Programming Exercise Questions on Indexer

Q1) Create a class **`WeekDays`** that stores the names of the days of the week. Use an indexer to access the days using an index (0 for "Sunday", 1 for "Monday", and so on).

Example:

```
WeekDays week = new WeekDays();
Console.WriteLine(week[0]);  // Output: Sunday
week[6] = "Funday";
Console.WriteLine(week[6]);  // Output: Funday
```

Q2) Create a class StudentRecord that stores information about a student, such as Name, Grade, and Age. Use an indexer to access these properties using string keys ("Name", "Grade", "Age").

Example:

```
StudentRecord student = new StudentRecord ();
student["Name"] = "Rajesh";
student["Grade"] = "A";
student["Age"] = "28";
Console.WriteLine(student["Name"]); // Output: Rajesh
```

Q3) Create a class `Library` that holds a collection of books i.e. array of books (represented as strings). Use an indexer to access the books in the library by their position in the array (by using index number).

Example :

```
Library library = new Library();
library[0] = "Let us C";
library[1] = "JAVA black book";
library[1] = " .NET Programming Black Book";
Console.WriteLine(library[0]);  // Output: Let us C
```

Q4) Create a class Temperature that stores temperatures for 7 days. Use a read-only indexer to access the temperatures by day of the week.

Example :

```
Temperature temp = new Temperature();
Console.WriteLine(temp["Monday"]); // Output: 25 (Assume 25 is the temperature for Monday)
```

Q5) Create a class Matrix that stores a 2D array of integers. Implement an indexer that allows accessing elements of the matrix using two indices (row and column).

Example :

```
Matrix matrix = new Matrix(3, 3);  // 3x3 matrix
matrix[0, 0] = 1;
matrix[1, 1] = 2;
Console.WriteLine(matrix[0, 0]);  // Output: 1
```