|  |
| --- |
| namespace \_10.\_SoftUni\_Course\_Planning  {  class Program  {  static void Main(string[] args)  {  List<string> lessons = Console  .ReadLine()  .Split(", ")  .ToList();  string input;  while ((input = Console.ReadLine()) !=  "course start")  {  string[] commands = input  .Split(":")  .ToArray();  switch (commands[0])  {  case "Add":  string addLessonName = commands[1];  AddingLesson(lessons, addLessonName);  break;  case "Insert":  string insertlessonName = commands[1];  int insertIndex = int.Parse(commands[2]);  InsertingLessonToTheIndex(lessons, insertlessonName,  insertIndex);  break;  case "Remove":  string removeLessonName = commands[1];  RemovingLesson(lessons, removeLessonName);  break;  case "Swap":  string firstSwapLesson = commands[1];  string secondSwapLesson = commands[2];  SwapTwoLessons(lessons, firstSwapLesson,  secondSwapLesson);  break;  case "Exercise":  string exercise = commands[1];  string exerciseToBeAdded =  $"{exercise}-Exercise";  AddingExercise(lessons, exerciseToBeAdded,  exercise);  break;  }  }  for (int i = 0; i < lessons.Count; i++)  {  Console.WriteLine($"{i + 1}.{lessons[i]}");  }  }  static void AddingLesson(List<string> lessons,  string addLessonsName)  {  lessons.Add(addLessonsName);  }  static void InsertingLessonToTheIndex(List<string> lessons,  string insertLessonName, int insertIndex)  {  bool isExisting = false;  for (int i = 0; i < lessons.Count; i++)  {  if (lessons[i] == insertLessonName)  {  isExisting = true;  break;  }  }  if (!isExisting)  {  lessons.Insert(insertIndex, insertLessonName);  }  }  static void RemovingLesson(List<string> lessons,  string removeLessonName)  {  bool isExisting = false;  for (int i = 0; i < lessons.Count; i++)  {  if (lessons[i] == removeLessonName)  {  isExisting = true;  break;  }  }  if (isExisting)  {  lessons.Remove(removeLessonName);  }  }  public static void SwapTwoLessons(List<string> lessons,  string firstLessonToSwap, string secondLessonToSwap)  {  bool isFirstExisiting = false;  bool isSecondExisiting = false;  int indexOfFirst = 0;  int indexOfSecond = 0;  for (int i = 0; i < lessons.Count; i++)  {  if (lessons[i] == firstLessonToSwap)  {  isFirstExisiting = true;  indexOfFirst = lessons.IndexOf(firstLessonToSwap);  }  if (lessons[i] == secondLessonToSwap)  {  isSecondExisiting = true;  indexOfSecond = lessons.IndexOf(secondLessonToSwap);  }  }  if (isFirstExisiting && isSecondExisiting)  {  string temp = lessons[indexOfFirst];  lessons[indexOfFirst] = lessons[indexOfSecond];  lessons[indexOfFirst] = temp;  }  }  static void AddingExercise(List<string> lessons,  string exerciseToBeAdded, string exercise)  {  bool isExisting = false;  int indexOfLesson = 0;  for (int i = 0; i < lessons.Count; i++)  {  if (lessons[i] == exercise)  {  isExisting = true;  indexOfLesson = lessons.IndexOf(exercise);  break;  }  }  if (isExisting)  {  lessons.Insert(indexOfLesson + 1,  exerciseToBeAdded);  }  else  {  lessons.Add(exercise);  lessons.Add(exerciseToBeAdded);  }  }  }  } |