

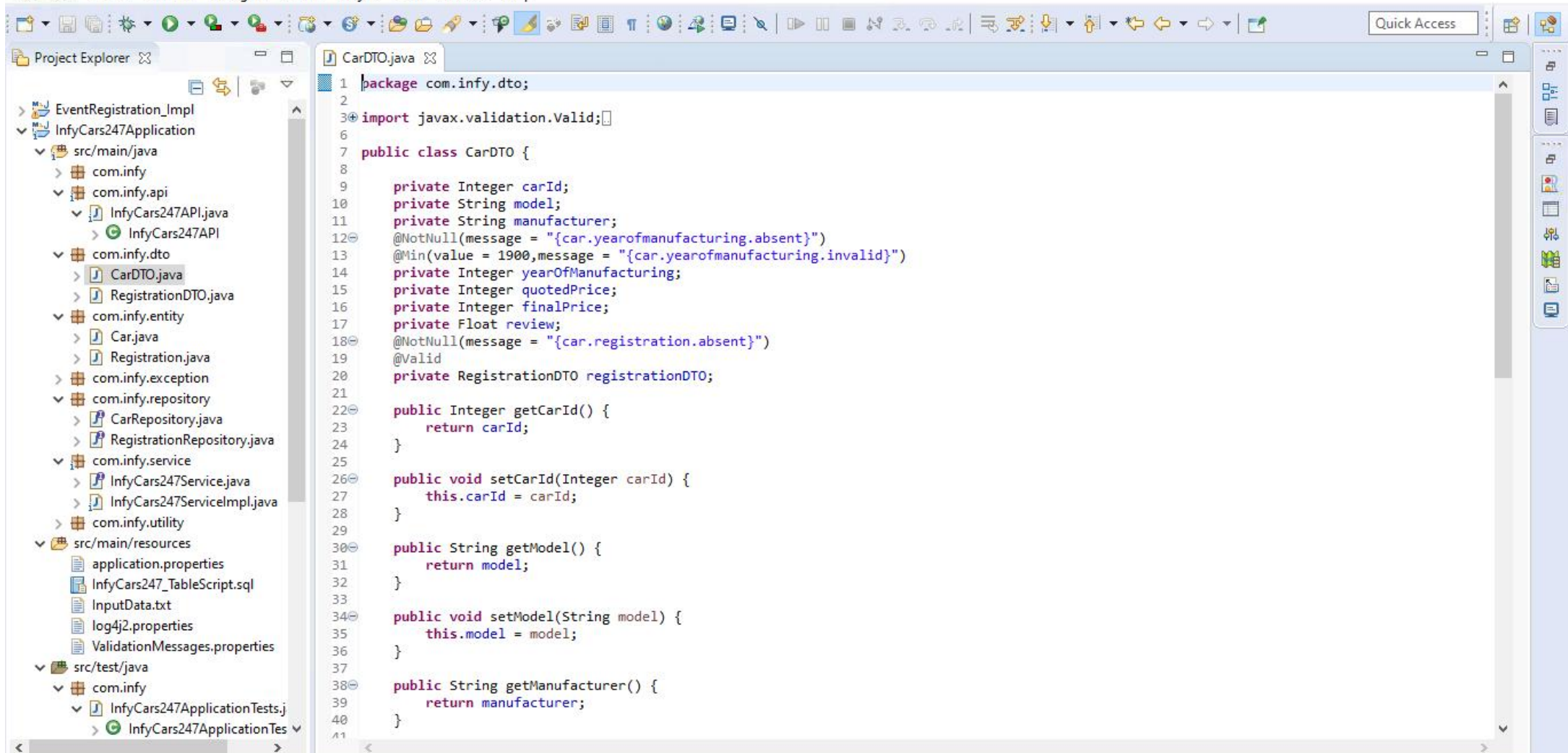
```
1 package com.infy.api;
2
3 import javax.validation.Valid;
4 @RestController
5 @Validated
6 @RequestMapping("/infyCars247-api")
7 public class InfyCars247API {
8     @Autowired
9     private InfyCars247Service infyCars247Service;
10    @Autowired
11    private Environment environment;
12    @PostMapping("/cars")
13    public ResponseEntity<String> sellCar(@Valid @RequestBody CarDTO carDTO) throws InfyCars247Exception{
14        Integer carId = infyCars247Service.sellCar(carDTO);
15        String successMsg = environment.getProperty("API.CAR_REGISTERED_SUCCESS")+carId;
16        return new ResponseEntity<String>(successMsg,HttpStatus.CREATED);
17    }
18
19    @GetMapping("/users/{phoneNumber}")
20    public ResponseEntity<RegistrationDTO> getUserDetails(@Pattern(regexp = "[1-9][0-9]{9}", message = "{registration.userphone.invalid}")
21        @PathVariable String phoneNumber) throws InfyCars247Exception{
22
23        return new ResponseEntity<>(infyCars247Service.getUserDetailsByPhone(phoneNumber), HttpStatus.FOUND);
24    }
25 }
```

Writable

Smart Insert

45 : 1 : 1825





Project Explorer

- EventRegistration\_Impl
- InfyCars247Application
  - src/main/java
    - com.infy
      - com.infy.api
        - InfyCars247API.java
        - InfyCars247API
      - com.infy.dto
        - CarDTO.java
        - RegistrationDTO.java
      - com.infy.entity
        - Car.java
        - Registration.java
      - com.infy.exception
      - com.infy.repository
        - CarRepository.java
        - RegistrationRepository.java
      - com.infy.service
        - InfyCars247Service.java
        - InfyCars247ServiceImpl.java
      - com.infy.utility
    - src/main/resources
      - application.properties
      - InfyCars247\_TableScript.sql
      - InputData.txt
      - log4j2.properties
      - ValidationMessages.properties
    - src/test/java
      - com.infy
        - InfyCars247ApplicationTests.java
        - InfyCars247ApplicationTests

