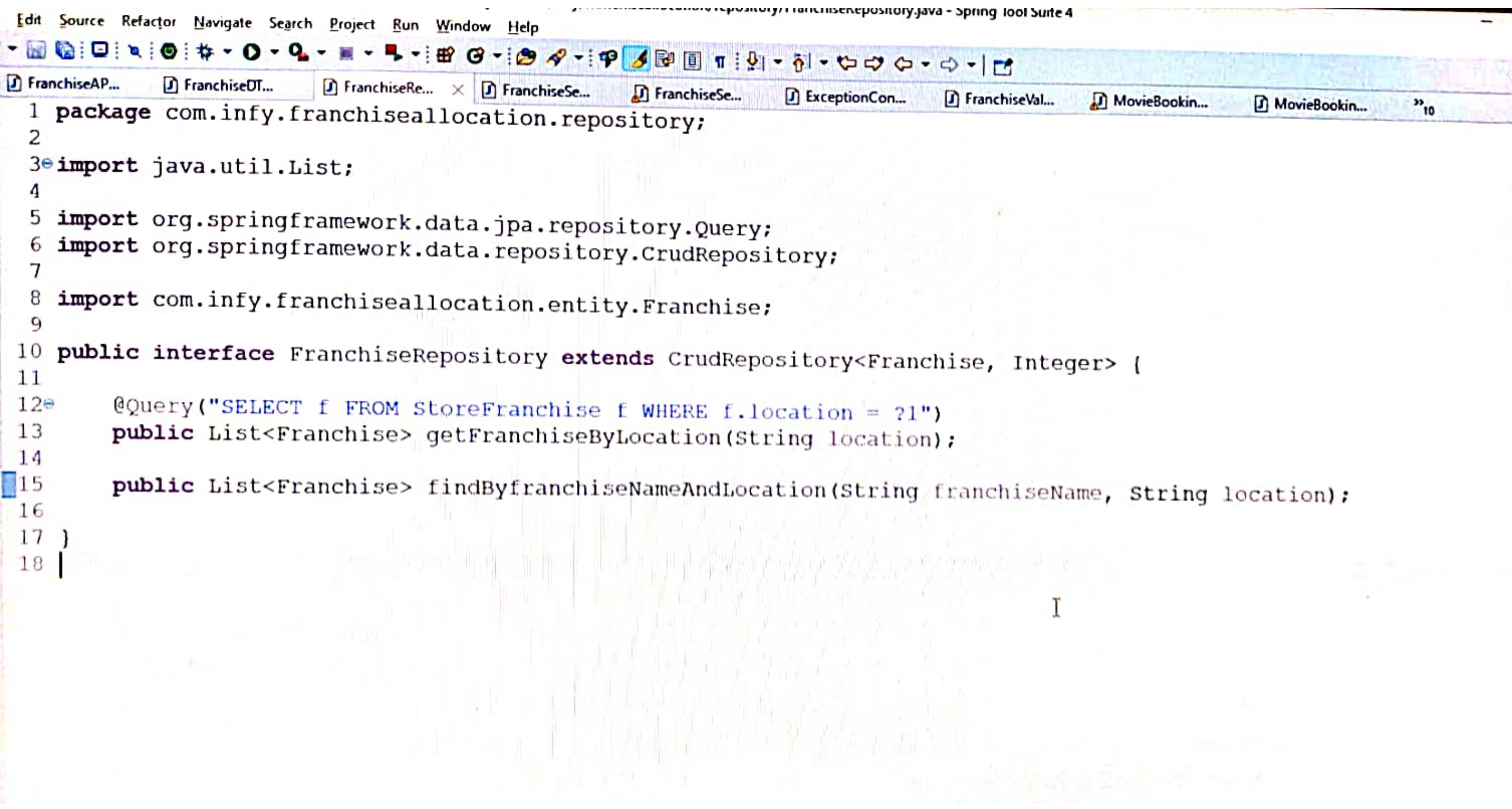
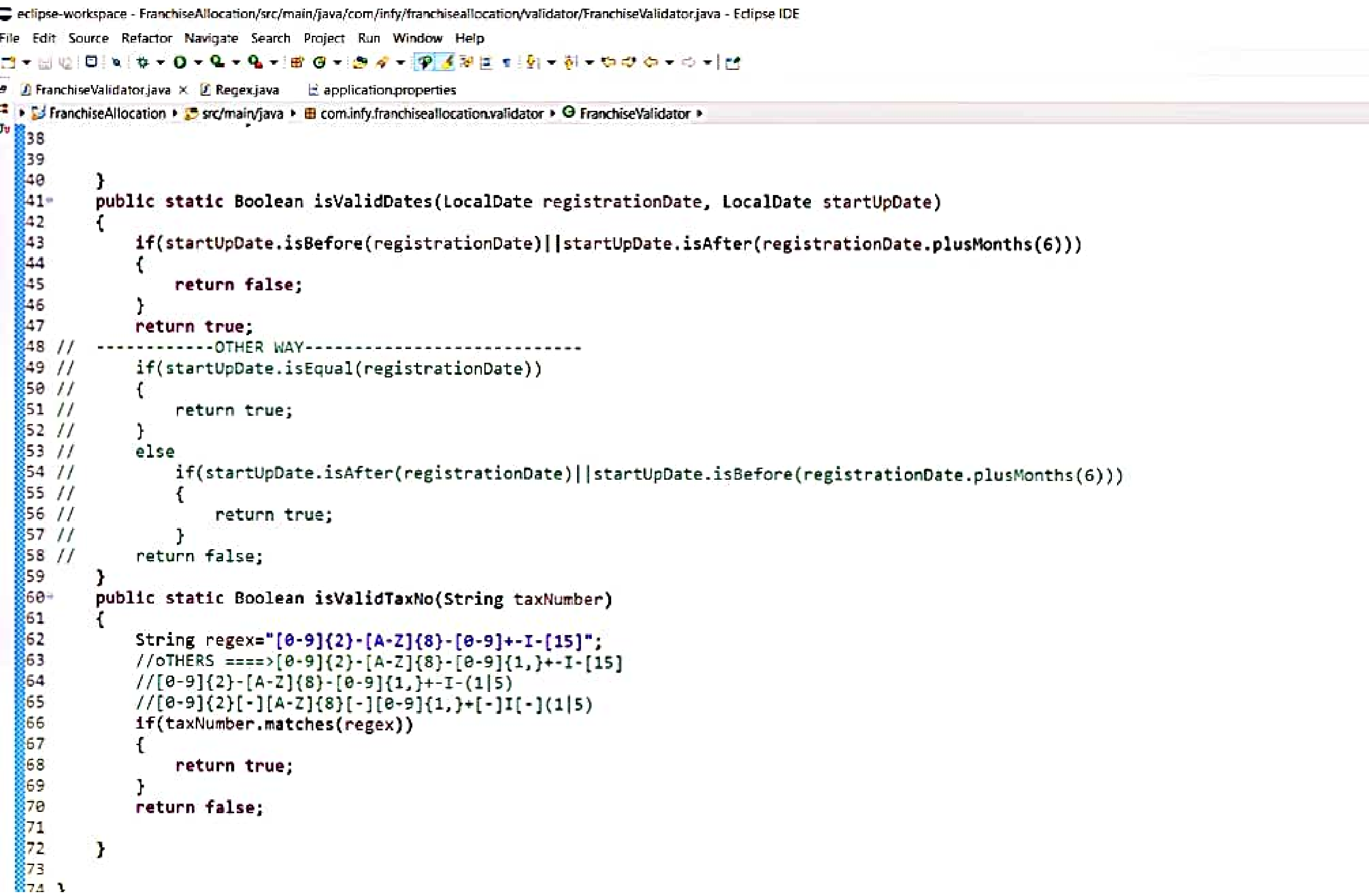
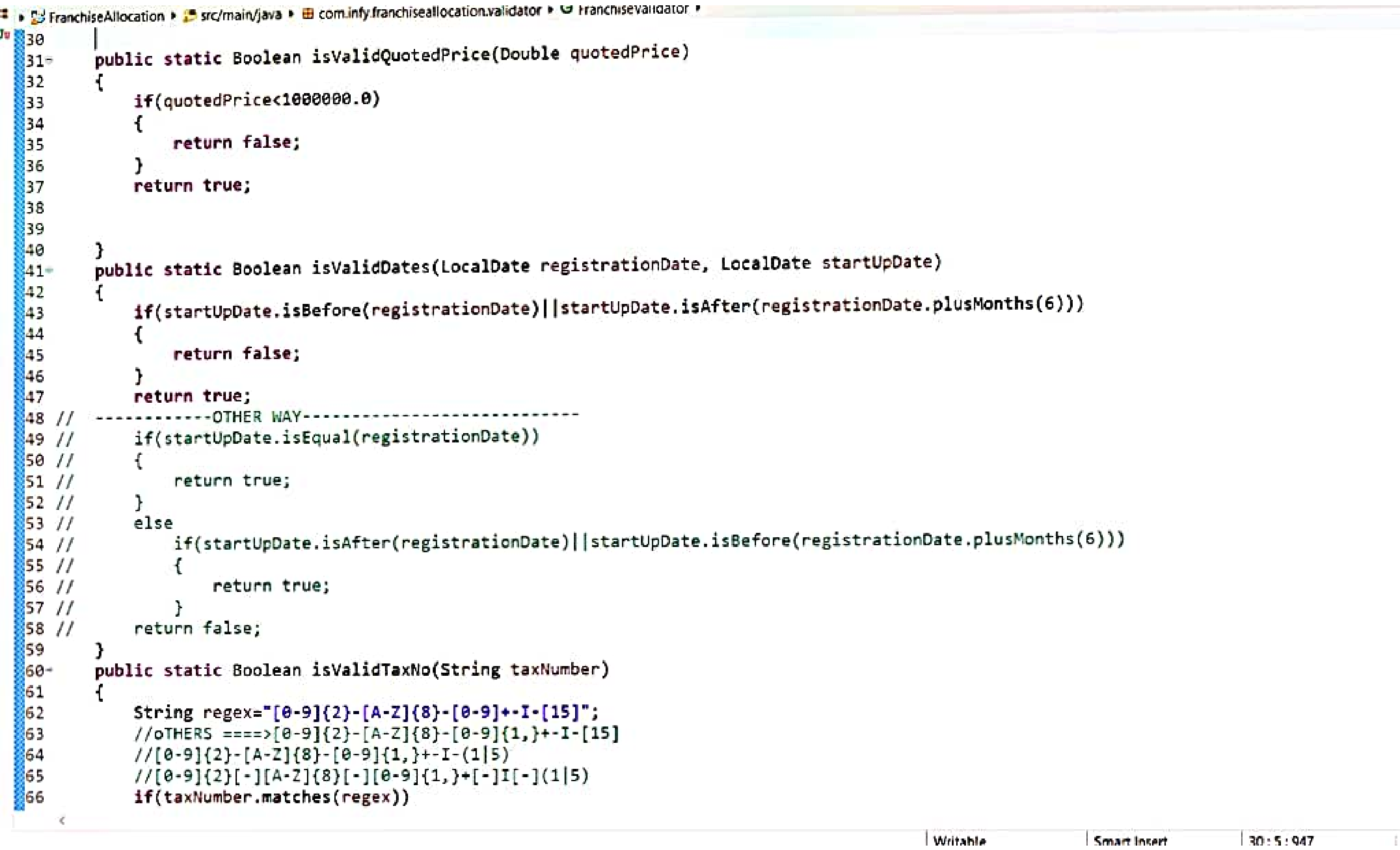


```java
package com.infy.franchiseallocation.repository;

import java.util.List;

import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.CrudRepository;

import com.infy.franchiseallocation.entity.Franchise;

public interface FranchiseRepository extends CrudRepository<Franchise, Integer> {

    @Query("SELECT f FROM StoreFranchise f WHERE f.location = ?1")
    public List<Franchise> getFranchiseByLocation(String location);

    public List<Franchise> findByfranchiseNameAndLocation(String franchiseName, String location);

}
```

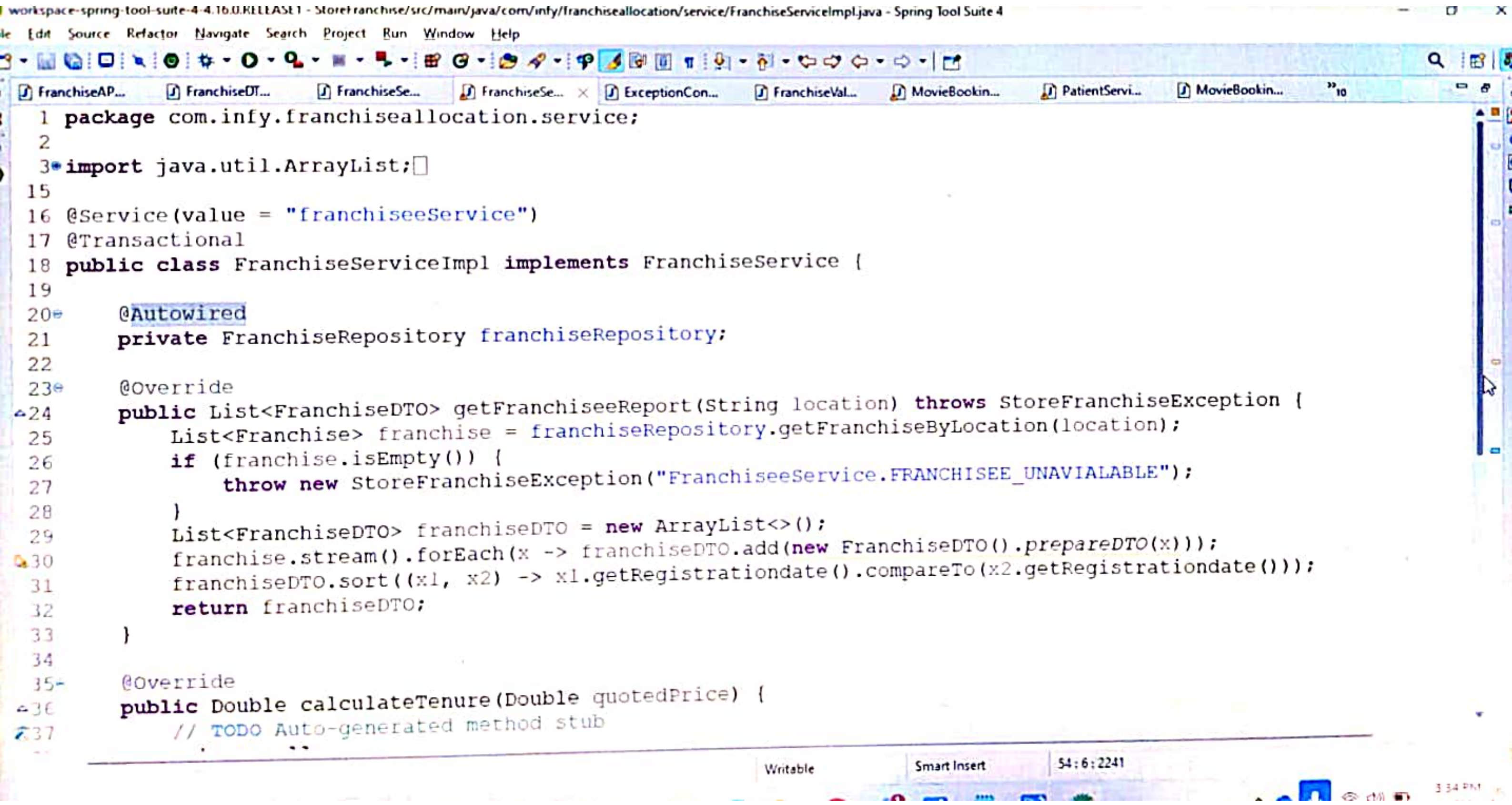

