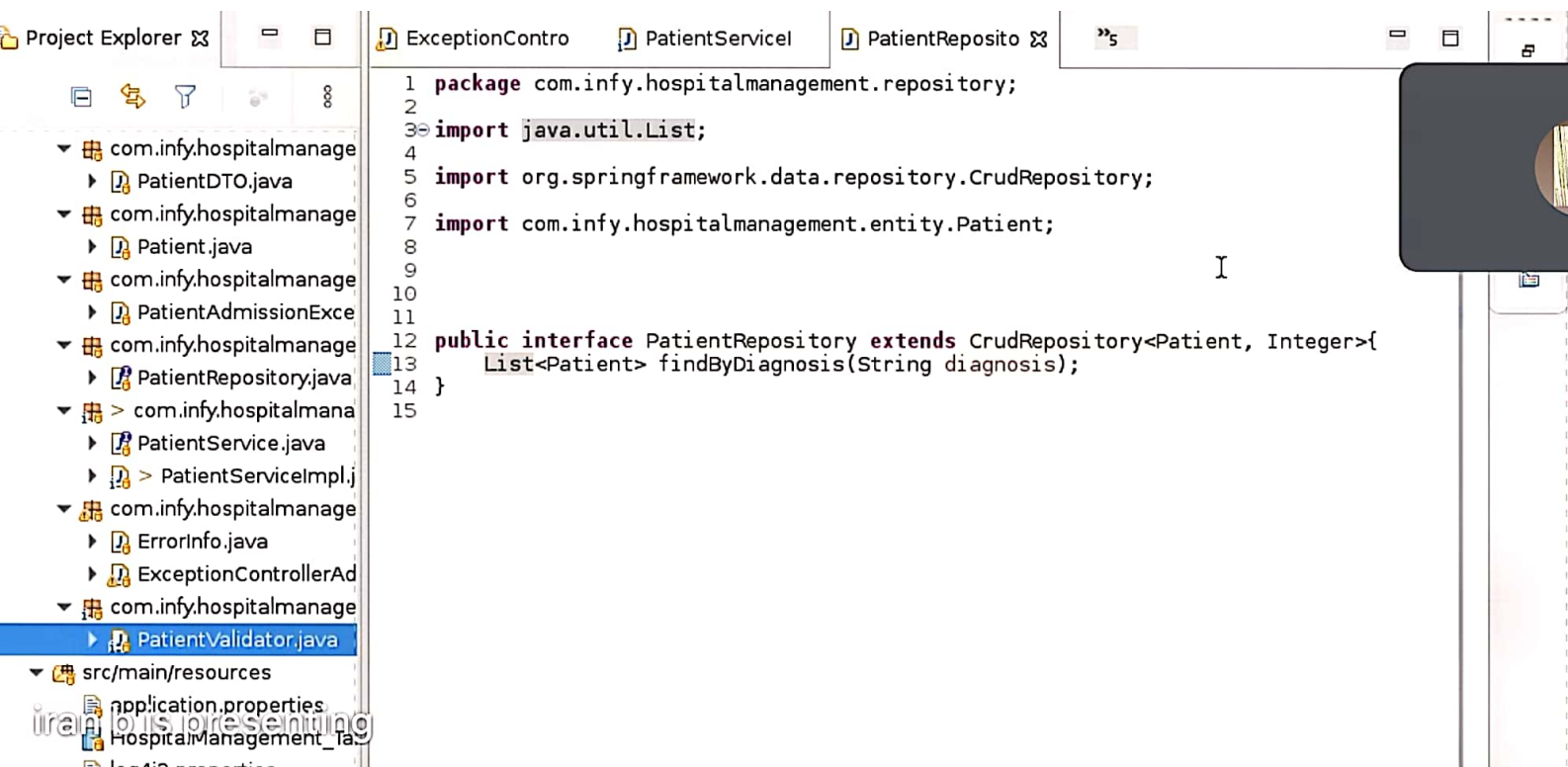


com.infy.hospitalmanage
PatientDTO.java
com.infy.hospitalmanage
Patient.java
com.infy.hospitalmanage
PatientAdmissionExce
com.infy.hospitalmanage
PatientRepository.java
com.infy.hospitalmanage
PatientService.java
PatientServiceImpl.jav
com.infy.hospitalmanage
ErrorInfo.java
ExceptionHandlerAd
com.infy.hospitalmanage
PatientValidator.java
main/resources
application.properties
HospitalManagement_Tal
og4j2.properties
validationMessages.pron
test/java

```

9  import org.springframework.beans.factory.annotation.Autowired;
10 import org.springframework.http.HttpStatus;
11 import org.springframework.http.ResponseEntity;
12 import org.springframework.validation.annotation.Validated;
13 import org.springframework.web.bind.annotation.GetMapping;
14 import org.springframework.web.bind.annotation.PathVariable;
15 import org.springframework.web.bind.annotation.PostMapping;
16 import org.springframework.web.bind.annotation.RequestBody;
17 import org.springframework.web.bind.annotation.RequestMapping;
18 import org.springframework.web.bind.annotation.RestController;
19
20 import com.infy.hospitalmanagement.dto.PatientDTO;
21 import com.infy.hospitalmanagement.exception.PatientAdmissionException;
22 import com.infy.hospitalmanagement.service.PatientService;
23 @RestController
24 @RequestMapping(value = "api")
25 @Validated
26 public class PatientAPI {
27     @Autowired
28     private PatientService patientService;
29     @GetMapping(value="patients/{diagnosis}")
30     public ResponseEntity<List<PatientDTO>> getPatientsByDiagnosis(@NotNull(message=
31     @Pattern(regexp = "[A-Za-z0-9 ]+", message="{patient.diagnosis.invalid}")
32     @PathVariable String diagnosis)
33         throws PatientAdmissionException {
34         List<PatientDTO> dto= patientService.getListOfPatients(diagnosis);
35         return new ResponseEntity<>(dto,HttpStatus.OK);
36     }
37     @PostMapping(value="patients")
38     public ResponseEntity<PatientDTO> admitPatient(@RequestBody @Valid PatientDTO |
39         throws PatientAdmissionException {