CarDTO.java

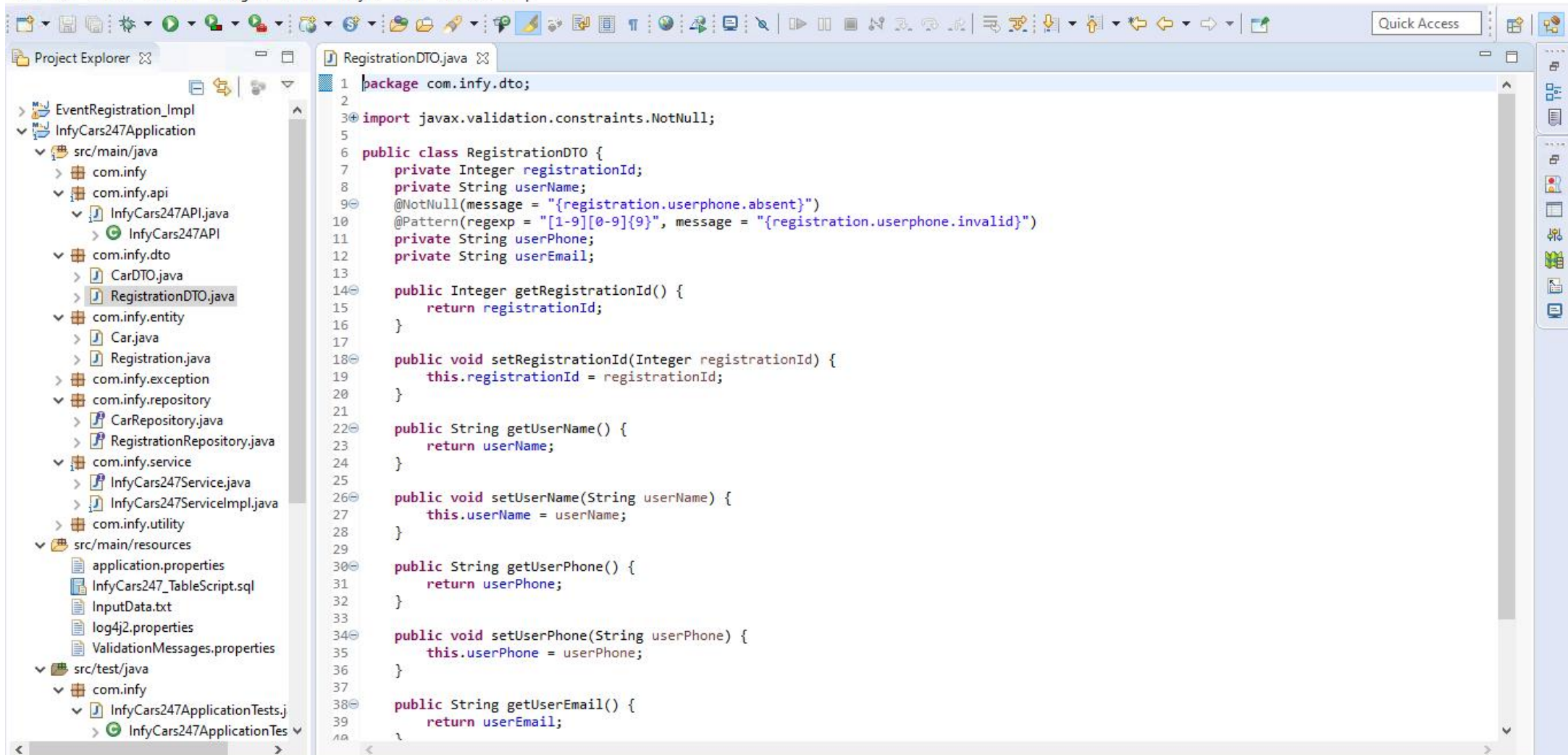
```
1 package com.infy.dto;
2
3 import javax.validation.Valid;
4
5
6
7 public class CarDTO {
8
9     private Integer carId;
10    private String model;
11    private String manufacturer;
12    @NotNull(message = "{car.yearofmanufacturing.absent}")
13    @Min(value = 1900,message = "{car.yearofmanufacturing.invalid}")
14    private Integer yearOfManufacturing;
15    private Integer quotedPrice;
16    private Integer finalPrice;
17    private Float review;
18    @NotNull(message = "{car.registration.absent}")
19    @Valid
20    private RegistrationDTO registrationDTO;
21
22    public Integer getCarId() {
23        return carId;
24    }
25
26    public void setCarId(Integer carId) {
27        this.carId = carId;
28    }
29
30    public String getModel() {
31        return model;
32    }
33
34    public void setModel(String model) {
35        this.model = model;
36    }
37
38    public String getManufacturer() {
39        return manufacturer;
40    }
41}
```

Writable

Smart Insert

1:1:0





Project Explorer

- EventRegistration\_Impl
- InfyCars247Application
  - src/main/java
    - com.infy
    - com.infy.api
      - InfyCars247API.java
      - InfyCars247API
    - com.infy.dto
      - CarDTO.java
      - RegistrationDTO.java
    - com.infy.entity
      - Car.java
      - Registration.java
    - com.infy.exception
    - com.infy.repository
      - CarRepository.java
      - RegistrationRepository.java
    - com.infy.service
      - InfyCars247Service.java
      - InfyCars247ServiceImpl.java
    - com.infy.utility
  - src/main/resources
    - application.properties
    - InfyCars247\_TableScript.sql
    - InputData.txt
    - log4j2.properties
    - ValidationMessages.properties
  - src/test/java
    - com.infy
      - InfyCars247ApplicationTests.j
      - InfyCars247ApplicationTes