eclipse-workspace - FranchiseAllocation/src/main/java/com/infy/franchiseallocation/validator/FranchiseValidator.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
FranchiseValidator.java
Regex.java
application.properties
FranchiseAllocation
src/main/java
com.infy.franchiseallocation.validator
FranchiseValidator

```java


    }
    public static Boolean isValidDates(LocalDate registrationDate, LocalDate startUpDate)
    {
        if(startUpDate.isBefore(registrationDate)||startUpDate.isAfter(registrationDate.plusMonths(6)))
        {
            return false;
        }
        return true;
//  ------------OTHER WAY----------------------------
//      if(startUpDate.isEqual(registrationDate))
//      {
//          return true;
//      }
//      else
//          if(startUpDate.isAfter(registrationDate)||startUpDate.isBefore(registrationDate.plusMonths(6)))
//          {
//              return true;
//          }
//      return false;
    }
    public static Boolean isValidTaxNo(String taxNumber)
    {
        String regex="[0-9]{2}-[A-Z]{8}-[0-9]+-I-[15]";
        //oTHERS ====>[0-9]{2}-[A-Z]{8}-[0-9]{1,}+-I-[15]
        //[0-9]{2}-[A-Z]{8}-[0-9]{1,}+-I-(1|5)
        //[0-9]{2}[-][A-Z]{8}[-][0-9]{1,}+[-]I[-](1|5)
        if(taxNumber.matches(regex))
        {
            return true;
        }
        return false;

    }

}
```

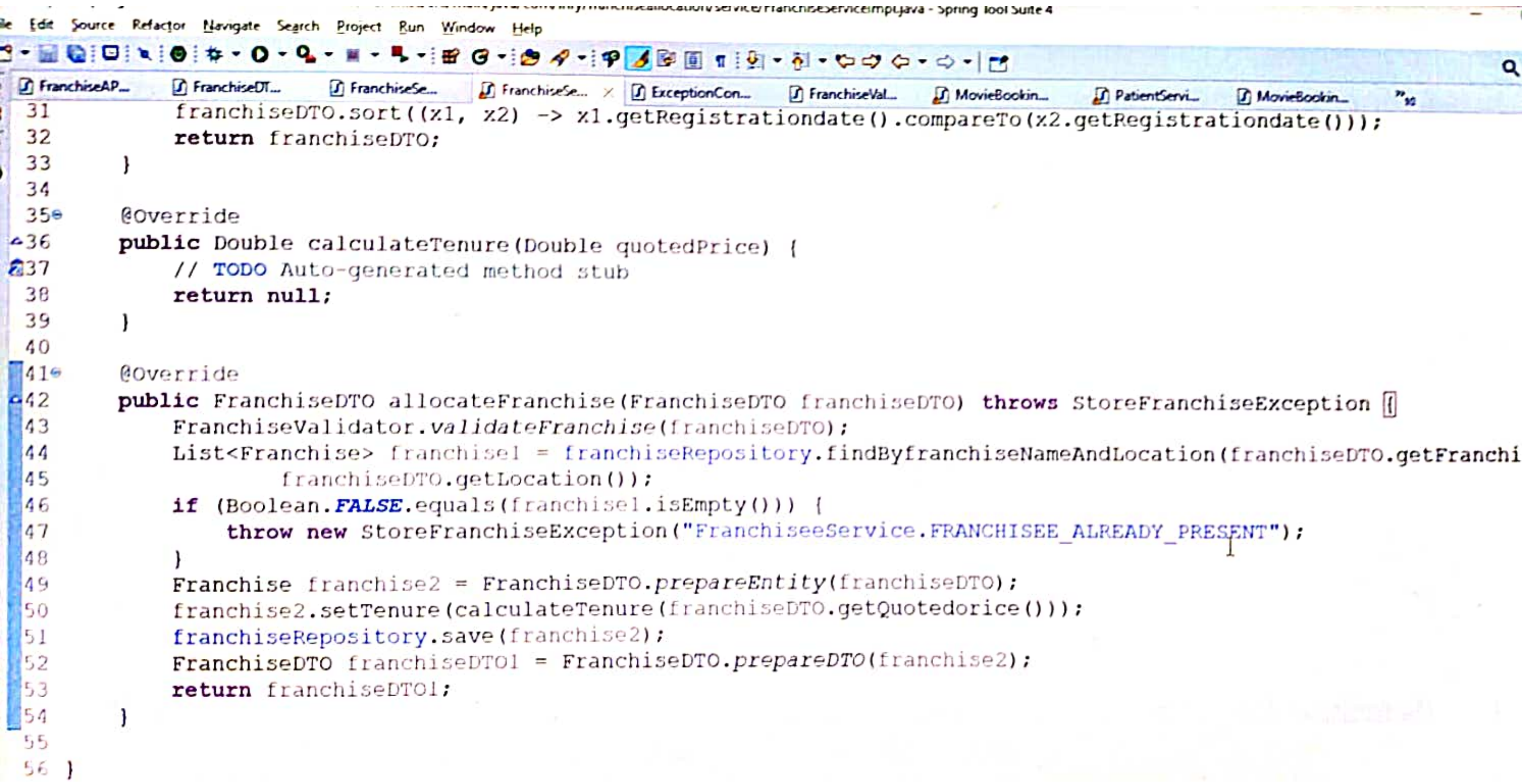


FranchiseAllocation ▸ src/main/java ▸ com.infy.franchiseallocation.validator ▸ FranchiseValidator ▸
Writable
Smart Insert
30 : 5 : 947

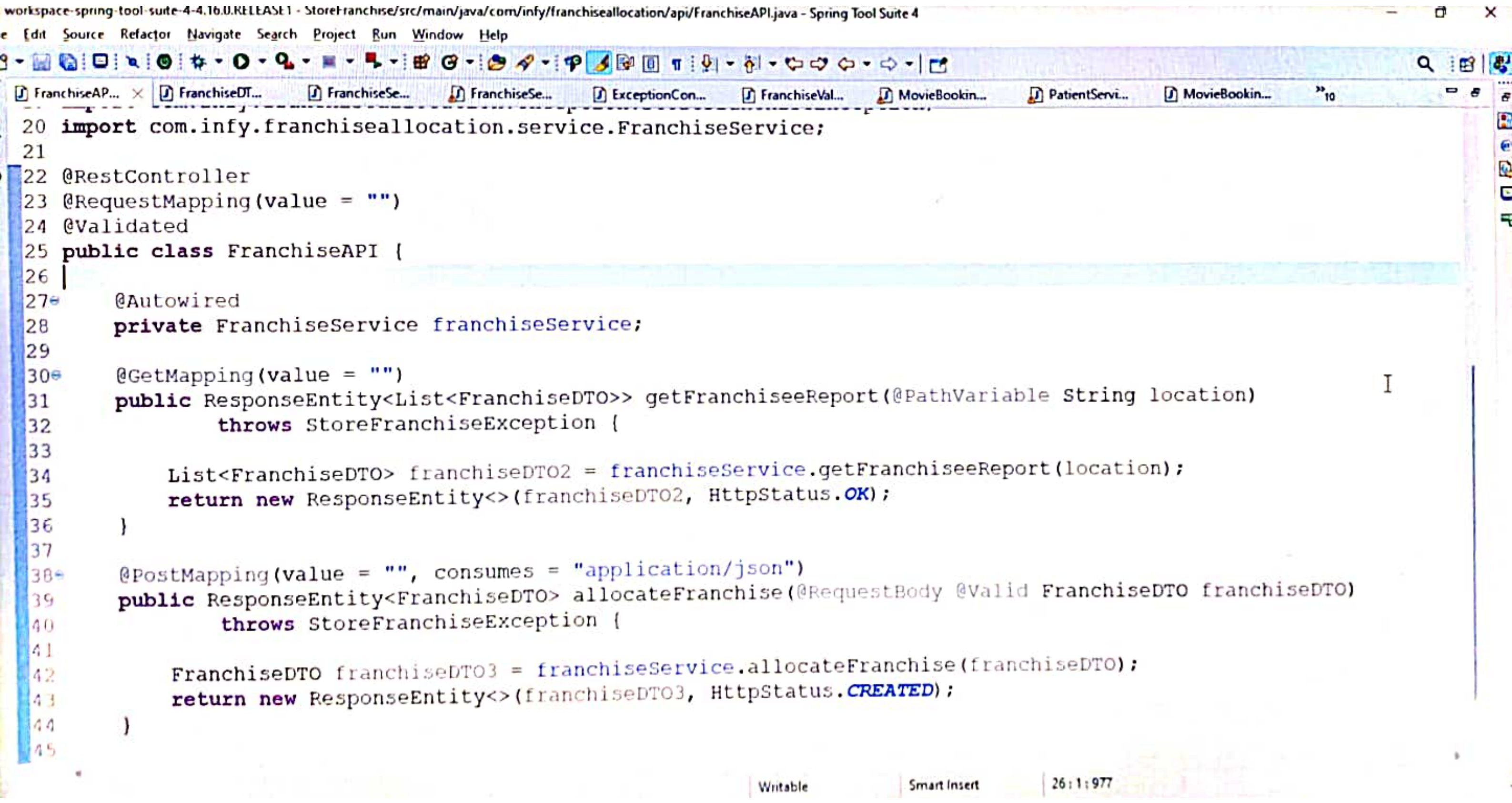
