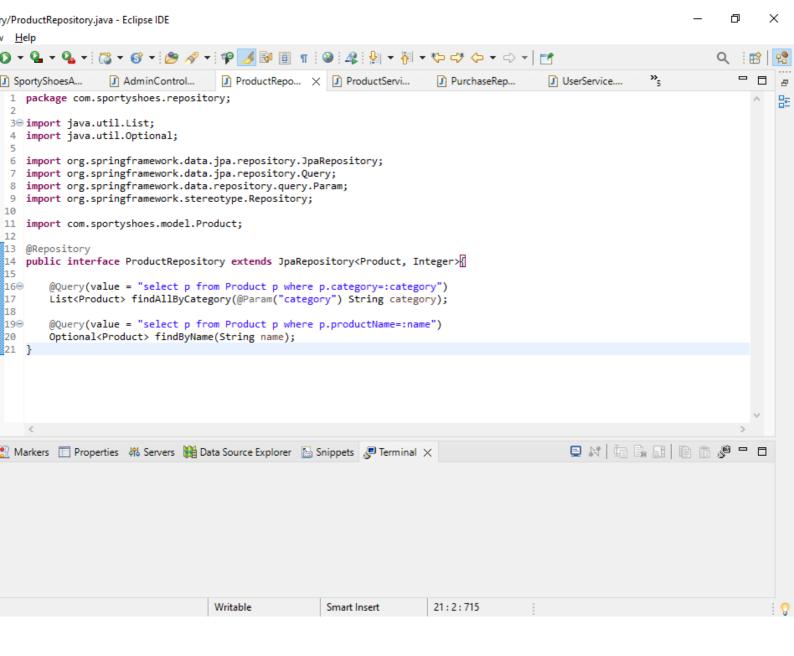
indow <u>H</u>elp s + O + Q + Q + i 👸 + G + B 🔗 + i 🕾 🔗 + i 💬 📝 📵 📵 🔞 🔞 🖓 - i 😭 - i 🖓 i 🖟 i + i 🖰 + i 🖰 - i 🖰 + i 3 - -ProductServi...
PurchaseRep... ☑ SportyShoesA... × ☑ AdminControl... UserReposit... UserService.... package com.sportyshoes; 3⊕ import org.springframework.boot.SpringApplication; [6 @SpringBootApplication public class SportyShoesApplication { 8 public static void main(String[] args) {
 SpringApplication.run(SportyShoesApplication.class, args); 9⊝ 10 11 12 13 } 14 🦹 Markers 🔲 Properties 🚜 Servers 🛍 Data Source Explorer 📔 Snippets 🦑 Terminal 🗙 Writable Overwrite 7:22:191



```
<u>H</u>elp
› ▼ QL ▼ QL ▼ [改 ▼ 양 ▼ | 🤔 🔗 ▼ | 약 🥖 ഈ 세 만 📵 ጠ | @ | 과 | 성 ▼ 전 ▼ 변 ▼ 바 라 수 ▼ 다 → ▼ | ef
                                                                                                                                                                                                                   Q 🔡 🖺
                                                                                                                                                                                                                         - - -
                                                               ☑ ProductRepo...
SportyShoesA...
                             AdminControl...
                                                                                             PurchaseRep... X PurchaseRep...
                                                                                                                                                                   J UserService....
package com.sportyshoes.repository;
3⊖ import java.util.Date;
4 import java.util.List;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.data.jpa.repository.Query;
import org.springframework.stereotype.Repository;

import com.sportyshoes.model.PurchaseReport;
import com.sportyshoes.model.PurchaseReport;

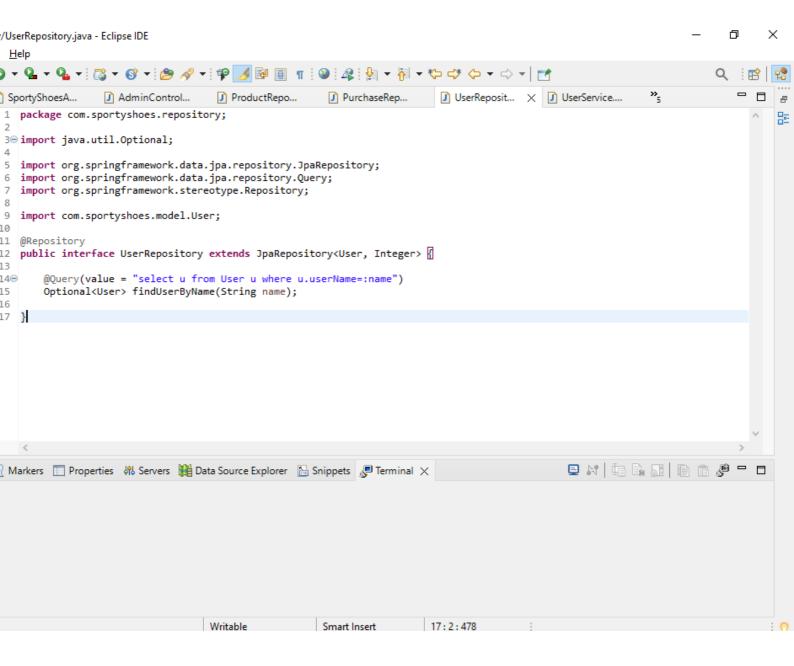
@Repository
public interface PurchaseReportRepository extends JpaRepository<PurchaseReport, Integer> {

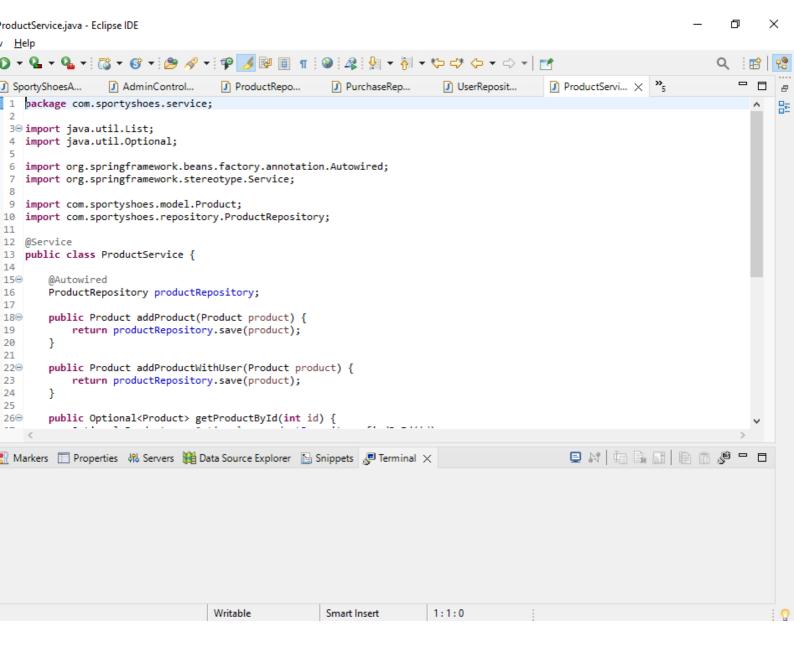
@Query("select pr from PurchaseReport pr where pr.categoryOfProduct=:category")
