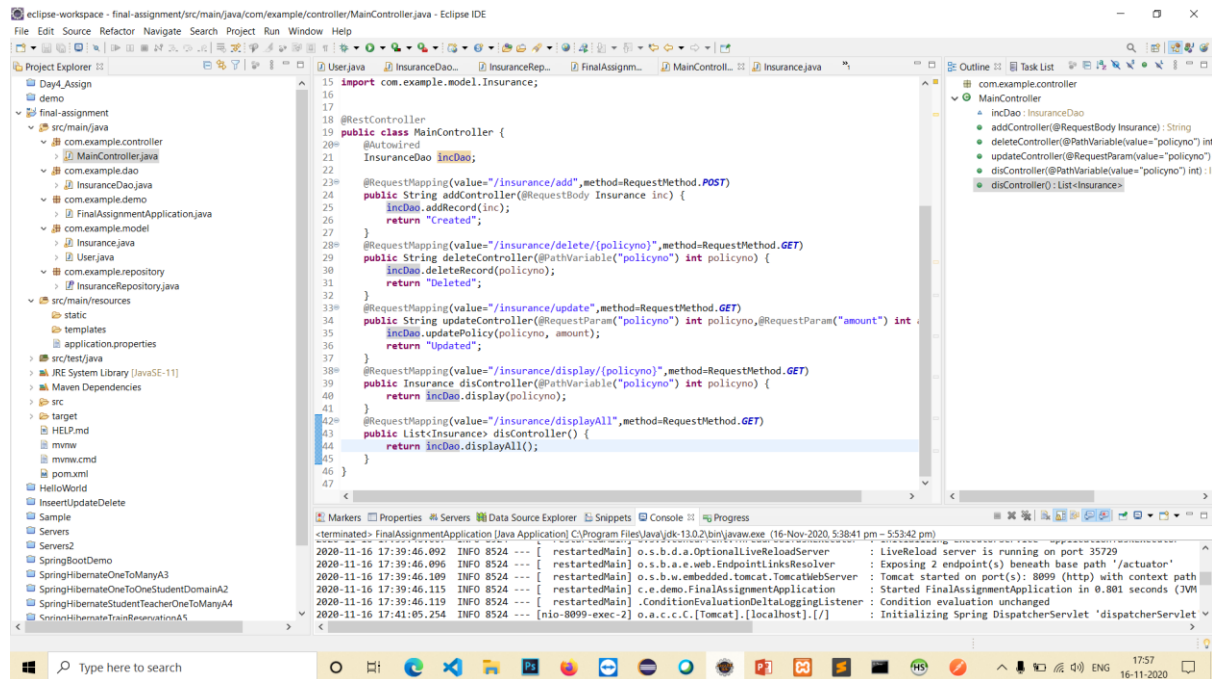


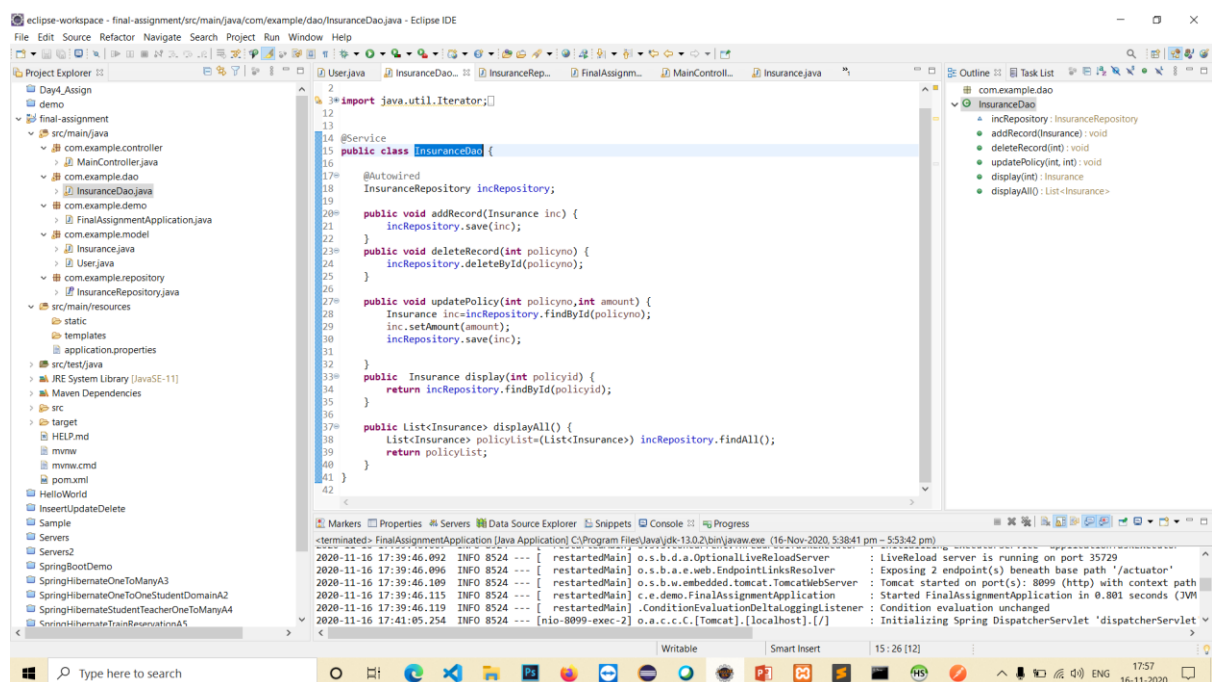
Final Assignment Task2: Write a spring boot application to perform CRUD operations on Insurance domain. Looking at the real time application, you can decide the fields. Insert, Update, Delete and display all functionality data should reflect in the database table.

