Module: Java + UI + PHP

Course: Core Java

Session 20: Generics and Legacy Classes

Trainer Notes

1 Session Plan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Time  (min) | Content | Methodology | Trainer  Approach | Learner  Activity | Learning  Outcome  (Bloom's) | Learning  Outcome  (Gardner's) |
| 15 | Legacy classes & interfaces in Collection API | Reference to  Reading  Material and  Slides | Facilitate,  Elicit  responses | Think,  Respond,  Identify | Remember,  Understand | Intrapersonal,  interpersonal |
| 15 | Demos of legacy classes | Reference to  Reading  Material and  Slides | Facilitate,  Elicit  responses | Think,  Respond,  Identify | Remember,  Understand | Intrapersonal,  interpersonal |
| 15 | Utility Classes (StringBuffer, Date, Calendar,  Properties) | Reference to  Reading  Material and  Slides | Facilitate,  Elicit  responses | Think,  Respond,  Identify | Remember,  Understand | Intrapersonal,  interpersonal |
| 15 | Demos of Utility classes | 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 about Generic Collections
* Explain about Legacy classes & interfaces in Collection API
* Work with Utility Classes
* Use of Some important utility classes(StringBuffer, Date, Calendar, Properties)

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. Guided classroom activities

**Ordering Collections**

* To use ArrayLists to store employees in the Organization class.
* 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 array lists instead of arrays to manager employees in the Organization class.
* Classes given to you in the candidate project are: Employee, Trainer, Sourcing, Manager, Organization, SelfEmployed and TaxUtil. These classes already have fields, constructors and the respective methods. The Trainer, Manager and sourcing classes are sub-classes of Employee.
* Java has extensive libraries for the support of storage, retrieval and sorting of groups of entities. They are called the Collection framework. Among them a commonly used collection class is the ArrayList. Use the array list class as described below in the Organization class of your application.
* Implement the methods given below.
* Change the employeeList field to be of type ArrayList<Employee> instead of an array of employees
* Modify the getter and setter accordingly.
* The field noOfEmployees is removed as it is no longer needed.
* addEmployee(Employee e) : This method adds the given employee to the arrayList of employees in the class. Note that this method has to check if the employee id of the given employee already exists in the list. If it exists, do not add this employee and return -2. If the given employee id does not exist in the listalready, add this employee to the list and return 0. For any other error return -1;
* searchEmployeeByName(string name): This method searches the array of employees to see if an employee of the given name exists. If yes, return the employee object, else return null.
* getEmployeeById(int id): This method should look for an employee with the given ID in the array. If it exists, return the employee, else return null.
* deleteEmployee(int id): Delete the employee if he exists in the list and return true. Otherwise return false.