public static string codegenerator(int isbn,string coursecode,string bookcode)

{

string ans=isbn+""+coursecode+bookcode;

return ans;

}

public static int random0()

{

Random rand = new Random();

int a = rand.Next(1000, 4000);

return a;

}

public static int random1()

{

Random rand = new Random();

int a = rand.Next(1200,6000);

return a;

}

public static string split(string a)

{

char x=' ';

string []sp= a.Split(x);

string emp = "";

for (int i = 0; i < sp.Length; i++)

{

emp += sp[i][0];

}

return emp.ToUpper();

}

static void Main(string[] args)

{

string[,] arr = new string[2, 2];

for (int i = 0; i < 2; i++)

{

for (int j = 0; j < 2; j++)

{

if (j == 0)

{

Console.WriteLine("Enter course name");

}

if (j == 1)

{

Console.WriteLine("Enter book name");

}

arr[i, j] = Console.ReadLine();

}

}

Console.WriteLine("-----------------------");

int isbn=-1;

for (int i = 0; i < 2; i++)

{

string ccode = split(arr[i, 0]), bcode = split(arr[i, 1]);

if (i==0)

{

isbn=random0();

Console.Write(isbn+" ");

}

else if (i==1)

{

isbn=random1();

Console.Write(isbn+" ");

}

for (int j = 0; j < 2; j++)

{

Console.Write(arr[i,j]+" ");

}

Console.Write(codegenerator(isbn,ccode,bcode));

Console.WriteLine();

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace lab\_1

{

class Queue

{

public int rare = 0;

public int front = 0;

public int[] arr = new int[2];

public Queue()

{ }

public bool enqueue(int a)

{

if (!overflow())

{

arr[rare] = a;

rare++;

return true;

}

else

{

return false;

}

}

public bool overflow()

{

if (rare == arr.Length)

{

return true;

}

else

{

return false;

}

}

public int dequeue()

{

if (!underflow())

{

return front++;

}

else

{

return -1;

}

}

public bool underflow()

{

if (front == 0)

{

return true;

}

else

{

return false;

}

}

public void display()

{

Console.WriteLine("Underflow : "+underflow());

Console.WriteLine("Overflow : " + overflow());

}

}

}

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading;

namespace lab\_1

{

class Program

{

//string[] a = new string[5];

//string[] b = new string[5];

//string[] c = new string[a.Length + b.Length];

//Console.WriteLine("------ Enter Elements of first Array ------");

//for (int i = 0; i < a.Length; i++)

//{

// Console.Write("{0} - Element = ", (i + 1));

// a[i] = Console.ReadLine();

//}

//Console.WriteLine("------ Enter Elements of Second Array ------");

//for (int i = 0; i < b.Length; i++)

//{

// Console.Write("{0} - Element = ", (i + 1));

// b[i] = Console.ReadLine();

//}

//int count = 0;

//for (int i = 0; i < a.Length; i++)

//{

// c[i] = a[i];

// count++;

//}

//for (int i = 0; i < b.Length; i++)

//{

// c[count++] = b[i];

//}

//Console.WriteLine("--------------------------------------------");

//for (int i = 0; i < c.Length; i++)

//Console.WriteLine(c[i]);

// String[] a = new String[5];

// String[] b = new String[5];

// String[] c = new String[a.Length + b.Length];

// int count = 0;

// Console.WriteLine("Elements of Array-1 :");

// for (int i = 0; i < a.Length; i++)

// {

// a[i] = Console.ReadLine();

// }

// Console.WriteLine("Elements of Array-2");

// for (int j = 0; j < b.Length; j++)

// {

// b[j] = Console.ReadLine();

// }

//Console.WriteLine("==============After Merging unsorted form is given below :================");

// for (int i = 0; i < a.Length; i++)

// {

// c[i] = a[i];

// count++;

// }

// for (int j = 0; j < b.Length; j++)

// {

// c[count++] = b[j];

// }

// for (int i = 0; i < c.Length; i++)

// Console.WriteLine((c[i] + " "));

//----------------------------------------------------------------------------

//Console.WriteLine("Enter the Length of Array");

//int n = int.Parse(Console.ReadLine());

//int[] arr = new int[n];

//Console.WriteLine("Subsets Of Elements are");

//for (int i = 1; i < arr.Length; i++)

//{

// for (int j = 1; j < arr.Length; j++)

// {

// Console.WriteLine("[{0},{1}]",i,j);

// }

//}

//------------------------------------------------------------------------------------

// Console.WriteLine("How much Names You want to Enter : ");

// int n=int.Parse(Console.ReadLine());

// string[] arr = new string[n];

// Console.WriteLine("Enter Elements : ");

// for (int i = 0; i < arr.Length; i++)

// {

// Console.Write("{0} = ",i+1);

// arr[i] = Console.ReadLine();

// }

// Console.WriteLine("Array Before Sorting");

// foreach (string item in arr)