RegistrationDTO.java

```
1 package com.infy.dto;
2
3 import javax.validation.constraints.NotNull;
4
5
6 public class RegistrationDTO {
7     private Integer registrationId;
8     private String userName;
9     @NotNull(message = "{registration.userphone.absent}")
10    @Pattern(regexp = "[1-9][0-9]{9}", message = "{registration.userphone.invalid}")
11    private String userPhone;
12    private String userEmail;
13
14    public Integer getRegistrationId() {
15        return registrationId;
16    }
17
18    public void setRegistrationId(Integer registrationId) {
19        this.registrationId = registrationId;
20    }
21
22    public String getUserName() {
23        return userName;
24    }
25
26    public void setUserName(String userName) {
27        this.userName = userName;
28    }
29
30    public String getUserPhone() {
31        return userPhone;
32    }
33
34    public void setUserPhone(String userPhone) {
35        this.userPhone = userPhone;
36    }
37
38    public String getUserEmail() {
39        return userEmail;
40    }
41 }
```

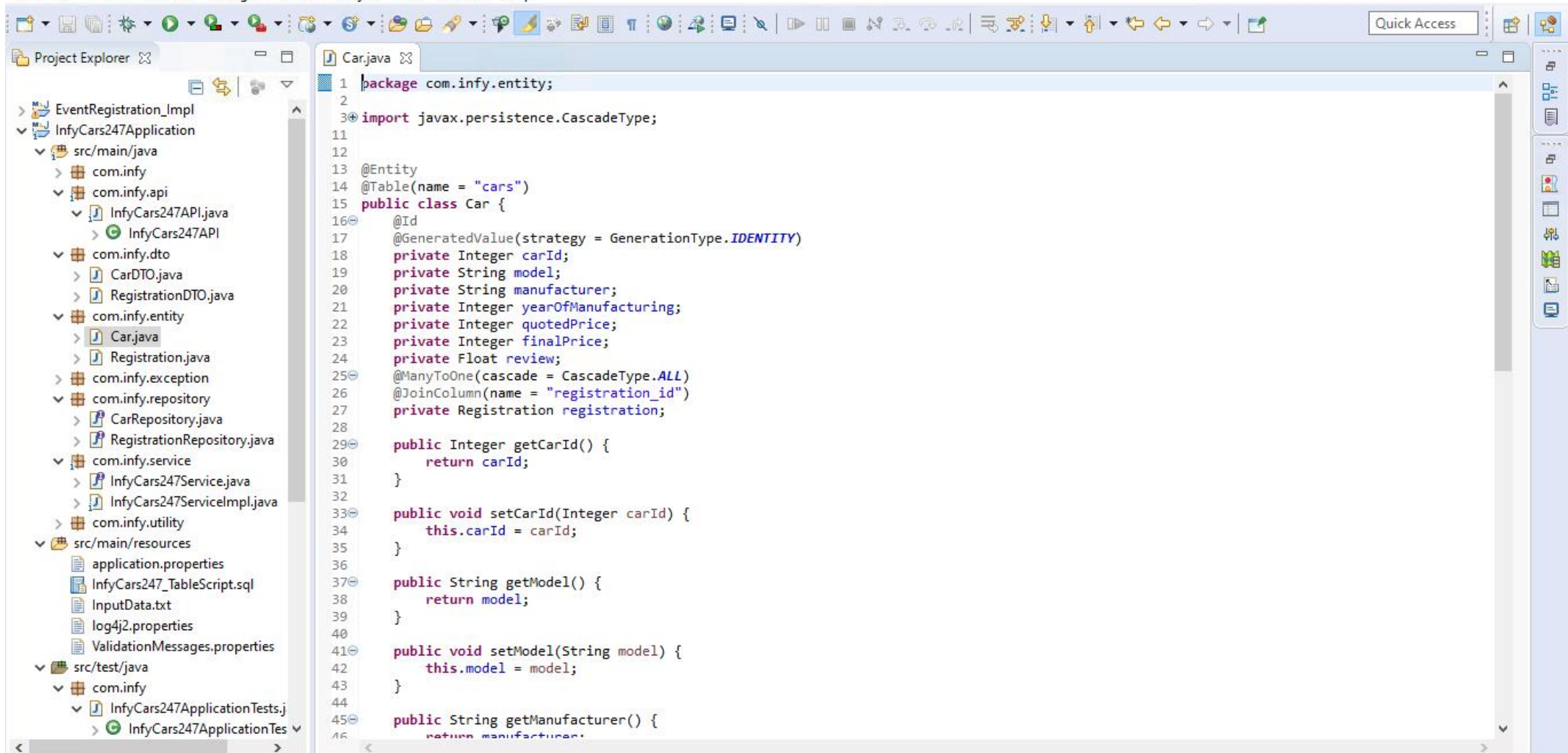
Writable

Smart Insert

1:1:0







The screenshot shows the Eclipse IDE interface. On the left, the Project Explorer displays the project structure for 'InfyCars247Application'. The 'src/main/java' directory is expanded, showing the 'com.infy.entity' package containing 'Car.java'. The main editor window displays the code for 'Car.java'. The code defines a JPA entity 'Car' with attributes 'carId', 'model', 'manufacturer', 'yearOfManufacturing', 'quotedPrice', 'finalPrice', and 'review'. It includes annotations for 'Id', 'GeneratedValue', 'Entity', 'Table', 'ManyToMany', and 'JoinColumn'. The code also includes getter and setter methods for 'carId' and 'model'.

```
1 package com.infy.entity;
2
3 import javax.persistence.CascadeType;
4
5
6
7
8
9
10
11
12
13 @Entity
14 @Table(name = "cars")
15 public class Car {
16     @Id
17     @GeneratedValue(strategy = GenerationType.IDENTITY)
18     private Integer carId;
19     private String model;
20     private String manufacturer;
21     private Integer yearOfManufacturing;
22     private Integer quotedPrice;
23     private Integer finalPrice;
24     private Float review;
25     @ManyToMany(cascade = CascadeType.ALL)
26     @JoinColumn(name = "registration_id")
27     private Registration registration;
28
29     public Integer getCarId() {
30         return carId;
31     }
32
33     public void setCarId(Integer carId) {
34         this.carId = carId;
35     }
36
37     public String getModel() {
38         return model;
39     }
40
41     public void setModel(String model) {
42         this.model = model;
43     }
44
45     public String getManufacturer() {
46         return manufacturer;
47     }
48 }
```