```
    
    public static Boolean isValidQuotedPrice(Double quotedPrice)
    {
        if(quotedPrice<1000000.0)
        {
            return false;
        }
        return true;


    }
    public static Boolean isValidDates(LocalDate registrationDate, LocalDate startUpDate)
    {
        if(startUpDate.isBefore(registrationDate)||startUpDate.isAfter(registrationDate.plusMonths(6)))
        {
            return false;
        }
        return true;
//  ------------OTHER WAY----------------------------
//      if(startUpDate.isEqual(registrationDate))
//      {
//          return true;
//      }
//      else
//          if(startUpDate.isAfter(registrationDate)||startUpDate.isBefore(registrationDate.plusMonths(6)))
//          {
//              return true;
//          }
//      return false;
    }
    public static Boolean isValidTaxNo(String taxNumber)
    {
        String regex="[0-9]{2}-[A-Z]{8}-[0-9]+-I-[15]";
        //oTHERS ====>[0-9]{2}-[A-Z]{8}-[0-9]{1,}+-I-[15]
        //[0-9]{2}-[A-Z]{8}-[0-9]{1,}+-I-(1|5)
        //[0-9]{2}[-][A-Z]{8}[-][0-9]{1,}+[-]I[-](1|5)
        if(taxNumber.matches(regex))
```


workspace-spring-tool-suite-4-4.16.0.RELEASE1 - StoreFranchise/src/main/java/com/infy/franchiseallocation/service/FranchiseServiceImpl.java - Spring Tool Suite 4
ile Edit Source Refactor Navigate Search Project Run Window Help
FranchiseAP...
FranchiseDT...
FranchiseSe...
FranchiseSe...
ExceptionCon...
FranchiseVal...
MovieBookin...
PatientServi...
MovieBookin...
