TASK1:

```
class InvalidNameException extends Exception {
  public InvalidNameException(String message) {
   super(message);
 }
}
class Employee {
  private String firstName;
  private String lastName;
  public Employee(String firstName, String lastName) throws InvalidNameException {
   if (firstName == null || firstName.trim().isEmpty() || lastName == null || lastName.trim().isEmpty()) {
     throw new InvalidNameException("First name and last name cannot be blank.");
   }
   this.firstName = firstName;
   this.lastName = lastName;
 }
  public String getFullName() {
   return firstName + " " + lastName;
 }
  public static void main(String[] args) {
   try {
     Employee emp = new Employee("Thilak", "Rai");
     System.out.println("Employee full name: " + emp.getFullName());
   } catch (InvalidNameException e) {
     System.out.println(e.getMessage());
   }
   try {
     Employee emp2 = new Employee("", "Smith");
   } catch (InvalidNameException e) {
```

```
System.out.println(e.getMessage());
}
```

TASK2:

```
class InvalidAgeException extends Exception {
   public InvalidAgeException(String message) {
      super(message);
   }
}
class Person {
   private int age;
   public Person(int age) throws InvalidAgeException {
    if (age <= 15) {</pre>
```

```
throw new InvalidAgeException("Age must be above 15.");
              }
              this.age = age;
       public int getAge() {
              return age;
       public static void main(String[] args) {
              try {
                     Person person = new Person(20);
                     System.out.println("Person's age: " + person.getAge());
              } catch (InvalidAgeException e) {
                     System.out.println(e.getMessage());
              }
              try {
                     Person <u>person2</u> = new Person(10);
              } catch (InvalidAgeException e) {
                     System.out.println(e.getMessage());
             }
workspace-spring-tool-suite-4-4.27.0.RELEASE - Assignment4/src/Person.java - Spring Tool Suite 4
 File Edit Source Refactor Navigate Search Project Run Window Help
                                                  Dehicle.java Denicle.java DestShapes.java DestShapes.java DestCalculator.java DestCalc
                                                                 4
5 }
                                                                              }
    Assignment1
    Assignment2
                                                            7 class Person [ 8 private int age;
  Assignment4
      > ■ JRE System Library [JavaSE-
                                                                              public Person(int age) throws InvalidAgeException {
   if (age <= 15) {
      throw new InvalidAgeException("Age must be above 15.");</pre>
          Employee.java
                > 🚨 Person.java
                                                                              public int getAge() {
    return age;
                                                                              public static void main(String[] args) {
                                                                                      try {
    Person person = new Person(20);
    System.out.println("Person's age: " + person.getAge());
} catch (InvalidageException e) {
    System.out.println(e.getNessage());
}
Boot Dashboard ×
                                                             <terminated> Person [Java Application] C:\Users\2380262\Downloads\sts-4.27.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.5.v20241023-1957\yire\bin\javaw.exe (Jan 7)
                                                            Person's age: 20
Age must be above 15.
                                                                                                                                                                                                                           Smart Insert
                                                                                                                                                                                                                                                                    35:2:880
```

TASK3:

```
package com.demo.exception;
class EmployeeException extends Exception {
 public EmployeeException(String message) {
   super(message);
 }
}
class Employee {
 private double salary;
 public Employee(double salary) throws EmployeeException {
   if (salary < 3000) {
     throw new EmployeeException("Salary cannot be below 3000.");
   }
   this.salary = salary;
 }
 public double getSalary() {
   return salary;
 public static void main(String[] args) {
   try {
     Employee emp = new Employee(3500);
     System.out.println("Employee's salary: " + emp.getSalary());
   } catch (EmployeeException e) {
     System.out.println(e.getMessage());
   }
```

```
try {
    Employee emp2 = new Employee(2500);
} catch (EmployeeException e) {
    System.out.println(e.getMessage());
}
}
```

```
🧓 workspace-spring-tool-suite-4-4.27.0.RELEASE - Assignment4/src/com/demo/exception/EmployeeException.java - Spring Tool Suite 4

Assignment3

Assignment4

NRE System Library [JavaSE-7]

Str.

8
 9 class Employee {
10 private double salary;
11 public Employee(double
13 if (salary < 3000)
14 throw new Emplo
15 }
16 this.salary = salar
17 }
18
19 public double getSalary
20 return salary;
21 }
22
23 public static void main
24 try {
Employee emp =
   ∨ 🥬 src
      # (default package)
          Employee.java
            Person.java
                                                public Employee(double salary) throws EmployeeException {
   if (salary < 3000) {
      throw new EmployeeException("Salary cannot be below 3000.");
}</pre>

→ 

# com.demo.exception

          EmployeeException.ja
   this.salary = salary;
                                                public double getSalary() {
    return salary;
}
                                                public static void main(String[] args) {
    try {
        Employee emp = new Employee(3500);
 ■ Boot Dashboard ×
5 8 9 8 8 7 E 9 + 7 +
                                                                                                                                                                                              □ Console ×
                                     <terminated> Person [Java Application] C\Users\2380262\Downloads\sts-4.27.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_21.0.5.v20241023-1957\jre\bin\javaw.exe (Jan 7, Employee's salary: 3500.0
Salary cannot be below 3000.
 Type tags, projects, or working set
                                                                                                                                     Smart Insert
                                                                                                                                                              36:6:949
```