Java Exercises on Exceptions

Exercise 1: Write a Java Program to validate the full name of an employee. Create and throw a user-defined exception if firstName and lastName are blank.

```java
class InvalidNameException extends Exception {
   public InvalidNameException(String message) {
      super(message);
   }
}


public class EmployeeNameValidator {
   public static void validateName(String firstName, String lastName) throws InvalidNameException {
      if (firstName == null || firstName.isBlank() || lastName == null || lastName.isBlank()) {
         throw new InvalidNameException("First name or last name cannot be blank.");
      }
   }


   public static void main(String[] args) {
      try {
         validateName("Sybil", "Jacob");
         validateName("", "Tendulkar");
      } catch (InvalidNameException e) {
         System.out.println("Error: " + e.getMessage());
      }
   }
}
```

Output

```
Error: First name or last name cannot be blank.
```

Exercise 2: Validate the age of a person and display a proper message by using a user-defined exception. The age of a person should be above 15.

```
class InvalidAgeException extends Exception {
   public InvalidAgeException(String message) {
      super(message);
   }
}

public class AgeValidator {
   public static void validateAge(int age) throws InvalidAgeException {
      if (age <= 15) {
         throw new InvalidAgeException("Age must be above 15.");
      }
   }

   public static void main(String[] args) {
      try {
         validateAge(23);
         validateAge(13);
      } catch (InvalidAgeException e) {
         System.out.println("Error: " + e.getMessage());
      }
   }
}
```

Output

```
Error: Age must be above 15.
```

Exercise 3: Create an Exception class named as “EmployeeException” (User-defined Exception) in a package named as “com.demo.exception” and throw an exception if the salary of an employee is below 3000. Use the Exception Handling mechanism to handle the exception properly.

```
package com.demo.exception;

class EmployeeException extends Exception {
    public EmployeeException(String message) {
        super(message);
    }
}

public class SalaryValidator {
    public static void validateSalary(double salary) throws EmployeeException {
        if (salary < 3000) {
            throw new EmployeeException("Salary cannot be less than 3000.");
        }
    }

    public static void main(String[] args) {
        try {
            validateSalary(6000);
            validateSalary(2500);
        } catch (EmployeeException e) {
            System.out.println("Error: " + e.getMessage());
        }
    }
}
```

Output

```
Error: Salary cannot be less than 3000.
```