»10
Writable
Smart Insert
54 : 6 : 2241

```java
package com.infy.franchiseallocation.service;

import java.util.ArrayList;

@Service(value = "franchiseeService")
@Transactional
public class FranchiseServiceImpl implements FranchiseService {

    @Autowired
    private FranchiseRepository franchiseRepository;

    @Override
    public List<FranchiseDTO> getFranchiseeReport(String location) throws StoreFranchiseException {
        List<Franchise> franchise = franchiseRepository.getFranchiseByLocation(location);
        if (franchise.isEmpty()) {
            throw new StoreFranchiseException("FranchiseeService.FRANCHISEE_UNAVIALABLE");
        }
        List<FranchiseDTO> franchiseDTO = new ArrayList<>();
        franchise.stream().forEach(x -> franchiseDTO.add(new FranchiseDTO().prepareDTO(x)));
        franchiseDTO.sort((x1, x2) -> x1.getRegistrationdate().compareTo(x2.getRegistrationdate()));
        return franchiseDTO;
    }

    @Override
    public Double calculateTenure(Double quotedPrice) {
        // TODO Auto-generated method stub
```

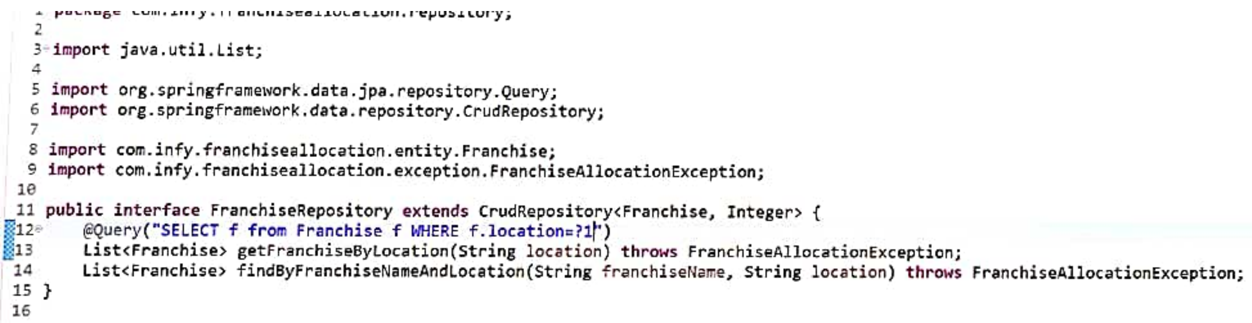


```
        franchiseDTO.sort((x1, x2) -> x1.getRegistrationdate().compareTo(x2.getRegistrationdate()));
        return franchiseDTO;
    }

    @Override
    public Double calculateTenure(Double quotedPrice) {
        // TODO Auto-generated method stub
        return null;
    }

    @Override
    public FranchiseDTO allocateFranchise(FranchiseDTO franchiseDTO) throws StoreFranchiseException {
        FranchiseValidator.validateFranchise(franchiseDTO);
        List<Franchise> franchise1 = franchiseRepository.findByfranchiseNameAndLocation(franchiseDTO.getFranchi
                franchiseDTO.getLocation());
        if (Boolean.FALSE.equals(franchise1.isEmpty())) {
            throw new StoreFranchiseException("FranchiseeService.FRANCHISEE_ALREADY_PRESENT");
        }
        Franchise franchise2 = FranchiseDTO.prepareEntity(franchiseDTO);
        franchise2.setTenure(calculateTenure(franchiseDTO.getQuotedorice()));
        franchiseRepository.save(franchise2);
        FranchiseDTO franchiseDTO1 = FranchiseDTO.prepareDTO(franchise2);
        return franchiseDTO1;
    }

}
```

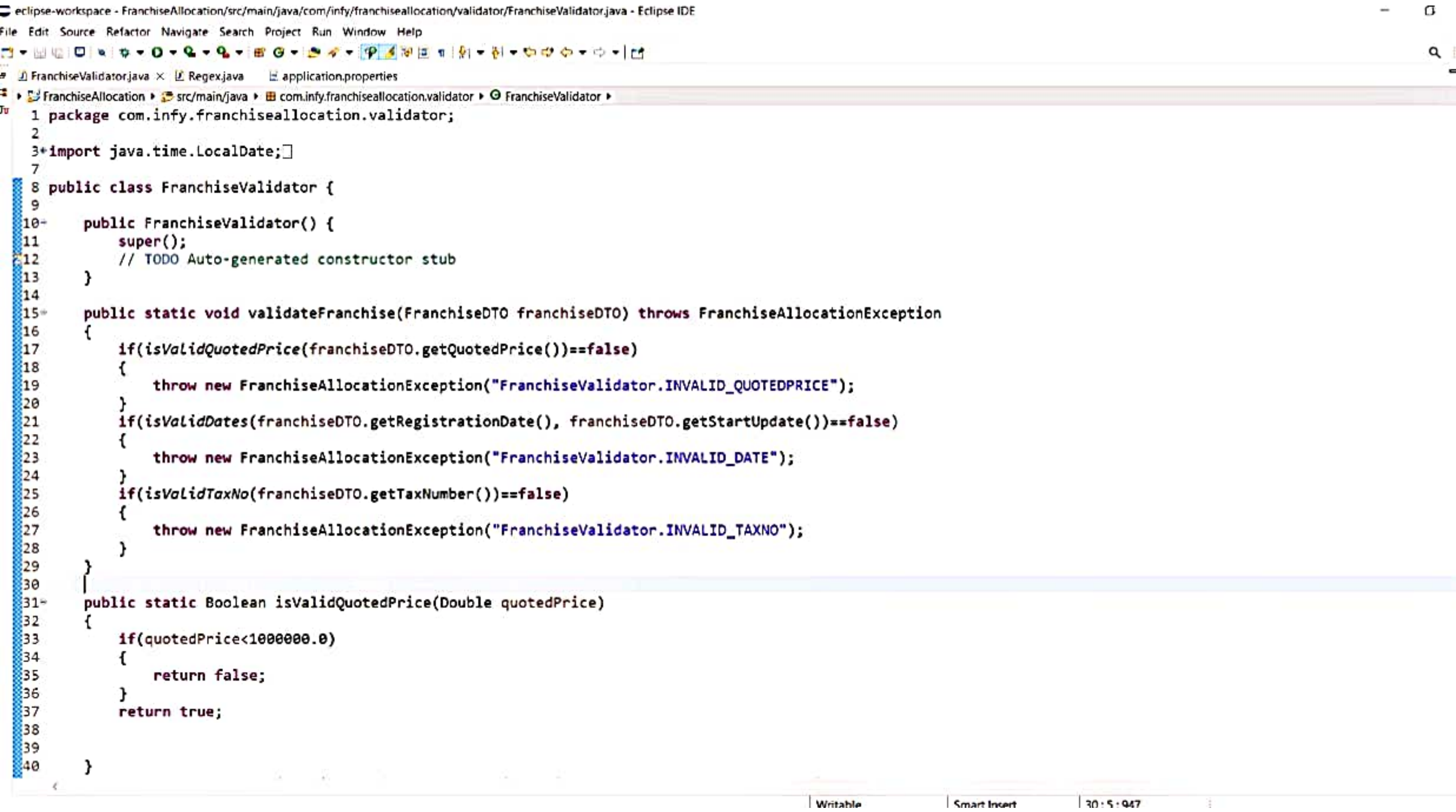


workspace-spring-tool-suite-4-4.16.0.RELEASE1 - StoreFranchise/src/main/java/com/infy/franchiseallocation/api/FranchiseAPI.java - Spring Tool Suite 4
Edit Source Refactor Navigate Search Project Run Window Help
FranchiseAP...