// {

// Console.WriteLine(item);

// }

// Quicksort(arr, 0, arr.Length - 1);

// Console.WriteLine("Array After Sorting");

// foreach (string item in arr)

// {

// Console.WriteLine(item);

// }

// }

//public static void Quicksort(string[] A, int LB, int UB)

//{

// if (LB<=UB)

// {

// int pv = partiton(A,LB,UB);

// Quicksort(A, LB, pv - 1);

// Quicksort(A,LB+1,UB);

// }

//}

//public static int partiton(string[]A,int LB,int UB)

//{

// int pv = LB;

// int Lo = LB + 1;

// int hi = UB;

// while (Lo<=UB && hi>=LB)

// {

// if (A[Lo].CompareTo(A[pv])<0)

// {

// Lo++;

// continue;

// }

// if (A[hi].CompareTo(A[pv])>0)

// {

// hi--;

// continue;

// }

// if (Lo<hi)

// {

// string temp1 = A[hi];

// A[hi] = A[Lo];

// A[Lo] = temp1;

// }

// else

// {

// break;

// }

// }

// string temp = A[pv];

// A[pv] = A[hi];

// A[hi] = temp;

// pv = hi;

// return pv;

//------------------------------------------ MERGE SORT -------------------------------------------

//Console.WriteLine("---------------Merge Sort----------------");

//int[] A = new int[] { 11,12,13,14,15};

//int[] B = new int[] { 1,3,5,7,9};

//Console.Write("A = ");

//foreach (int arrA in A)

//{

// Console.Write(arrA+ " ");

//}

//Console.WriteLine();

//Console.Write("B = ");

//foreach (int arrB in B)

//{

// Console.Write(arrB+" ");

//}

//Console.WriteLine("\n");

//int[] c = new int[A.Length + B.Length];

//int i, j, k;

//for (i = 0,j=0,k=0; i < A.Length&& j<B.Length; k++)

//{

// if (A[i]<=B[j])

// {

// c[k] = A[i];

// i++;

// }

// else

// {

// c[k] = B[j];

// j++;

// }

//}

//for (; i < A.Length; i++,k++)

//{

// c[k] = A[i];

//}

//for (; j < A.Length; j++, k++)

//{

// c[k] = A[j];

//}

//foreach (int item in c)

//{

// Console.WriteLine(item+" ");

//}

//-----------------------------------------------------------------

//Console.WriteLine("Enter Your String");

//string a = Console.ReadLine();

//string sp = "Computer Programming";

//char a = ' ';

//sp.Split(a);

//Random generator = new Random();

//int rand = generator.Next(100, 500);

//int x = rand \* 10;

//if (x < 0)

//{

// Console.WriteLine(x \* (-1));

//}

//else

//{

// Console.WriteLine(x);

//}

//char chr = (char)rand;

//Console.Write(chr + " ");

//------------------------------------------------------------------

//Queue solv = new Queue();

////solv.enqueue(2);

////solv.enqueue(2);

////solv.enqueue(2);

//solv.dequeue();

//solv.display();

//------------------------------------------------------------------

public static string codegenerator(int isbn,string coursecode,string bookcode)

{

string ans=isbn+""+coursecode+bookcode;

return ans;

//for (int i = 0; i < 2; i++)

//{

// if (i==0)

// {

// Console.Write(x+"");

// }

// else if(i==1)

// {

// Console.Write(x2 + "");

// }

// for (int j = 0; j < 2; j++)

// {

// Console.Write("-"+arr[i, j] );

// }

// Console.WriteLine();

//} return "";

}

public static int random0()

{

Random rand = new Random();

int a = rand.Next(1000, 4000);

return a;

}

public static int random1()

{

Random rand = new Random();

int a = rand.Next(1200,6000);

return a;

}

public static string split(string a)

{

char x=' ';

string []sp= a.Split(x);

string emp = "";

for (int i = 0; i < sp.Length; i++)

{

emp += sp[i][0];

}

return emp.ToUpper();

}

static void Main(string[] args)

{

string[,] arr = new string[2, 2];

for (int i = 0; i < 2; i++)

{

for (int j = 0; j < 2; j++)

{

if (j == 0)

{

Console.WriteLine("Enter course name");

}

if (j == 1)

{

Console.WriteLine("Enter book name");

}

arr[i, j] = Console.ReadLine();

}

}

Console.WriteLine("-----------------------");

int isbn=-1;

for (int i = 0; i < 2; i++)

{

string ccode = split(arr[i, 0]), bcode = split(arr[i, 1]);

if (i==0)

{

isbn=random0();

Console.Write(isbn+" ");

}

else if (i==1)

{

isbn=random1();

Console.Write(isbn+" ");

}

for (int j = 0; j < 2; j++)

{

Console.Write(arr[i,j]+" ");

}

Console.Write(codegenerator(isbn,ccode,bcode));

Console.WriteLine();

}

}

}

}