40         PatientDTO dto= patientService.admitPatient(patientDTO);
41         return new ResponseEntity<>(dto,HttpStatus.CREATED);
42     }

```



- ▼ com.infy.hospitalmanage
 - PatientDTO.java
- ▼ com.infy.hospitalmanage
 - Patient.java
- ▼ com.infy.hospitalmanage
 - PatientAdmissionExce
- ▼ com.infy.hospitalmanage
 - PatientRepository.java
- ▼ com.infy.hospitalmana
 - PatientService.java
 - PatientServiceImpl.j
- ▼ com.infy.hospitalmanage
 - ErrorInfo.java
 - ExceptionControllerAd
- ▼ com.infy.hospitalmanage
 - PatientValidator.java
- ▼ src/main/resources
 - application.properties
 - HospitalManagement_Tal
 - log4j2.properties
 - ValidationMessages.prop
- ▼ src/test/java
- ▼ JRE System Library [JavaSE-

```

9
10 import com.infy.hospitalmanagement.dto.PatientDTO;
11 import com.infy.hospitalmanagement.entity.Patient;
12 import com.infy.hospitalmanagement.exception.PatientAdmissionException;
13 import com.infy.hospitalmanagement.repository.PatientRepository;
14 import com.infy.hospitalmanagement.validator.PatientValidator;
15 @Service(value = "patientService")
16 @Transactional
17 public class PatientServiceImpl implements PatientService{
18     @Autowired
19     private PatientRepository patientRepository;
20
21     @Override
22     public List<PatientDTO> getListOfPatients(String diagnosis) throws PatientAdmis
23         List<Patient> list=patientRepository.findByDiagnosis(diagnosis);
24         if(list.isEmpty()) {
25             throw new PatientAdmissionException("PatientService.PATIENT_UNAVAILABLE
26         }
27         List<PatientDTO> dtoList= new ArrayList<PatientDTO>();
28         for (Patient patient : list) {
29             //PatientDTO.prepareDTO(patient);
30             dtoList.add(PatientDTO.prepareDTO(patient));
31         }
32         return dtoList;
33     }
34
35     @Override
36     public PatientDTO admitPatient(PatientDTO patientDTO) throws PatientAdmissionEx
37         PatientValidator.validatePatient(patientDTO);
38         // Patient p =new Patient();
39         Patient p=PatientDTO.prepareEntity(patientDTO);
40         patientRepository.save(p);
41         return patientDTO;

