

## Question: Content Management System for Online Learning Platform

Create an abstract class in Java named `CourseContent` that will serve as the base class for different types of educational content in an online learning platform. This abstract class should encapsulate the common features of course content as follows:

### Attributes:

- `title` (String): the title of the content.
- `description` (String): a brief description of the content.
- `duration` (int): the estimated duration to go through the content, in minutes.
- `difficultyLevel` (String): the difficulty level (e.g., "Beginner", "Intermediate", "Advanced").

### Constructor:

- The class should have a constructor that initializes the `title`, `description`, `duration`, and `difficultyLevel`.

### Abstract Methods:

- Define an abstract method `accessContent()` that simulates accessing the content. This method should print a message indicating that the content is being accessed.
- Define an abstract method `rateContent(int rating)` that allows users to rate the content on a scale from 1 to 5.

### Concrete Methods:

- Implement a method `displayContentDetails()` that prints the title, description, duration, and difficulty level.
- Implement a method `getEstimatedCompletionTime()` that calculates and returns the total time required to complete the content, based on the user's pace (minutes per page or minutes per slide).

### Task:

- Create two subclasses of `CourseContent`: `ReadingMaterial` and `VideoLecture`.
  - For `ReadingMaterial`, implement additional attributes like `numberOfPages` (int), and adjust the `getEstimatedCompletionTime()` method to account for pages.
  - For `VideoLecture`, implement additional attributes like `videoLength` (int), and adjust the `getEstimatedCompletionTime()` method to account for video duration.

- Each subclass should override the `accessContent()` method to include specific actions, like opening a book or playing a video.
- Include a method `updateDescription(String newDescription)` in both subclasses that allows updating the content's description.
- Implement the `rateContent(int rating)` method in both subclasses to keep track of the user's rating. You may store the ratings and calculate an average rating.

### Additional Tasks:

- Implement an interface `Searchable` with a method `search(String keyword)` that allows searching within the content. Integrate this interface with the `ReadingMaterial` class.
- For the `VideoLecture` class, implement an additional method `skipTo(int minute)` that simulates skipping to a specific minute in the video.

### Main Class:

- Write a `main` method in a separate class to demonstrate the creation of instances of `ReadingMaterial` and `VideoLecture`, show how their details can be displayed, updated, accessed, and rated by users.
- Illustrate the search functionality in the `ReadingMaterial` instance and the skip functionality in the `VideoLecture` instance.
- Show the calculation of the estimated completion time based on user pace for both types of content.