Controller



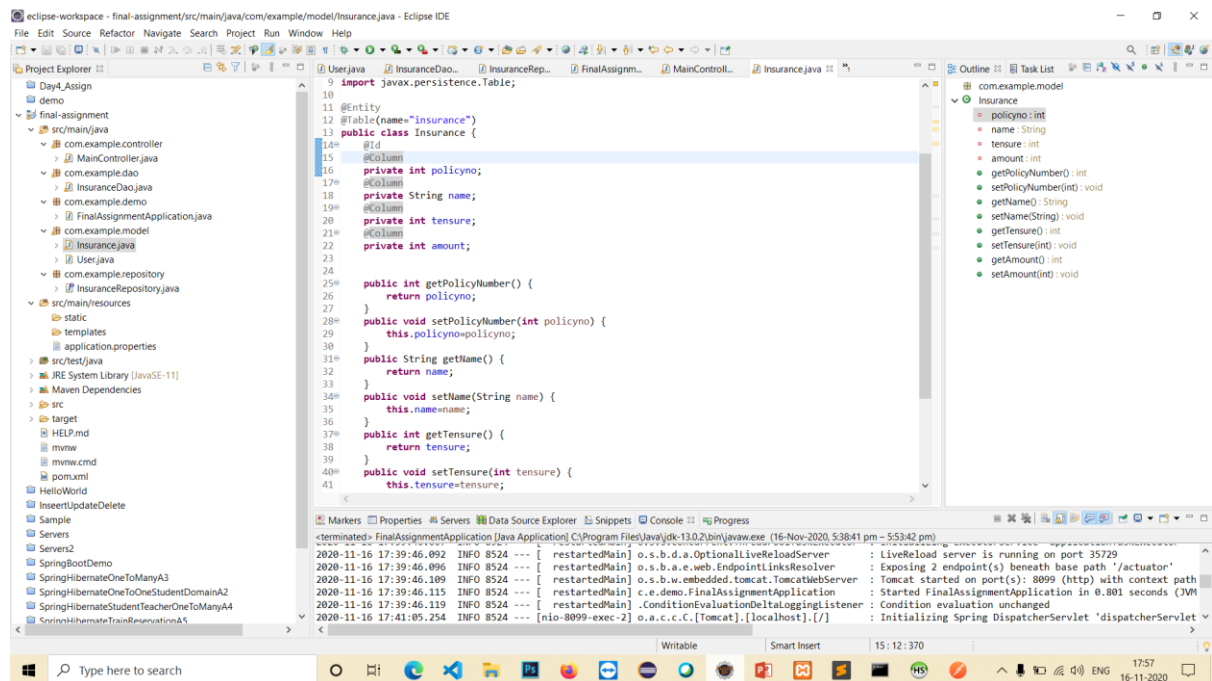
```
15 import com.example.model.Insurance;
16
17
18 @RestController
19 public class MainController {
20     @Autowired
21     InsuranceDao incDao;
22
23     @RequestMapping(value="/insurance/add",method=RequestMethod.POST)
24     public String addController(@RequestBody Insurance inc) {
25         incDao.addRecord(inc);
26         return "Created";
27     }
28
29     @RequestMapping(value="/insurance/delete/{policyno}",method=RequestMethod.GET)
30     public String deleteController(@PathVariable("policyno") int policyno) {
31         incDao.deleteRecord(policyno);
32         return "Deleted";
33     }
34
35     @RequestMapping(value="/insurance/update",method=RequestMethod.GET)
36     public String updateController(@RequestParam("policyno") int policyno,@RequestParam("amount") int amount) {
37         incDao.updatePolicy(policyno, amount);
38         return "Updated";
39     }
40
41     @RequestMapping(value="/insurance/display/{policyno}",method=RequestMethod.GET)
42     public Insurance disController(@PathVariable("policyno") int policyno) {
43         return incDao.display(policyno);
44     }
45
46     @RequestMapping(value="/insurance/displayAll",method=RequestMethod.GET)
47     public List<Insurance> disController() {
48         return incDao.displayAll();
49     }
50 }
```