FranchiseDT...
FranchiseSe...
FranchiseSe...
ExceptionCon...
FranchiseVal...
MovieBookin...
PatientServi...
MovieBookin...
»10
Writable
Smart Insert
26 : 1 : 977

```
import com.infy.franchiseallocation.service.FranchiseService;

@RestController
@RequestMapping(value = "")
@Validated
public class FranchiseAPI {

    @Autowired
    private FranchiseService franchiseService;

    @GetMapping(value = "")
    public ResponseEntity<List<FranchiseDTO>> getFranchiseeReport(@PathVariable String location)
            throws StoreFranchiseException {

        List<FranchiseDTO> franchiseDTO2 = franchiseService.getFranchiseeReport(location);
        return new ResponseEntity<>(franchiseDTO2, HttpStatus.OK);
    }

    @PostMapping(value = "", consumes = "application/json")
    public ResponseEntity<FranchiseDTO> allocateFranchise(@RequestBody @Valid FranchiseDTO franchiseDTO)
            throws StoreFranchiseException {

        FranchiseDTO franchiseDTO3 = franchiseService.allocateFranchise(franchiseDTO);
        return new ResponseEntity<>(franchiseDTO3, HttpStatus.CREATED);
    }

```

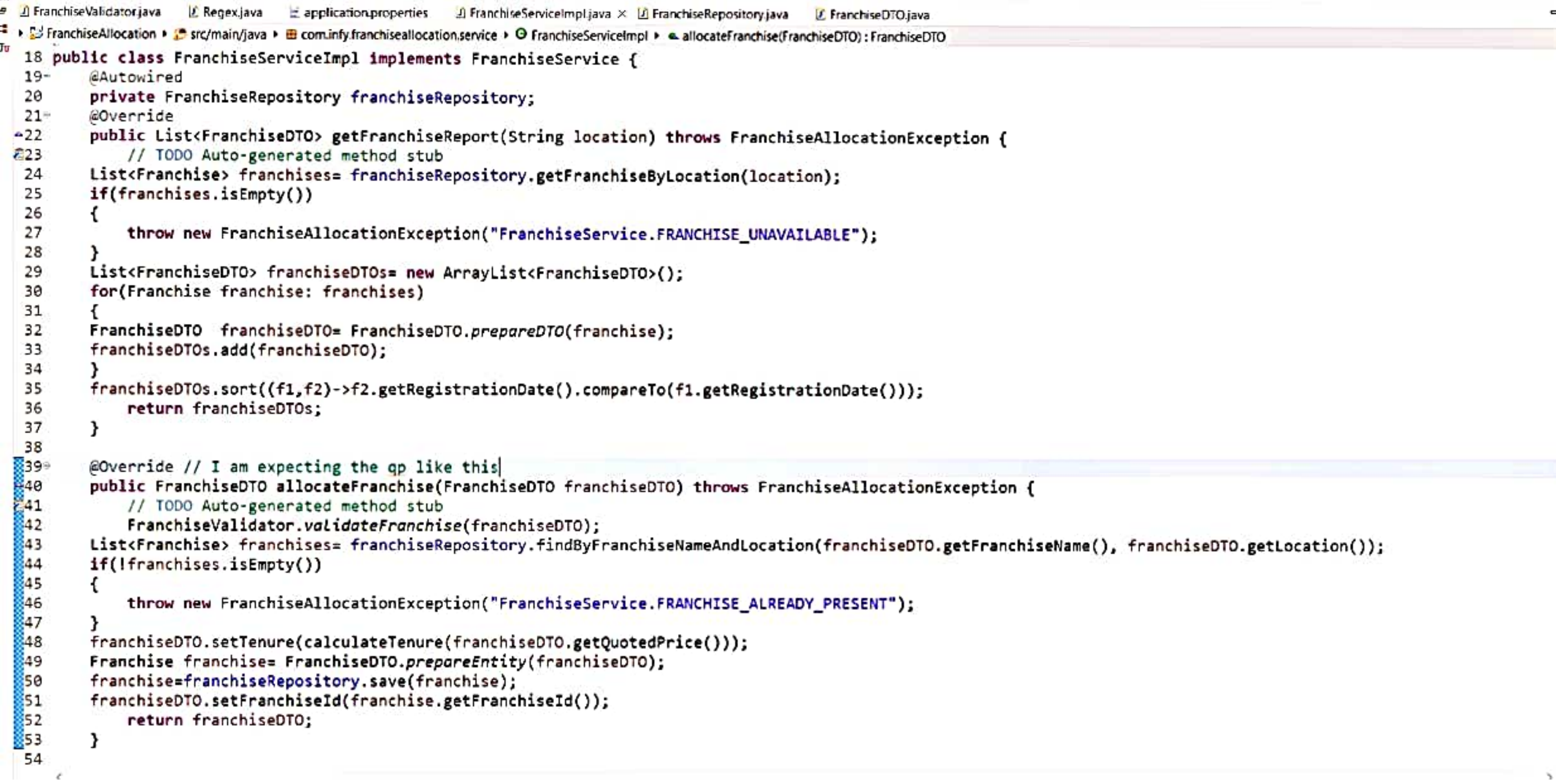


```
package com.infy.franchiseallocation.repository;

import java.util.List;

import org.springframework.data.jpa.repository.Query;
import org.springframework.data.repository.CrudRepository;

import com.infy.franchiseallocation.entity.Franchise;
import com.infy.franchiseallocation.exception.FranchiseAllocationException;

public interface FranchiseRepository extends CrudRepository<Franchise, Integer> {
    @Query("SELECT f from Franchise f WHERE f.location=?1")
    List<Franchise> getFranchiseByLocation(String location) throws FranchiseAllocationException;
    List<Franchise> findByFranchiseNameAndLocation(String franchiseName, String location) throws FranchiseAllocationException;
}
```



eclipse-workspace - FranchiseAllocation/src/main/java/com/infy/franchiseallocation/validator/FranchiseValidator.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
FranchiseValidator.java
Regex.java
application.properties
FranchiseAllocation ▸ src/main/java ▸ com.infy.franchiseallocation.validator ▸ FranchiseValidator ▸
Writable
Smart Insert
30 : 5 : 947

```java
package com.infy.franchiseallocation.validator;

import java.time.LocalDate;

public class FranchiseValidator {

    public FranchiseValidator() {
        super();
        // TODO Auto-generated constructor stub
    }

    public static void validateFranchise(FranchiseDTO franchiseDTO) throws FranchiseAllocationException
    {
        if(isValidQuotedPrice(franchiseDTO.getQuotedPrice())==false)
        {
