**Scenario**: Virtual Classroom Manager Programming Exercise

**Overview:** The Virtual Classroom Manager is an innovative platform designed to enhance the online learning experience for educators and students alike. It streamlines the management of virtual classrooms by providing tools for scheduling, attendance tracking, and real-time collaboration.

**Working:**

**Classes:**

* ClassroomManager: This is the main class responsible for handling user input and interactions. It manages a list of Classroom objects and provides methods for adding, removing, listing classrooms, managing student enrollment, and assignments.
* Classroom: This class represents a classroom with a name, a set of enrolled student IDs (studentIds), and a list of assigned assignments (assignments). It provides methods for adding students, scheduling assignments, handling student submissions, and listing enrolled students.

**Command Interface:**

* The program uses a loop to continuously prompt the user for commands.
* Valid commands include:

**add\_classroom [name]:** Creates a new classroom with the provided name.

**remove\_classroom [name]:** Removes a classroom with the provided name.

**add\_student [student\_id] [class\_name**]: Enrolls a student (identified by ID) in a specific classroom.

**schedule\_assignment [class\_name] [assignment\_details]:** Schedules an assignment for a specific classroom with details.

**submit\_assignment [student\_id] [class\_name] [assignment\_details]:** Submits an assignment for a student in a specific classroom with details.

**list\_classrooms:** Lists all available classrooms.

**list\_students [class\_name**]: Lists all students enrolled in a specific classroom.

**exit**: Terminates the program.

**Error Handling:**

* The code incorporates various try-catch blocks to handle potential exceptions:

IllegalArgumentException: Thrown if the user provides invalid arguments for specific commands (e.g., missing arguments).

ArrayIndexOutOfBoundsException: Thrown if user input is malformed (e.g., not enough parts after splitting by space).

Exception: Catches any unexpected exceptions and provides a generic error message.

**Functionality Breakdown:**

* The ClassroomManager class finds classrooms using the findClassroom method that iterates through the classroomList and compares names.
* Adding and removing classrooms update the classroomList.
* Enrolling students involves adding their IDs to the specific classroom's studentIds set.
* Scheduling assignments adds details to the classroom's assignments list.
* Submitting assignments currently just prints a placeholder message, but could be extended to handle actual assignment storage or grading.
* Listing classrooms and students iterates through the respective lists and prints their names/IDs.

Benefits:

 **Enhanced Engagement**: Interactive features like quizzes and discussion boards promote active participation, making learning more enjoyable and effective.

 **Streamlined Management**: Simplifies administrative tasks such as scheduling and attendance tracking, allowing educators to focus more on teaching.

 **Performance Insights**: Comprehensive analytics provide valuable insights into student progress, enabling personalized instruction tailored to individual needs.

Overall, this code demonstrates a basic classroom management system with functionalities for managing classrooms, students, and assignments through user commands.