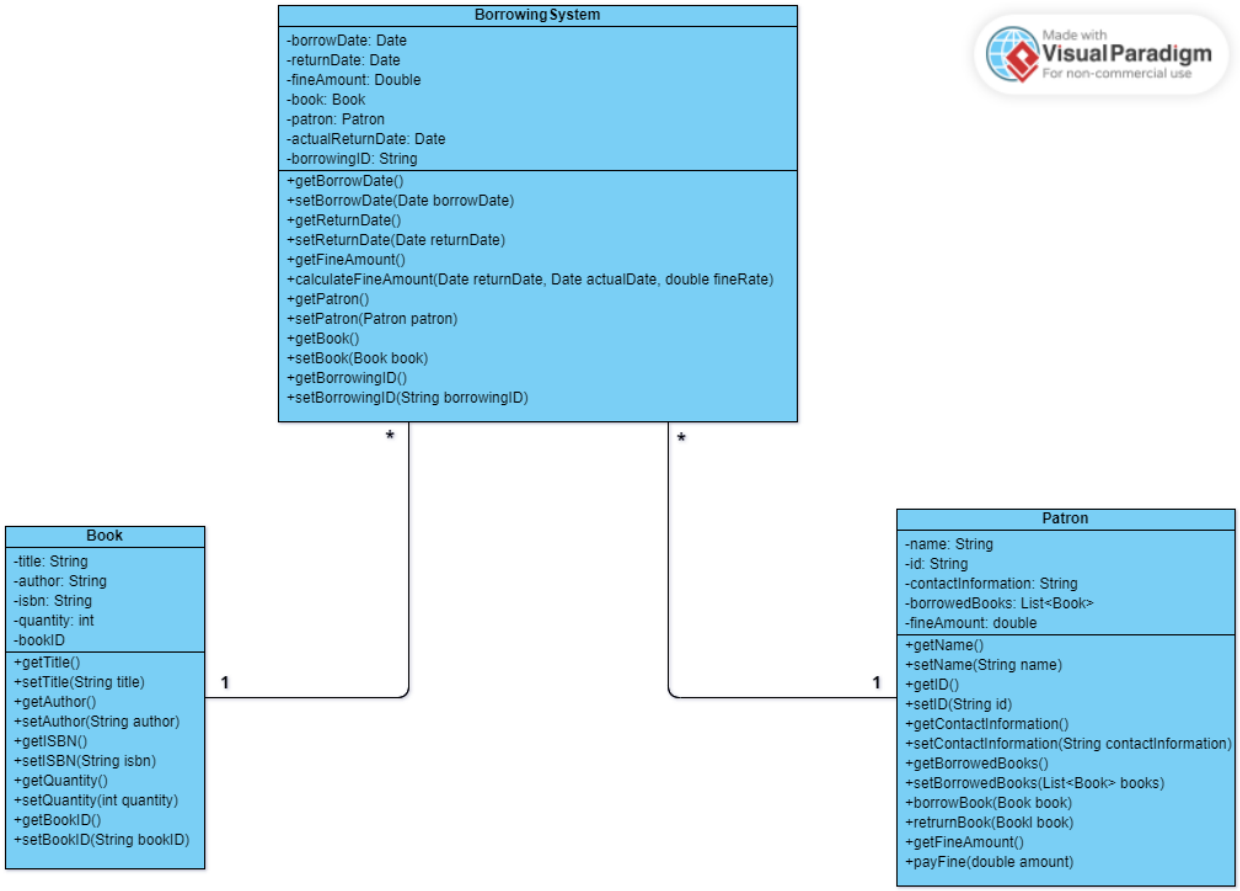
Problem 1: Library Management System

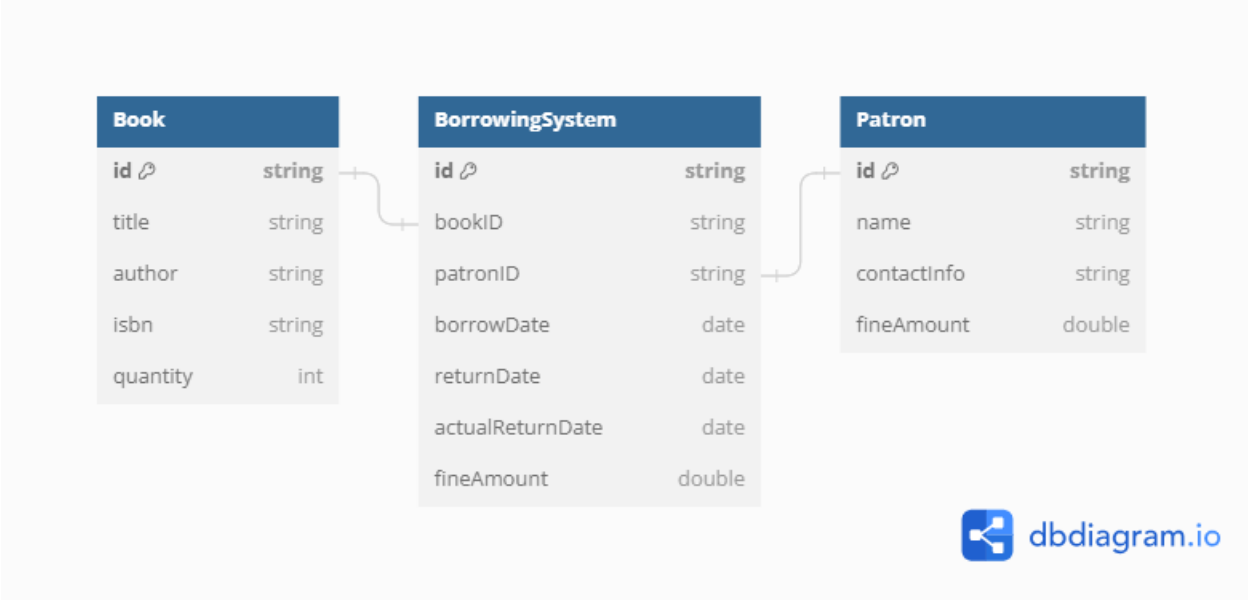
1. Class Diagram:



Relationships between classes:

- Book and Borrowing (One-To-Many Relationship): More than one borrowing record can be connected to a single book. Although each borrowing record is specific to a single borrowing occurrence, they can all be related to the same book.
- Patron and Borrowing (One-To-Many Relationship): Multiple borrowing records may be held by a single patron. Every entry denotes a distinct borrowing occurrence, perhaps involving different books.
- Patron and Book (Indirect Many-To-Many Relationship): A single book can be borrowed by multiple patrons on a long-term basis, and a single patron can borrow multiple books.

2. Database Diagram:



Problem 2: Online Quiz System

1. **Logical Design:**

Question Structure: an object with the following properties:

- id: A unique identifier.
- question: The text of the question.
- options: An array of four choices.
- correctAnswerr: The text of the corect answer

Scoring System:

- initialize with 0 a score variable
- increment the score if the user's answer is correct
- after all questions are answered display the final score

Random Selection Algorithm:

- maintain an array of remaining questions
- when selecting a question, randomly pick from that array
- remove the selected question from the array to avoid repetition
- continue until the array is empty