            throw new FranchiseAllocationException("FranchiseValidator.INVALID_QUOTEDPRICE");
        }
        if(isValidDates(franchiseDTO.getRegistrationDate(), franchiseDTO.getStartUpdate())==false)
        {
            throw new FranchiseAllocationException("FranchiseValidator.INVALID_DATE");
        }
        if(isValidTaxNo(franchiseDTO.getTaxNumber())==false)
        {
            throw new FranchiseAllocationException("FranchiseValidator.INVALID_TAXNO");
        }
    }

    public static Boolean isValidQuotedPrice(Double quotedPrice)
    {
        if(quotedPrice<1000000.0)
        {
            return false;
        }
        return true;


    }
```



FranchiseValidator.java
Regex.java
application.properties
FranchiseServiceImpl.java ×
FranchiseRepository.java
FranchiseDTO.java
FranchiseAllocation ▸ src/main/java ▸ com.infy.franchiseallocation.service ▸ FranchiseServiceImpl ▸ allocateFranchise(FranchiseDTO) : FranchiseDTO

```
public class FranchiseServiceImpl implements FranchiseService {
    @Autowired
    private FranchiseRepository franchiseRepository;
    @Override
    public List<FranchiseDTO> getFranchiseReport(String location) throws FranchiseAllocationException {
        // TODO Auto-generated method stub
    List<Franchise> franchises= franchiseRepository.getFranchiseByLocation(location);
    if(franchises.isEmpty())
    {
        throw new FranchiseAllocationException("FranchiseService.FRANCHISE_UNAVAILABLE");
    }
    List<FranchiseDTO> franchiseDTOs= new ArrayList<FranchiseDTO>();
    for(Franchise franchise: franchises)
    {
    FranchiseDTO  franchiseDTO= FranchiseDTO.prepareDTO(franchise);
    franchiseDTOs.add(franchiseDTO);
    }
    franchiseDTOs.sort((f1,f2)->f2.getRegistrationDate().compareTo(f1.getRegistrationDate()));
        return franchiseDTOs;
    }

    @Override // I am expecting the qp like this
    public FranchiseDTO allocateFranchise(FranchiseDTO franchiseDTO) throws FranchiseAllocationException {
        // TODO Auto-generated method stub
        FranchiseValidator.validateFranchise(franchiseDTO);
    List<Franchise> franchises= franchiseRepository.findByFranchiseNameAndLocation(franchiseDTO.getFranchiseName(), franchiseDTO.getLocation());
    if(!franchises.isEmpty())
    {
        throw new FranchiseAllocationException("FranchiseService.FRANCHISE_ALREADY_PRESENT");
    }
    franchiseDTO.setTenure(calculateTenure(franchiseDTO.getQuotedPrice()));
    Franchise franchise= FranchiseDTO.prepareEntity(franchiseDTO);
    franchise=franchiseRepository.save(franchise);
    franchiseDTO.setFranchiseId(franchise.getFranchiseId());
        return franchiseDTO;
    }

```