

```

1 import java.util.Scanner;
2
3 public class SquareRootCalculator {
4     public static void main(String[] args) {
5         Scanner scanner = new Scanner(System.in);
6         System.out.print("Enter a number: ");
7
8         try {
9             double number = Double.parseDouble(scanner.nextLine());
10            if (number < 0) {
11                System.out.println("Error: Cannot calculate the square root of a negative number.");
12            } else {
13                System.out.println("Square root: " + Math.sqrt(number));
14            }
15        } catch (NumberFormatException e) {
16            System.out.println("Error: Invalid input. Please enter a numeric value.");
17        } finally {
18            scanner.close();
19        }
20    }
21 }
22

```

input

Enter a number: -16

Error: Cannot calculate the square root of a negative number.

...Program finished with exit code 0

Press ENTER to exit console.

ATMWithdrawalSy... :

```
10 - public InsufficientBalanceException(String message) {
11     super(message);
12 }
13 }
14
15 - public class ATMWithdrawalSystem {
16     private static final int CORRECT_PIN = 1234;
17     private static double balance = 3000;
18
19     public static void main(String[] args) {
20         Scanner scanner = new Scanner(System.in);
21
22         try {
23             System.out.print("Enter PIN: ");
24             int enteredPin = scanner.nextInt();
25
26             if (enteredPin != CORRECT_PIN) {
27                 throw new InvalidPinException("Error: Invalid PIN.");
28             }
29
30             System.out.print("Withdraw Amount: ");
31             double amount = scanner.nextDouble();
32
33             if (amount > balance) {
34                 throw new InsufficientBalanceException("Error: Insufficient balance.");
35             }
36
37             balance -= amount;
```

▼ ↗ 🖨 ⚙ 📌

input

Enter PIN: 1234

Withdraw Amount: 5000

Error: Insufficient balance. Current Balance: 3000.0

Final Balance: 3000.0

...Program finished with exit code 0

Press ENTER to exit console.

```
UniversityEnrollme...
7      super(message);
8  }
9  }
10
11 class PrerequisiteNotMetException extends Exception {
12     public PrerequisiteNotMetException(String message) {
13         super(message);
14     }
15 }
16
17 class UniversityEnrollmentSystem {
18     private static final int COURSE_CAPACITY = 2;
19     private static final Map<String, Integer> courseSeats = new HashMap<>();
20     private static final Map<String, String> coursePrerequisites = new HashMap<>();
21     private static final Map<String, Boolean> completedCourses = new HashMap<>();
22
23     static {
24         courseSeats.put("Advanced Java", 2);
25         courseSeats.put("Core Java", 2);
26
27         coursePrerequisites.put("Advanced Java", "Core Java");
28         completedCourses.put("Core Java", false);
29     }
30
31     public static void enrollStudent(String course) throws CourseFullException, PrerequisiteNotMet
32     if (coursePrerequisites.containsKey(course)) {
33         String prerequisite = coursePrerequisites.get(course);
34
```

input

Enroll in Course: Advanced Java  
Error: PrerequisiteNotMetException - Complete Core Java before enrolling in Advanced Java.

...Program finished with exit code 0  
Press ENTER to exit console.