2. **Algorithm implementation**

```
const [currentQuestion, setCurrentQuestion] = useState(null);
const [remainingQuestions, setRemainingQuestions] = useState(questionsPool);
const [isQuizOver, setIsQuizOver] = useState(false);
const [score, setScore] = useState(0);

useEffect(() => {
  setRemainingQuestions(questionsPool);
  selectRandomQuestion();
}, []);

const selectRandomQuestion = () => {
  if (remainingQuestions.length === 0) {
    setIsQuizOver(true);
    setRemainingQuestions(questionsPool);
    return;
  }
  const randomIndex = Math.floor(Math.random() * remainingQuestions.length);
  const newQuestion = remainingQuestions[randomIndex];
  setCurrentQuestion(newQuestion);

  setRemainingQuestions((prev) => prev.filter((q) => q !== newQuestion));
};

const handleAnswerClick = (isCorrect) => {
  if (isCorrect) setScore(score + 1);
  selectRandomQuestion();
};

const handleResetClick = () => {
  setScore(0);
  selectRandomQuestion();
  setIsQuizOver(false);
};
```

What is the chemical symbol for Gold?

Au

Ag

Fe

Cu

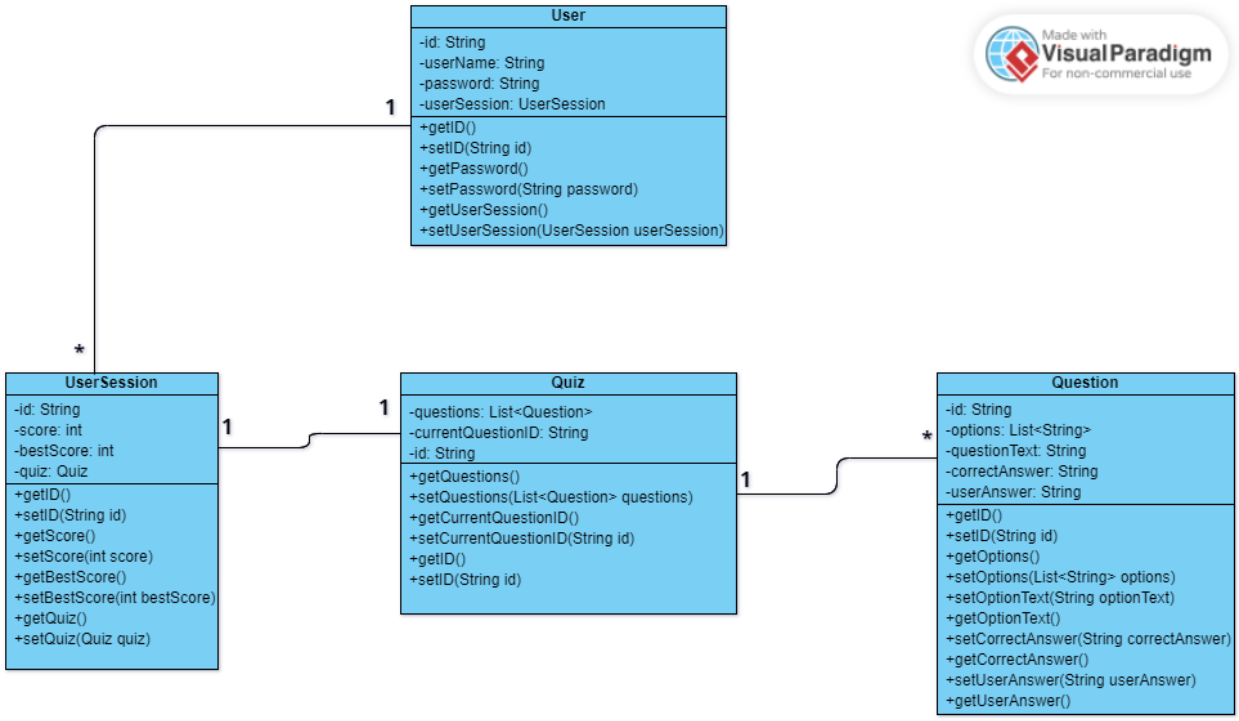
Submit Answer

You got 40/50 answers right!