    List<PurchaseReport> findAllByCategory(String category);

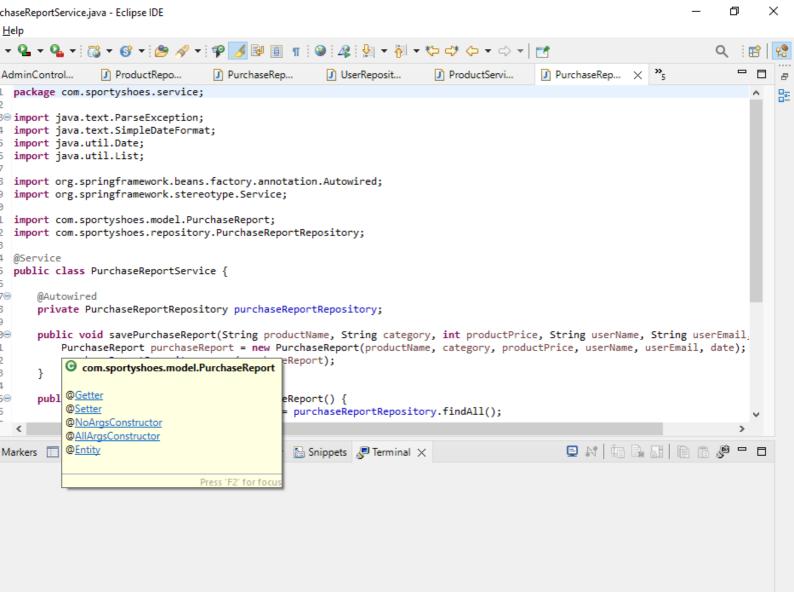
@Query("select pr from PurchaseReport pr where pr.dateOfProductPurchase=:date")
    List<PurchaseReport> findAllByDate(Date date);

@Query("select pr from PurchaseReport pr where pr.dateOfProductPurchase=:date")

21 }
                                                                                                                                                                       Markers 📃 Properties 🚜 Servers 💥 Data Source Explorer 📔 Snippets 🥜 Terminal 🔀
```







```
serService.java - Eclipse IDE
<u>H</u>elp
Q 🔡 🖺
                                                                                                            - - -
                              ProductRepo...
             PurchaseRep...
package com.sportyshoes.service;
30 import java.util.List;
4 import java.util.Optional;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
8
9 import com.sportyshoes.model.User;
10 import com.sportyshoes.repository.UserRepository;
11
12 @Service
13 public class UserService {
14
15⊜
      @Autowired
16
      UserRepository userRepository;
17
18⊜
      public User signUp(User user) {
19
         return userRepository.save(user);
20
21
22⊜
23
      public User saveUserWithProduct(User user) {
         return userRepository.save(user);
24
25
26⊖
      public List<User> allSignedUpUsers() {
Markers 🔳 Properties 🚜 Servers 💥 Data Source Explorer 🔓 Snippets 🥊 Terminal 🗶
```

Х s - Eclipse IDE Q 🔡 🖺 PurchaseRep... UserReposit... ProductServi... PurchaseRep... J UserService.... application.... × »5 1 spring.datasource.url=jdbc:mysql://localhost:3306/phase3project 2 spring.datasource.username=root 3 spring.datasource.password=admin 4 spring.datasource.driver-class-name=com.mysql.jdbc.Driver 5 spring.jpa.properties.hibernate.dialect= org.hibernate.dialect.MySQL5InnoDBDialect 7 spring.jpa.generate-ddl=true 8 spring.jpa.hibernate.ddl-auto= update 9 spring.jpa.show-sql=true 10 spring.jpa.properties.hibernate.format_sql=true 11 logging.level.org.hibernate.type=trace 船 Markers 🔳 Properties 🚜 Servers 🛍 Data Source Explorer 📔 Snippets 🥷 Terminal 🗶