Module: Java + UI + PHP

Course: Core Java

Session 12: Exception Handling

Trainer Notes

1 Session Plan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Time  (min) | Content | Methodology | Trainer  Approach | Learner  Activity | Learning  Outcome  (Bloom's) | Learning  Outcome  (Gardner's) |
| 15 | Basic of exceptions | Reference to  Reading  Material and  Slides | Facilitate,  Elicit  responses | Think,  Respond,  Identify | Remember,  Understand | Intrapersonal,  interpersonal |
| 15 | Try, catch, throw, throws, finally | Reference to  Reading  Material and  Slides | Facilitate,  Elicit  responses | Think,  Respond,  Identify | Remember,  Understand | Intrapersonal,  interpersonal |
| 15 | Demo to handle predefined exceptions | Reference to  Reading  Material and  Slides | Facilitate,  Elicit  responses | Think,  Respond,  Identify | Remember,  Understand | Intrapersonal,  interpersonal |
| 15 | User defined exceptions | Reference to  Reading  Material and  Slides | Facilitate,  Elicit  responses | Think,  Respond,  Identify | Remember,  Understand | Intrapersonal,  interpersonal |
| 25 | Guided Classroom  Activities | Group Activities | Facilitate | Work on guided activities | Remember,  Understand,  Coding | Intrapersonal,  interpersonal |
| 05 | Conclusion | Discussion | Question,  Facilitate,  Guides | Participates,  Recollect  concepts | Remember | Intrapersonal,  interpersonal |

2 Objectives

* Explain Exception
* Differentiate Exception and Error
* Explain Types of exceptions
* Use try-catch-finally construct
* Use throws construct.
* User-defined exceptions.

3 Materials Needed

* Slides

1. Presentation Description

The Facilitator is expected to follow the Presentation Slides as a guideline for the flow of the session.

1. Classroom Activities

To use exceptions to handle errors in the classes of the employee payroll system.

The payroll system of an organization involves calculating the gross salary of each type of employee and the tax applicable to each. The entity classes, their fields and methods are already given in your candidate project. Your task today is to use exceptions to indicate errors in the classes of the employee payroll system.

Classes given to you in the candidate project are: Employee, Trainer, Sourcing, Manager, Organization, SelfEmployed and TaxUtil and IncomeCalculatorInterface. These classes already have fields, constructors and the respective methods. The Trainer, Manager and sourcing classes are sub-classes of Employee.

So far, when you needed to return an error, you have returned an integer such as -1 or a boolean false to indicate errors. The more appropriate way to return and propagate errors is to throw exceptions. Declare and throw appropriate exceptions as described below.

An Exception class called InvalidEmployeeException is given to you. In the Organization class, the getEmployeebyId method throws this exception when an employee who does not exist in the list is given as input. Similar to the above Exception, you will need to define and throw four more exceptions. They are described below.

1. Define an Exception class called InvalidIdException. Throw this exception in the setId method of the employee class when the input to the setId method is less than zero. Throw the same exception from the constructor of Employee class for invalid Id.
2. Define an Exception class called InvalidNameException. Throw this exception in the setName method of the employee class when the input to the setName method is invalid (contains characters other than alphabets and spaces). Throw the same exception from the constructor of Employee class for invalid name.

3. Define an Exception class called InvalidInputException. This exception should be thrown in the following set methods whenever the input to these methods is less than zero. Note that the constructors of the Trainer class and Sourcing class should also throw these exceptions for invalid inputs of the respective fields. \* setBatchCount in Trainer class. \* setEnrollmentTarget in Sourcing class \* setEnrollmentReached in Sourcing class 4. Define an Exception class called InvalidIncomeException. This exception should be thrown in the following set methods whenever the input to these methods is less than zero. Note that the constructors of the classes given below should also throw these exceptions for invalid inputs of the respective fields. \* setBasicSalary of Employee class \* setHRAPer of Employee Class. \* setDAPer of Employee Class. \* setProjectAllowance of Manager class. \* SetPerkPerBatch in Trainer class \* setPerkPerEnrollment in Sourcing class \* setTotalIncome in SelfEmployed class \* setTotalExpense in SelfEmployed class.