```

Writable

Smart Insert

41 : 17 : 1487

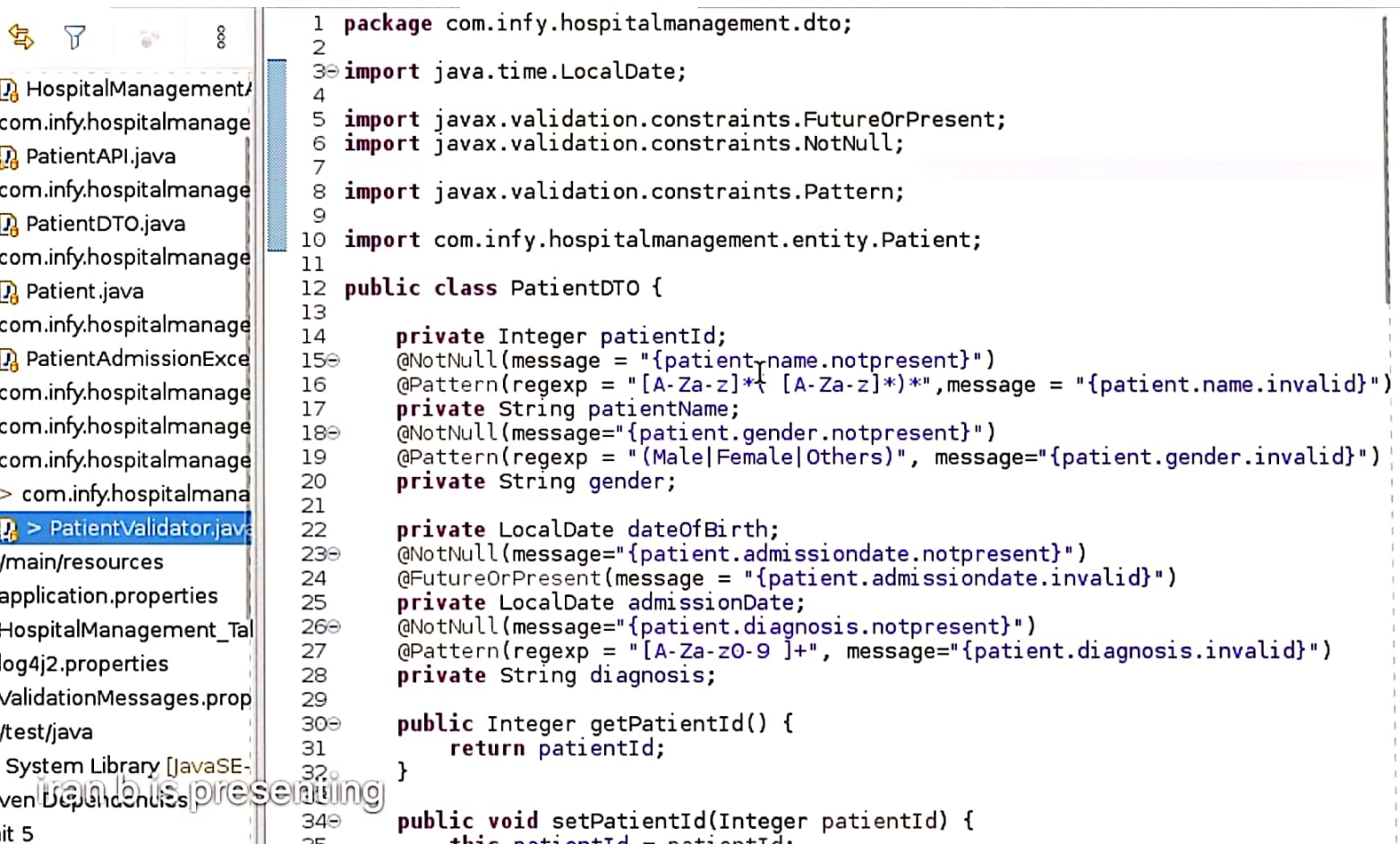
Project Explorer

- HospitalManagement/
- com.infy.hospitalmanage
 - PatientAPI.java
 - PatientDTO.java
 - Patient.java
 - PatientAdmissionExce
 - PatientValidator.java
- src/main/resources
 - application.properties
 - HospitalManagement_Tal
 - ValidationMessages.prop

iranbots presenting

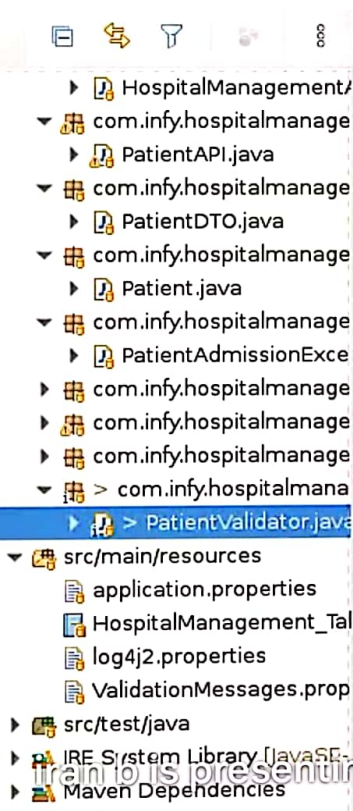
Patient.java PatientDTO.java ValidationMessages. PatientValidator.jav

```
5 import javax.persistence.Entity;
6 import javax.persistence.GeneratedValue;
7 import javax.persistence.GenerationType;
8 import javax.persistence.Id;
9 import javax.persistence.Table;
10 @Entity
11 @Table(name="patient")
12 public class Patient {
13     @Id
14     @GeneratedValue(strategy = GenerationType.IDENTITY)
15     private Integer patientId;
16     private String patientName;
17     private String gender;
18     private LocalDate dateOfBirth;
19     private LocalDate admissionDate;
20     private String diagnosis;
21
22     public Integer getPatientId() {
23         return patientId;
24     }
25
26     public void setPatientId(Integer patientId) {
27         this.patientId = patientId;
28     }
29
30     public String getPatientName() {
31         return patientName;
32     }
33
34     public void setPatientName(String patientName) {
```

