**JAVA for Exam**

**1.След като съм разархивирал папката отварям проекта с Intelij.**

**2.Стартирам XAMPP+MySQL**

**3.Опитвам се да билдна проекта,ако дава грешка отивам в src/main/java/ime na proekta/entity и отварям файла,който се намира там.Записвам стойностите на колоните които ще ми съдържа базата данни и после си правя Getters and Setters(Alt+Insert).**

**@Id  
@GeneratedValue(strategy = GenerationType.*IDENTITY*)  
private int id;  
  
@Column(nullable = false)  
private String name;  
  
@Column(nullable = false)  
private String genre;  
  
@Column(nullable = false)  
private String director;  
  
@Column(nullable = false)  
private int year;**

**4.Връщам се в главния контролер и задавам всички функции да имат return.На първият ред дописвам следните редове:**

**private final FilmRepository filmRepository;  
  
@Autowired  
public FilmController(FilmRepository filmRepository) {  
   
 this.filmRepository=filmRepository;  
  
}**

**5.Правя си метода за Index: и тествам дали работи**

**@GetMapping("/")  
public String index(Model model) {  
 List<Film> films = filmRepository.findAll();  
 model.addAttribute("films", films);  
 model.addAttribute("view", "film/index");  
 return "base-layout";  
}**

**6.Отварям HeidySQL и си търся базата данни(по име).Ръчно си създавам 1 акаунт и пак тествам дали работи приложението.**

**7.Отвам в src/main/java/ime na proekta/bindingmodel и отварям файла,който се намира там.Пиша същите неща както в entity,но с малка разлика:**

**@NotNull  
@Size(min=1)  
private String name;  
@NotNull  
@Size(min=1)  
private String genre;  
@NotNull  
@Size(min=1)  
private String director;**

**И добавям Getters and Setters(Alt+Insert).**

**8.Правя методите Get and Post за Create:**

@GetMapping(**"/create"**)  
**public** String create(Model model) {  
 model.addAttribute(**"film"**, **new** FilmBindingModel());  
 model.addAttribute(**"view"**, **"film/create"**);  
 **return "base-layout"**;  
}  
  
@PostMapping(**"/create"**)  
**public** String createProcess(Model model, @Valid FilmBindingModel filmBindingModel, BindingResult bindingResult) {  
  
 **if** (bindingResult.hasErrors()) {  
 model.addAttribute(**"message"**, **"Invalid data."**);  
 model.addAttribute(**"film"**, filmBindingModel);  
 model.addAttribute(**"view"**, **"film/create"**);  
 **return "base-layout"**;  
 }  
 Film film = **new** Film();  
 film.setName(filmBindingModel.getName());  
 film.setGenre(filmBindingModel.getGenre());  
 film.setDirector(filmBindingModel.getDirector());  
 film.setYear(filmBindingModel.getYear());  
 **filmRepository**.saveAndFlush(film);  
 **return "redirect:/"**;  
}

**!!! @Valid е задължителна в полето createProcess иначе няма да работи!**

**9.Правя методите за Delete:**

@GetMapping(**"/delete/{id}"**)  
**public** String delete(Model model, @PathVariable **int** id) {  
 Film film = **this**.**filmRepository**.findOne(id);  
 **if** (film == **null**)  
 **return "redirect:/"**;  
 model.addAttribute(**"film"**, film);  
 model.addAttribute(**"view"**, **"film/delete"**);  
 **return "base-layout"**;  
}  
@PostMapping(**"/delete/{id}"**)  
**public** String deleteProcess(@PathVariable **int** id) {  
 Film film = **this**.**filmRepository**.findOne(id);  
 **if** (film == **null**)  
 **return "redirect:/"**;  
 **filmRepository**.delete(film);  
 **filmRepository**.flush();  
 **return "redirect:/"**;  
}

**10.Правя методите за Edit:**

@GetMapping(**"/edit/{id}"**)  
**public** String edit(Model model, @PathVariable **int** id) {  
 Film film = **this**.**filmRepository**.findOne(id);  
 **if** (film == **null**)  
 **return "redirect:/"**;  
  
 model.addAttribute(**"film"**, film);  
 model.addAttribute(**"view"**, **"film/edit"**);  
 **return "base-layout"**;  
  
}  
  
@PostMapping(**"/edit/{id}"**)  
**public** String editProcess(Model model, @PathVariable **int** id, @Valid FilmBindingModel filmBindingModel,BindingResult bindingResult) {  
  
 **if** (bindingResult.hasErrors()) {  
 model.addAttribute(**"message"**, **"Invalid data."**);  
 model.addAttribute(**"film"**, filmBindingModel);  
 model.addAttribute(**"view"**, **"film/edit"**);  
 **return "base-layout"**;  
 }  
  
 Film film = **this**.**filmRepository**.findOne(id);  
 **if** (film == **null**)  
 **return "redirect:/"**;  
  
 film.setName(filmBindingModel.getName());  
 film.setGenre(filmBindingModel.getGenre());  
 film.setDirector(filmBindingModel.getDirector());  
 film.setYear(filmBindingModel.getYear());  
 **filmRepository**.saveAndFlush(film);  
  
 **return "redirect:/"**;  
  
}

**!!!Във view/edit трябва да изтрия action(който се намира при метод POST) най-отгоре и да ми остане:**

<**h1**>Edit Film</**h1**>  
<**section**>  
 <**form method="POST"**>  
 <**div**>  
 <**label for="name"**>Name</**label**>  
 <**input type="text" id="name" th:value="${film.name}" name="name"** />  
 <**label for="genre"**>Genre</**label**>  
 <**input type="text" id="genre" th:value="${film.genre}" name="genre"** />  
 <**label for="director"**>Director</**label**>  
 <**input type="text" id="director" th:value="${film.director}" name="director"** />  
 <**label for="year"**>Year</**label**>  
 <**input type="text" id="year" th:value="${film.year}" name="year"** />  
 <**button class="accept" type="submit"**>Edit</**button**>  
 <**button type="button" th:onclick="'location.href = \'' + @{/} + '\''" class="cancel"**>Cancel</**button**>  
 </**div**>  
 </**form**>  
</**section**>