Insurance Dao Service

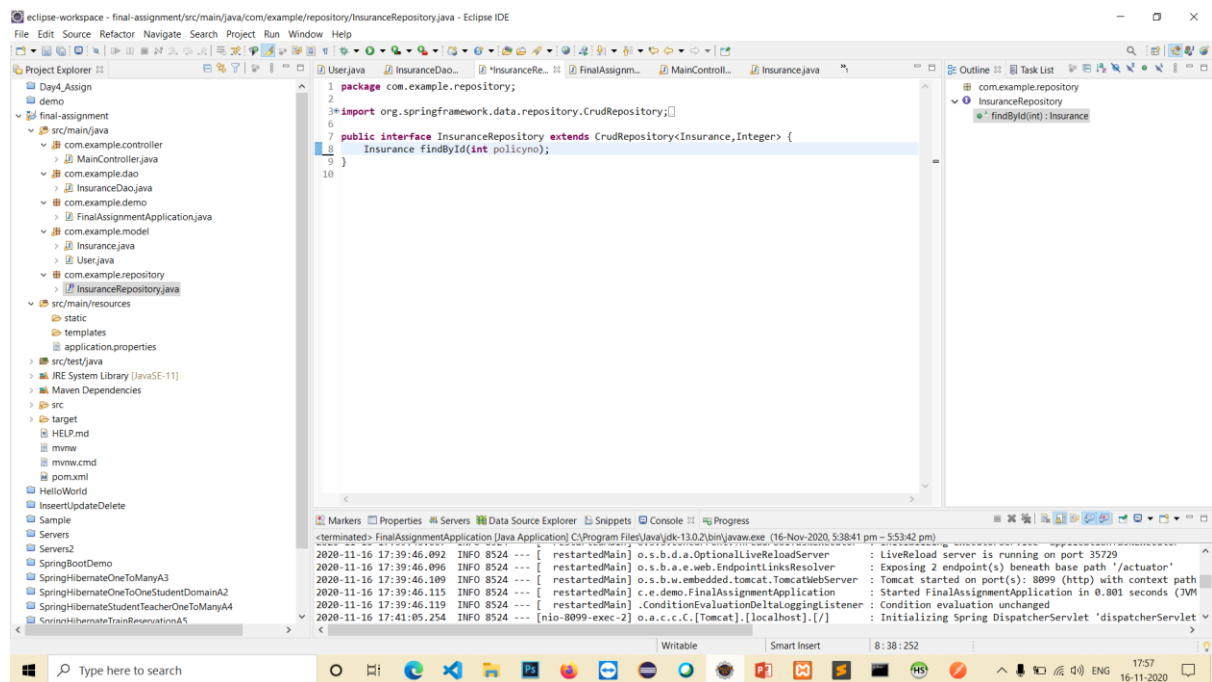


```
1 import java.util.Iterator;
2
3
4 @Service
5 public class InsuranceDao {
6
7     @Autowired
8     InsuranceRepository incRepository;
9
10     public void addRecord(Insurance inc) {
11         incRepository.save(inc);
12     }
13
14     public void deleteRecord(int policyno) {
15         incRepository.deleteByid(policyno);
16     }
17
18     public void updatePolicy(int policyno,int amount) {
19         Insurance inc=incRepository.findByid(policyno);
20         inc.setAmount(amount);
21         incRepository.save(inc);
22     }
23
24     public Insurance display(int policyid) {
25         return incRepository.findByid(policyid);
26     }
27
28     public List<Insurance> displayAll() {
29         List<Insurance> policyList=(List<Insurance>) incRepository.findAll();
30         return policyList;
31     }
32 }
```