The screenshot shows an IDE with a project explorer on the left and a code editor on the right. The project explorer lists files like HospitalManagementAPI.java, PatientAPI.java, PatientDTO.java, Patient.java, PatientAdmissionException.java, PatientValidator.java, application.properties, HospitalManagement_Tailor.java, log4j2.properties, ValidationMessages.properties, test/java, System Library [JavaSE-8], and Maven Dependencies. The code editor displays the PatientDTO.java file with the following content:

```
1 package com.infy.hospitalmanagement.dto;
2
3 import java.time.LocalDate;
4
5 import javax.validation.constraints.FutureOrPresent;
6 import javax.validation.constraints.NotNull;
7
8 import javax.validation.constraints.Pattern;
9
10 import com.infy.hospitalmanagement.entity.Patient;
11
12 public class PatientDTO {
13
14     private Integer patientId;
15     @NotNull(message = "{patient.name.notpresent}")
16     @Pattern(regexp = "[A-Za-z]*[A-Z-a-z]*", message = "{patient.name.invalid}")
17     private String patientName;
18     @NotNull(message = "{patient.gender.notpresent}")
19     @Pattern(regexp = "(Male|Female|Others)", message = "{patient.gender.invalid}")
20     private String gender;
21
22     private LocalDate dateOfBirth;
23     @NotNull(message = "{patient.admissiondate.notpresent}")
24     @FutureOrPresent(message = "{patient.admissiondate.invalid}")
25     private LocalDate admissionDate;
26     @NotNull(message = "{patient.diagnosis.notpresent}")
27     @Pattern(regexp = "[A-Za-z0-9 ]+", message = "{patient.diagnosis.invalid}")
28     private String diagnosis;
29
30     public Integer getPatientId() {
31         return patientId;
32     }
33
34     public void setPatientId(Integer patientId) {
35         this.patientId = patientId;
36     }
37 }
```



```
1 package com.infy.hospitalmanagement.dto;
2
3 import java.time.LocalDate;
4
5 import javax.validation.constraints.FutureOrPresent;
6 import javax.validation.constraints.NotNull;
7
8 import javax.validation.constraints.Pattern;
9
10 import com.infy.hospitalmanagement.entity.Patient;
11
12 public class PatientDTO {
13
14     private Integer patientId;
15     @NotNull(message = "{patient.name.notpresent}")
16     @Pattern(regexp = "[A-Za-z]*([A-Za-z]*)*", message = "{patient.name.invalid}")
17     private String patientName;
18     @NotNull(message = "{patient.gender.notpresent}")
19     @Pattern(regexp = "(Male|Female|Others)", message = "{patient.gender.invalid}")
20     private String gender;
21
22     private LocalDate dateOfBirth;
23     @NotNull(message = "{patient.admissiondate.notpresent}")
24     @FutureOrPresent(message = "{patient.admissiondate.invalid}")
25     private LocalDate admissionDate;
26     @NotNull(message = "{patient.diagnosis.notpresent}")
27     @Pattern(regexp = "[A-Za-z0-9 ]+", message = "{patient.diagnosis.invalid}")
28     private String diagnosis;
29
30     public Integer getPatientId() {
31         return patientId;
32     }
33
34     public void setPatientId(Integer patientId) {
```

Project Explorer

- HospitalManagement/
- com.infy.hospitalmanage
 - PatientAPI.java
 - PatientDTO.java
 - Patient.java
 - PatientAdmissionExce
- src/main/resources
 - application.properties
 - HospitalManagement_Tal
 - log4j2.properties
 - ValidationMessages.prop
- src/test/java
- JRE System Library [JavaSE-
- Maven Dependencies
- JUnit 5

Open files: Patient.java, PatientDTO.java, ValidationMessages., PatientValidator.jav

```
1 package com.infy.hospitalmanagement.validator;
2
3 import java.time.LocalDate;
4
5 public class PatientValidator {
6
7     public static void validatePatient(PatientDTO patientDTO) throws PatientAdmiss
8     {
9         if(isValidDateOfBirth(patientDTO.getDateOfBirth())==false) {
10             throw new PatientAdmissionException("PatientValidator.INVALID_DOB");
11         }
12     }
13
14     public static Boolean isValidDateOfBirth(LocalDate dateOfBirth) {
15         LocalDate today=LocalDate.now();
16         LocalDate future=today.minusYears(100);
17         if(dateOfBirth.isAfter(today)||!(dateOfBirth.isBefore(future))) {
18             return false;
19         }
20         return true;
21     }
22 }
23
24
25
26
27
28
```