Writable

Smart Insert

1:1:0



The screenshot shows the Eclipse IDE interface. On the left is the Project Explorer, displaying the project structure of 'InfyCars247Application'. The 'src/main/java' directory is expanded, showing the 'com.infy.entity' package. The 'Registration.java' file is selected and highlighted. The main editor window displays the code for 'Registration.java'. The code defines a JPA entity with attributes for registrationId, userName, userPhone, and userEmail, and includes getter and setter methods for each attribute.

```
1 package com.infy.entity;
2
3 import javax.persistence.Entity;
4
5
6
7
8 @Entity
9 @Table(name = "registration")
10 public class Registration {
11     @Id
12     private Integer registrationId;
13     private String userName;
14     private String userPhone;
15     private String userEmail;
16
17     public Integer getRegistrationId() {
18         return registrationId;
19     }
20
21     public void setRegistrationId(Integer registrationId) {
22         this.registrationId = registrationId;
23     }
24
25     public String getUserName() {
26         return userName;
27     }
28
29     public void setUserName(String userName) {
30         this.userName = userName;
31     }
32
33     public String getUserPhone() {
34         return userPhone;
35     }
36
37     public void setUserPhone(String userPhone) {
38         this.userPhone = userPhone;
39     }
40
41     public String getUserEmail() {
```

Writable

Smart Insert

1:1:0



Project Explorer

- EventRegistration\_Impl
- InfyCars247Application
  - src/main/java
    - com.infy
    - com.infy.api
    - com.infy.dto
    - com.infy.entity
    - com.infy.exception
    - com.infy.repository
      - CarRepository.java
      - CarRepository
      - RegistrationRepository.java
    - com.infy.service
      - InfyCars247Service.java
      - InfyCars247ServiceImpl.java
    - com.infy.utility
  - src/main/resources
    - application.properties
    - InfyCars247\_TableScript.sql
    - InputData.txt
    - log4j2.properties
    - ValidationMessages.properties
  - src/test/java
    - com.infy
      - InfyCars247ApplicationTests.j
        - InfyCars247ApplicationTes
  - Maven Dependencies
  - JRE System Library [JavaSE-11]
  - java
  - log
    - ErrorLog.log

CarRepository.java

```

1 package com.infy.repository;
2
3 import org.springframework.data.jpa.repository.Query;
4
5
6
7
8
9 public interface CarRepository extends CrudRepository<Car, Integer> {
10     @Query("select avg(c.review) from Car c where c.registration.registrationId=?1")
11     Float getAvgReview (Integer registrationId);
12
13 }
14

```



Project Explorer

- EventRegistration\_Impl
- InfyCars247Application
  - src/main/java
    - com.infy
    - com.infy.api
    - com.infy.dto
    - com.infy.entity
    - com.infy.exception
    - com.infy.repository
      - CarRepository.java
        - CarRepository
      - RegistrationRepository.java
    - com.infy.service
      - InfyCars247Service.java
      - InfyCars247ServiceImpl.java
    - com.infy.utility
  - src/main/resources
    - application.properties
    - InfyCars247\_TableScript.sql
    - InputData.txt
    - log4j2.properties
    - ValidationMessages.properties
  - src/test/java
    - com.infy
      - InfyCars247ApplicationTests.j
        - InfyCars247ApplicationTes
  - Maven Dependencies
  - JRE System Library [JavaSE-11]
  - java
  - log
    - ErrorLog.log

RegistrationRepository.java

```

1 package com.infy.repository;
2
3 import org.springframework.data.repository.CrudRepository;
4
5
6 public interface RegistrationRepository extends CrudRepository<Registration, Integer>{
7     Registration findByUserPhone(String phoneNumber);
8 }
9

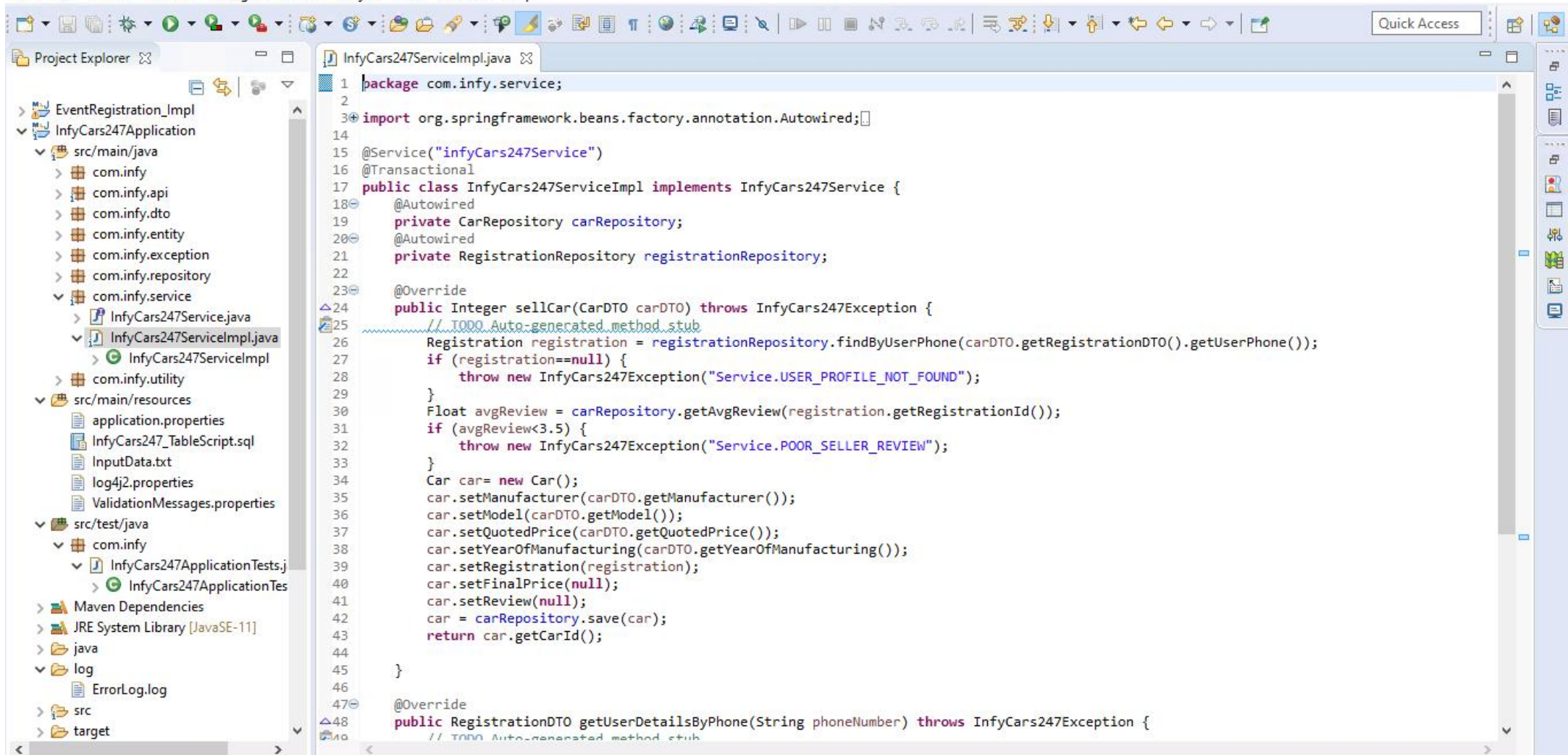
```