Insurance Class

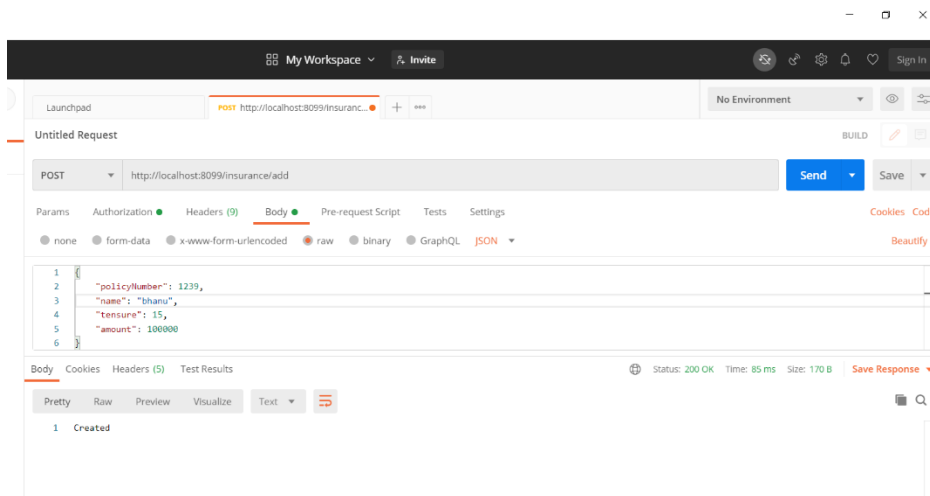


Insurance Repository

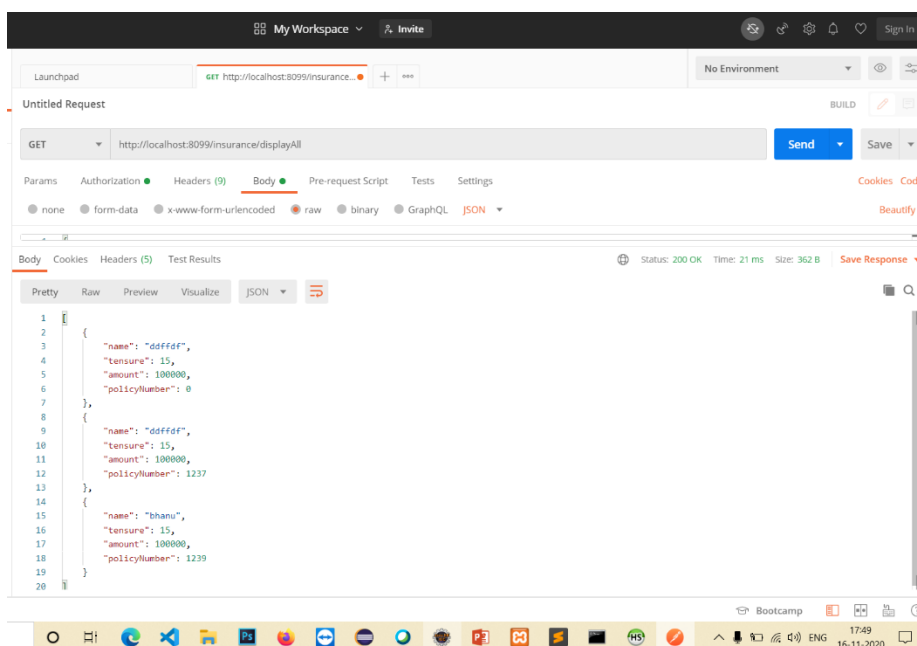


Results:

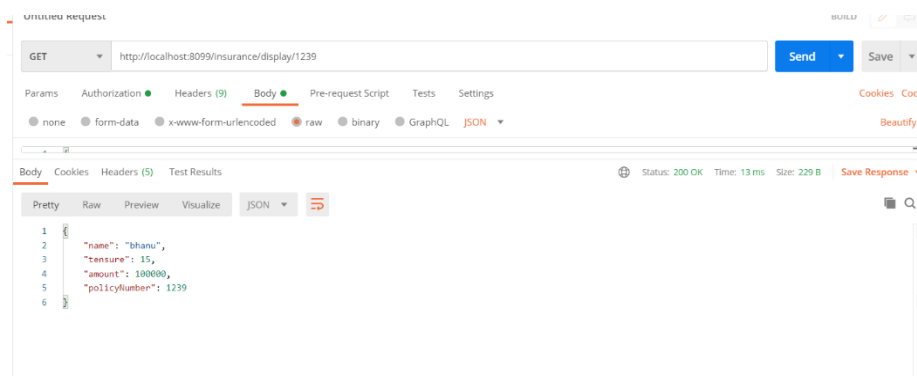
Create Policy: Sending Json Data using POST Method



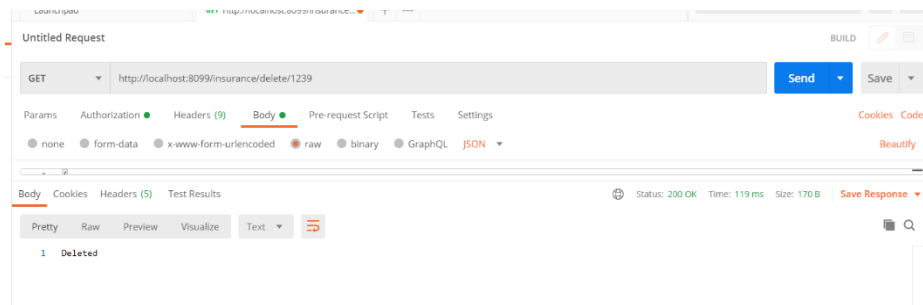
Displaying All Policies



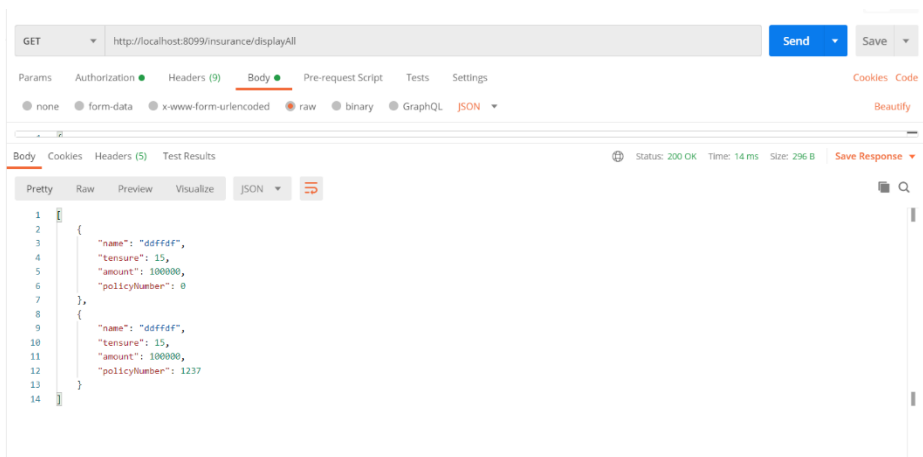
Displaying Single Policy: policy number send through URL



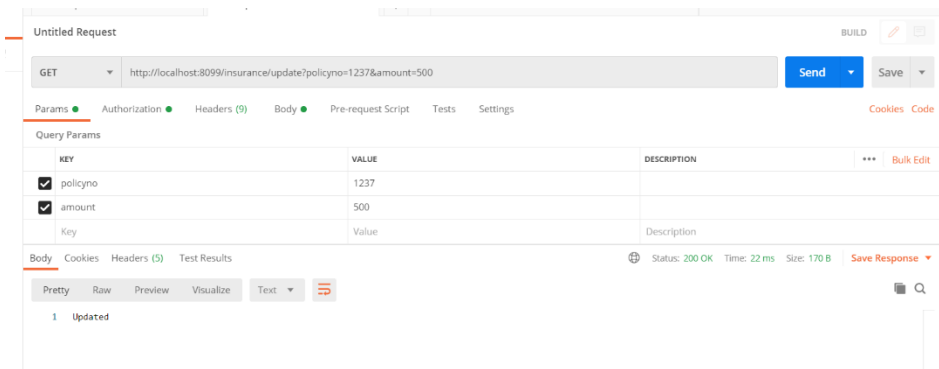
Delete Policy: policy number send through URL



Display All after Delete:



Update Policy:



Results after Update:

