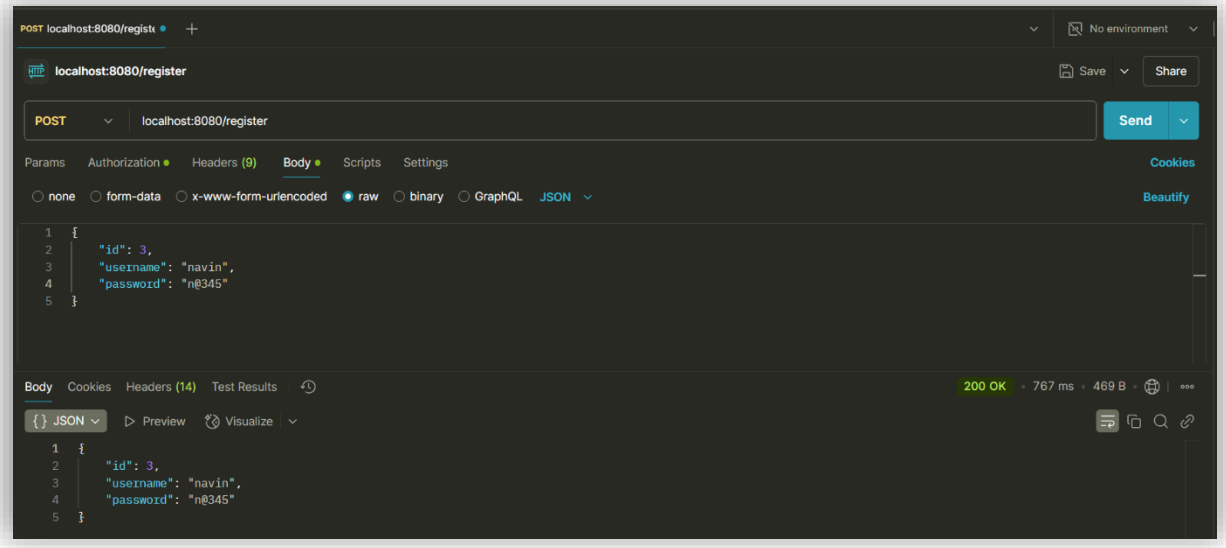
Example:

```java
package com.telusko.springsecdemo.service;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import com.telusko.springsecdemo.dao.UserRepo;
import com.telusko.springsecdemo.model.User;

@Service
public class UserService {
    @Autowired
    private UserRepo repo;

    public User saveUser(User user) {
        return repo.save(user);
    }
}
```

Output:



Query | Query History

```
1  SELECT * FROM users
2
```

Data Output | Messages | Notifications

SQL

	id [PK] integer	username text	password text
1	1	kiran	n@789
2	2	harsh	h@123
3	3	navin	n@345