Writable

Smart Insert

1:1:0





Project Explorer

- EventRegistration\_Impl
- InfyCars247Application
  - src/main/java
    - com.infy
    - com.infy.api
    - com.infy.dto
    - com.infy.entity
    - com.infy.exception
    - com.infy.repository
    - com.infy.service
      - InfyCars247Service.java
      - InfyCars247ServiceImpl.java
    - com.infy.utility
  - src/main/resources
    - application.properties
    - InfyCars247\_TableScript.sql
    - InputData.txt
    - log4j2.properties
    - ValidationMessages.properties
  - src/test/java
    - com.infy
      - InfyCars247ApplicationTests.java
      - InfyCars247ApplicationTest
  - Maven Dependencies
  - JRE System Library [JavaSE-11]
  - java
  - log
    - ErrorLog.log
  - src
  - target

InfyCars247ServiceImpl.java

```
1 package com.infy.service;
2
3 import org.springframework.beans.factory.annotation.Autowired;
4
5 @Service("infyCars247Service")
6 @Transactional
7 public class InfyCars247ServiceImpl implements InfyCars247Service {
8     @Autowired
9     private CarRepository carRepository;
10    @Autowired
11    private RegistrationRepository registrationRepository;
12
13    @Override
14    public Integer sellCar(CarDTO carDTO) throws InfyCars247Exception {
15        // TODO Auto-generated method stub
16        Registration registration = registrationRepository.findByUserPhone(carDTO.getRegistrationDTO().getUserPhone());
17        if (registration==null) {
18            throw new InfyCars247Exception("Service.USER_PROFILE_NOT_FOUND");
19        }
20        Float avgReview = carRepository.getAvgReview(registration.getRegistrationId());
21        if (avgReview<3.5) {
22            throw new InfyCars247Exception("Service.POOR_SELLER_REVIEW");
23        }
24        Car car= new Car();
25        car.setManufacturer(carDTO.getManufacturer());
26        car.setModel(carDTO.getModel());
27        car.setQuotedPrice(carDTO.getQuotedPrice());
28        car.setYearOfManufacturing(carDTO.getYearOfManufacturing());
29        car.setRegistration(registration);
30        car.setFinalPrice(null);
31        car.setReview(null);
32        car = carRepository.save(car);
33        return car.getCarId();
34    }
35
36    @Override
37    public RegistrationDTO getUserDetailsByPhone(String phoneNumber) throws InfyCars247Exception {
38        // TODO Auto-generated method stub
39    }
40}
```

Writable

Smart Insert

1:1:0





Project Explorer

- EventRegistration\_Impl
- InfyCars247Application
  - src/main/java
    - com.infy
    - com.infy.api
    - com.infy.dto
    - com.infy.entity
    - com.infy.exception
    - com.infy.repository
    - com.infy.service
      - InfyCars247Service.java
      - InfyCars247ServiceImpl.java
      - InfyCars247ServiceImpl
    - com.infy.utility
  - src/main/resources
    - application.properties
    - InfyCars247\_TableScript.sql
    - InputData.txt
    - log4j2.properties
    - ValidationMessages.properties
  - src/test/java
    - com.infy
      - InfyCars247ApplicationTests.j
      - InfyCars247ApplicationTes
  - Maven Dependencies
  - JRE System Library [JavaSE-11]
  - java
  - log
    - ErrorLog.log
  - src
  - target

InfyCars247ServiceImpl.java

```

27     if (registration==null) {
28         throw new InfyCars247Exception("Service.USER_PROFILE_NOT_FOUND");
29     }
30     Float avgReview = carRepository.getAvgReview(registration.getRegistrationId());
31     if (avgReview<3.5) {
32         throw new InfyCars247Exception("Service.POOR_SELLER_REVIEW");
33     }
34     Car car= new Car();
35     car.setManufacturer(carDTO.getManufacturer());
36     car.setModel(carDTO.getModel());
37     car.setQuotedPrice(carDTO.getQuotedPrice());
38     car.setYearOfManufacturing(carDTO.getYearOfManufacturing());
39     car.setRegistration(registration);
40     car.setFinalPrice(null);
41     car.setReview(null);
42     car = carRepository.save(car);
43     return car.getCarId();
44 }
45
46
47 @Override
48 public RegistrationDTO getUserDetailsByPhone(String phoneNumber) throws InfyCars247Exception {
49     // TODO Auto-generated method stub
50     Registration registration = registrationRepository.findByUserPhone(phoneNumber);
51     if (registration==null) {
52         throw new InfyCars247Exception("Service.USER_PROFILE_NOT_FOUND");
53     }
54     RegistrationDTO registrationDTO= new RegistrationDTO();
55     registrationDTO.setUserEmail(registration.getUserEmail());
56     registrationDTO.setUserName(registration.getUserName());
57     registrationDTO.setUserPhone(registration.getUserPhone());
58     registrationDTO.setRegistrationId(registration.getRegistrationId());
59
60     return registrationDTO;
61 }
62
63 }
64 }
65