Reset

3. Class and Database Representation

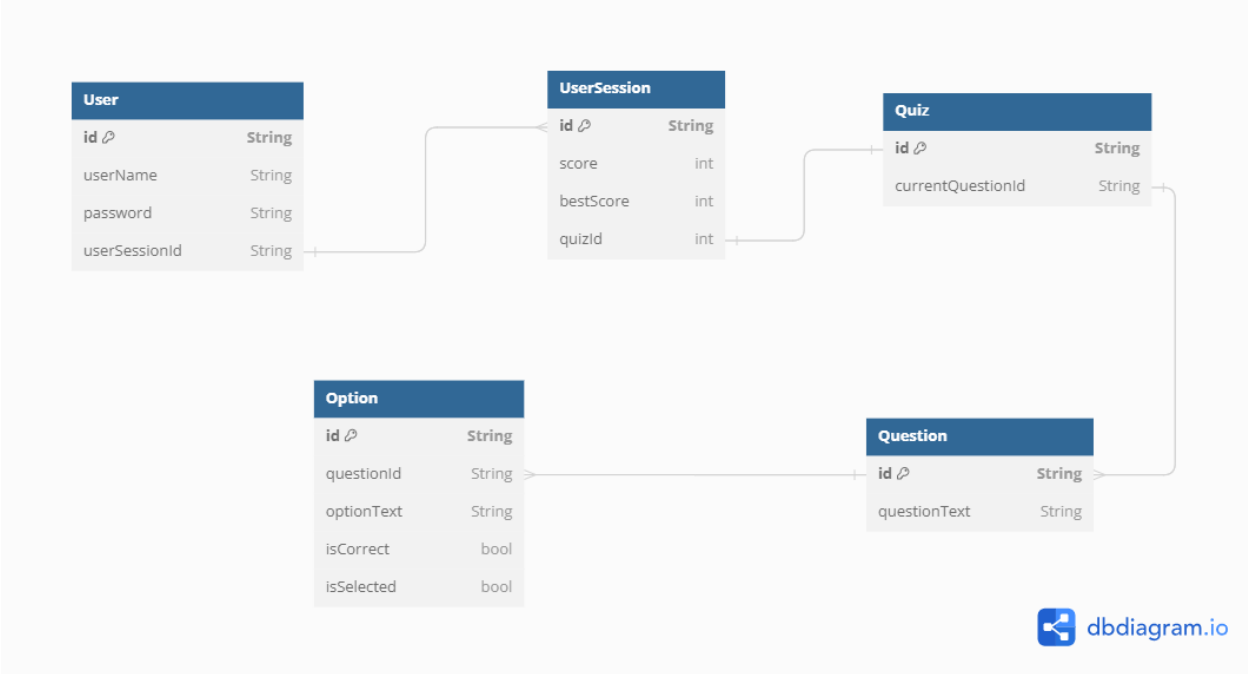
Class Diagram



Relationship between classes:

- User and UserSession (One-To-Many Relationship): One user can have multiple sessions, but each session is associated with only one.
- UserSession and Quiz (One-To-One Relationship): Each session is associated with one quiz, and each quiz is associated with one session.
- Quiz and Question (One-To-Many Relationship): A quiz contains many questions, but each question belongs to only one quiz.

Database Diagram



Relationship between tables:

- User and UserSession (One-To-Many Relationship): Each user can have multiple sessions, but each session is associated with only one user. The userSessionId from User table acts as a foreign key referencing the id in the UserSession table.
- UserSession and Quiz (One-To-One Relationship): Each session is associated with one quiz, and each quiz is associated with one session. The quizId from UserSession table acts as a foreign key referencing the id in the Quiz table.
- Quiz and Question (One-To-Many Relationship): A quiz consists of multiple questions, but each question is part of only one quiz. The currentQuestionId from Quiz table acts as a foreign key referencing the id in the Question table.
- Question and Option (One-To-Many Relationship). Each question can have multiple options, but each option is associated with only one question. The questionId from Option tables acts as a foreign key referencing the id in the Question table.