```

Quick Access

Writable Smart Insert 1:1:0

Project Explorer

- EventRegistration\_Impl
- InfyCars247Application
  - src/main/java
    - com.infy
    - com.infy.api
    - com.infy.dto
    - com.infy.entity
    - com.infy.exception
    - com.infy.repository
    - com.infy.service
    - com.infy.utility
      - ErrorInfo.java
      - ExceptionHandlerAdvice.java
      - LoggingAspect.java
      - LoggingAspect
  - src/main/resources
    - application.properties
    - InfyCars247\_TableScript.sql
    - InputData.txt
    - log4j2.properties
    - ValidationMessages.properties
  - src/test/java
    - com.infy
      - InfyCars247ApplicationTests.java
      - InfyCars247ApplicationTests
  - Maven Dependencies
  - JRE System Library [JavaSE-11]
  - java
  - log
    - ErrorLog.log
  - src

InfyCars247ServiceImpl.java    LoggingAspect.java

```

1 package com.infy.utility;
2
3 import org.apache.commons.logging.Log;
4
10 @Aspect
11 @Component
12 public class LoggingAspect {
13
14     private static final Log LOGGER = LoggerFactory.getLog(LoggingAspect.class);
15     @AfterThrowing(pointcut = "execution(* com.infy.service.InfyCars247ServiceImpl.*(..))", throwing = "exception")
16     public void logServiceException(InfyCars247Exception exception) {
17         LOGGER.error(exception.getMessage(), exception);
18     }
19 }
20
21

```

Quick Access

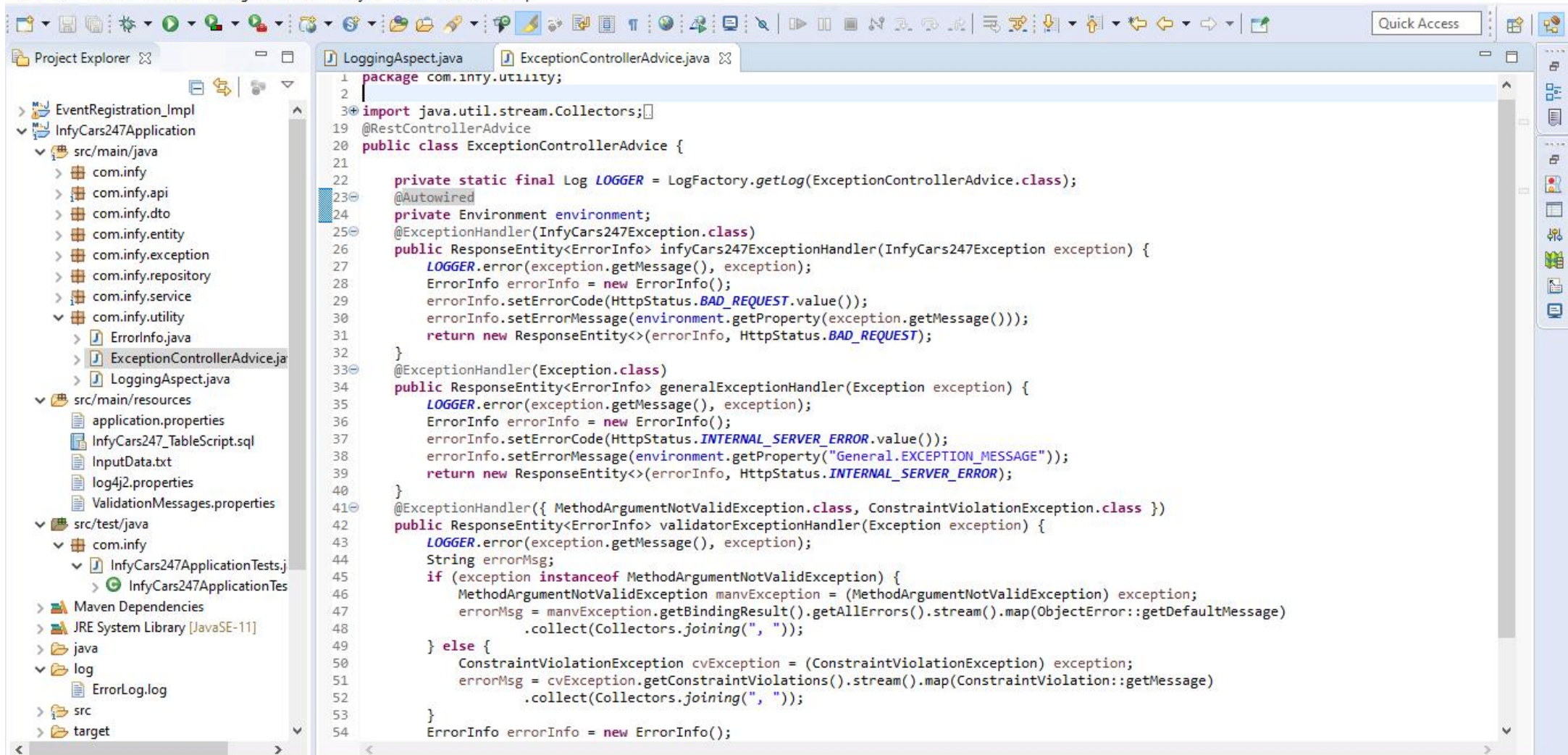
Writable

Smart Insert

1:1:0







```
1 package com.infy.utility;
2
3 import java.util.stream.Collectors;
4
19 @RestControllerAdvice
20 public class ExceptionControllerAdvice {
21
22     private static final Log LOGGER = LogFactory.getLog(ExceptionControllerAdvice.class);
23     @Autowired
24     private Environment environment;
25     @ExceptionHandler({ InfyCars247Exception.class })
26     public ResponseEntity<ErrorInfo> infyCars247ExceptionHandler(InfyCars247Exception exception) {
27         LOGGER.error(exception.getMessage(), exception);
28         ErrorInfo errorInfo = new ErrorInfo();
29         errorInfo.setErrorCode(HttpStatus.BAD_REQUEST.value());
30         errorInfo.setErrorMessage(environment.getProperty(exception.getMessage()));
31         return new ResponseEntity<>(errorInfo, HttpStatus.BAD_REQUEST);
32     }
33     @ExceptionHandler({ Exception.class })
34     public ResponseEntity<ErrorInfo> generalExceptionHandler(Exception exception) {
35         LOGGER.error(exception.getMessage(), exception);
36         ErrorInfo errorInfo = new ErrorInfo();
37         errorInfo.setErrorCode(HttpStatus.INTERNAL_SERVER_ERROR.value());
38         errorInfo.setErrorMessage(environment.getProperty("General.EXCEPTION_MESSAGE"));
39         return new ResponseEntity<>(errorInfo, HttpStatus.INTERNAL_SERVER_ERROR);
40     }
41     @ExceptionHandler({ MethodArgumentNotValidException.class, ConstraintViolationException.class })
42     public ResponseEntity<ErrorInfo> validatorExceptionHandler(Exception exception) {
43         LOGGER.error(exception.getMessage(), exception);
44         String errorMsg;
45         if (exception instanceof MethodArgumentNotValidException) {
46             MethodArgumentNotValidException manvException = (MethodArgumentNotValidException) exception;
47             errorMsg = manvException.getBindingResult().getAllErrors().stream().map(ObjectError::getDefaultMessage)
48                 .collect(Collectors.joining(", "));
49         } else {
50             ConstraintViolationException cvException = (ConstraintViolationException) exception;
51             errorMsg = cvException.getConstraintViolations().stream().map(ConstraintViolation::getMessage)
52                 .collect(Collectors.joining(", "));
53         }
54         ErrorInfo errorInfo = new ErrorInfo();
```

Writable

Smart Insert

2:1:27





Project Explorer

- EventRegistration\_Impl
- InfyCars247Application
  - src/main/java
    - com.infy
    - com.infy.api
    - com.infy.dto
    - com.infy.entity
    - com.infy.exception
    - com.infy.repository
    - com.infy.service
    - com.infy.utility
  - src/main/resources
    - application.properties
    - InfyCars247\_TableScript.sql
    - InputData.txt
    - log4j2.properties
    - ValidationMessages.properties
  - src/test/java
    - com.infy
      - InfyCars247ApplicationTests.j
  - Maven Dependencies
  - JRE System Library [JavaSE-11]
  - java
  - log
    - ErrorLog.log
  - src
  - target
  - verification
  - verificationResources
    - DetailedReport.json
  - HELP.md

LoggingAspect.java ExceptionControllerAdvice.java InfyCars247ApplicationTests.java

```

1 package com.infy;
2
3 import org.junit.jupiter.api.Assertions;
18 @SpringBootTest
19 public class InfyCars247ApplicationTests {
20     @Mock
21     private CarRepository carRepository;
22     @Mock
23     private RegistrationRepository registrationRepository;
24     @InjectMocks
25     private InfyCars247Service infyCars247Service = new InfyCars247ServiceImpl();
26     @Test
27     public void sellCarInvalidUserPhoneNumberTest() {
28         String phoneNumber = "9636245681";
29         Mockito.when(registrationRepository.findByUserPhone(Mockito.anyString())).thenReturn(null);
30         InfyCars247Exception exp = Assertions.assertThrows(InfyCars247Exception.class, () -> infyCars247Service.getUserDetailsByPhone(phoneNu
31         Assertions.assertEquals("Service.USER_PROFILE_NOT_FOUND", exp.getMessage());
32     }
33     @Test
34     public void sellCarInvalidReviewTest() {
35         Float avgReview = 3.0f;
36
37         RegistrationDTO registrationDTO = new RegistrationDTO();
38         registrationDTO.setRegistrationId(1001);
39         registrationDTO.setUserEmail("abc@infy.com");
40         registrationDTO.setUserName("abc");
41         registrationDTO.setUserPhone("9636245681");
42
43         CardDTO cardDTO = new CardDTO();
44         cardDTO.setCarId(1001);
45         cardDTO.setFinalPrice(null);
46         cardDTO.setManufacturer("ABC");
47         cardDTO.setModel("BC");
48         cardDTO.setQuotedPrice(100);
49         cardDTO.setReview(3.0f);
50         cardDTO.setYearOfManufacturing(2000);
51         cardDTO.setRegistrationDTO(registrationDTO);
52
53         Registration reg = new Registration();

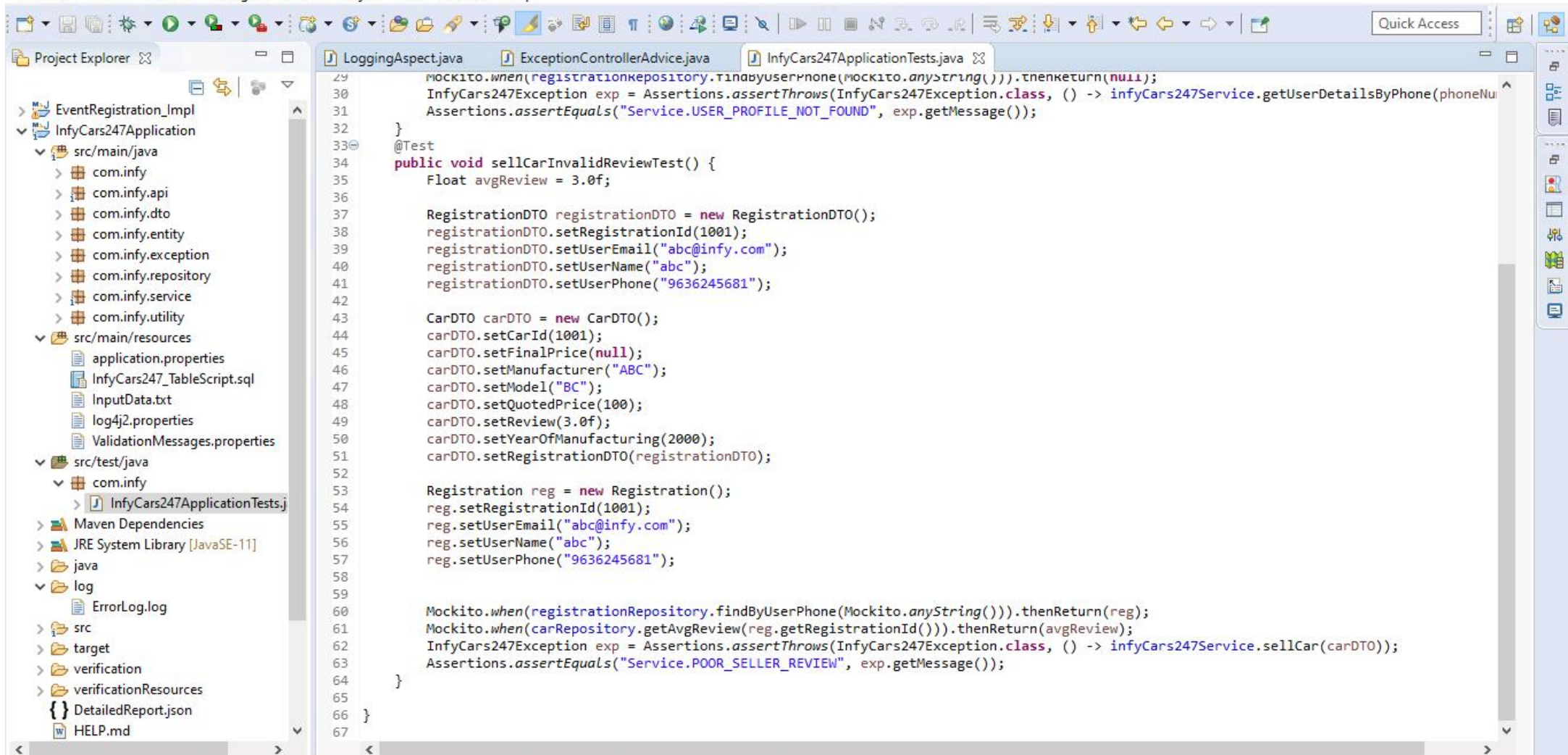
```

Writable

Smart Insert

1:1:0





The screenshot shows the Eclipse IDE interface. On the left is the Project Explorer showing the project structure. The main editor displays the file `InfyCars247ApplicationTests.java` with the following code:

```
29 Mockito.when(registrationRepository.findByUserPhone(Mockito.anyString())).thenReturn(null);
30 InfyCars247Exception exp = Assertions.assertThrows(InfyCars247Exception.class, () -> infyCars247Service.getUserDetailsByPhone(phoneNu
31 Assertions.assertEquals("Service.USER_PROFILE_NOT_FOUND", exp.getMessage());
32 }
33 @Test
34 public void sellCarInvalidReviewTest() {
35     Float avgReview = 3.0f;
36
37     RegistrationDTO registrationDTO = new RegistrationDTO();
38     registrationDTO.setRegistrationId(1001);
39     registrationDTO.setUserEmail("abc@infy.com");
40     registrationDTO.setUserName("abc");
41     registrationDTO.setUserPhone("9636245681");
42
43     CarDTO carDTO = new CarDTO();
44     carDTO.setCarId(1001);
45     carDTO.setFinalPrice(null);
46     carDTO.setManufacturer("ABC");
47     carDTO.setModel("BC");
48     carDTO.setQuotedPrice(100);
49     carDTO.setReview(3.0f);
50     carDTO.setYearOfManufacturing(2000);
51     carDTO.setRegistrationDTO(registrationDTO);
52
53     Registration reg = new Registration();
54     reg.setRegistrationId(1001);
55     reg.setUserEmail("abc@infy.com");
56     reg.setUserName("abc");
57     reg.setUserPhone("9636245681");
58
59     Mockito.when(registrationRepository.findByUserPhone(Mockito.anyString())).thenReturn(reg);
60     Mockito.when(carRepository.getAvgReview(reg.getRegistrationId())).thenReturn(avgReview);
61     InfyCars247Exception exp = Assertions.assertThrows(InfyCars247Exception.class, () -> infyCars247Service.sellCar(carDTO));
62     Assertions.assertEquals("Service.POOR_SELLER_REVIEW", exp.getMessage());
63 }
64
65
66
67
```

The Project Explorer on the left shows the following structure:

- EventRegistration\_Impl
- InfyCars247Application
  - src/main/java
    - com.infy
    - com.infy.api
    - com.infy.dto
    - com.infy.entity
    - com.infy.exception
    - com.infy.repository
    - com.infy.service
    - com.infy.utility
  - src/main/resources
    - application.properties
    - InfyCars247\_TableScript.sql
    - InputData.txt
    - log4j2.properties
    - ValidationMessages.properties
  - src/test/java
    - com.infy
      - InfyCars247ApplicationTests.j
  - Maven Dependencies
  - JRE System Library [JavaSE-11]
  - java
  - log
    - ErrorLog.log
  - src
  - target
  - verification
  - verificationResources
    - DetailedReport.json
  - HELP.md

Writable

Smart Insert

1:1:0

