|  |
| --- |
| **Day-13 assignment**  **By**  **Bhanu Rama Krishna Prakash Jakkamsetti**  **9/2/2022** |

|  |
| --- |
| 1. .Declare a 2 dimensional array of size (2,2) and initialize using indexes and print the values using nested for loop |
| Code: |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day13\_project1  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: Bhanu rama krishna Prakash jakkamsetti  \* purpose: Declare a 2d array  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  int[,] data = new int[2, 2];  data[0, 0] = 1;  data[0, 1] = 4;  data[1, 0] = 5;  data[1, 1] = 17;  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  Console.Write(data[i, j] + " ");  }  Console.Write("\n");  }  Console.ReadLine();  }  }  } |
| Output: |
|  |

|  |
| --- |
| 2. Declare a 2-D array of size (3,2) and initialize in the same line while declaring and print the values using nested for loop |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day13\_project2  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: Bhanu rama krishna Prakash jakkamsetti  \* purpose:initialize the values for 2d array  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  int[,] data = new int[,] { { 4, 5 }, { 6, 7 }, { 8, 9 } };  for (int i = 0; i < 3; i++)  {  for (int j = 0; j < 2; j++)  {  Console.Write(data[i, j] + " ");  }  Console.Write("\n");  }  Console.ReadLine();  }  }  } |
| Output: |
|  |

|  |
| --- |
| 3. Declare a 2-D array of size (3,3) and print trace of the array |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day13\_project3  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: Bhanu rama krishna Prakash jakkamsetti  \* purpose: add trace of array  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  int sum = 0;  int[,] data = new int[,] { { 5, 6, 7 }, { 8, 9, 10 }, { 11, 12, 13 } };  for (int i = 0; i < 3; i++)  {  for (int j = 0; j < 3; j++)  {  if (i == j)  sum = sum + data[i, j];  }  }  Console.WriteLine(sum);  Console.ReadLine();  }  }  } |
| Output: |
|  |

|  |
| --- |
| 4. Declare a 2-D array of size (2,2) and read values from  user and print the array values. |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day13\_project4  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: Bhanu rama krishna Prakash jakkamsetti  \* purpose: read values from user to declare 2D array  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  int[,] data = new int[2, 2];  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  Console.WriteLine("enter array value");  data[i, j] = Convert.ToInt32(Console.ReadLine());  }  }  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  Console.Write($"{data[i, j]} ");  }  Console.Write("\n");  }  Console.ReadLine();  }  }  } |
| Output: |
|  |

|  |
| --- |
| 5. Declare TWO 2-D arrays of size (2,2) and read values from  user and print the sum of the two matrices. |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day13\_project5  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: Bhanu rama krishna Prakash jakkamsetti  \* purpose: sum of two matrices  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  int[,] data = new int[2, 2];  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  Console.WriteLine("Enter 1st array value");  data[i, j] = Convert.ToInt32(Console.ReadLine());  }  }  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++) { }  }  int[,] data2 = new int[2, 2];  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  Console.WriteLine("Enter 2nd array value");  data2[i, j] = Convert.ToInt32(Console.ReadLine());  }  }  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  Console.Write(data[i, j] + data2[i, j] + " ");  }  Console.Write("\n");  }  Console.ReadLine();  }  }  } |
| Output: |
|  |

|  |
| --- |
| 6. Declare TWO 2-D arrays of size (2,2) and read values from  user and print the product of the two matrices. |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day13\_project6  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: Bhanu rama krishna Prakash jakkamsetti  \* purpose: product of two matrices  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  int[,] data = new int[2, 2];  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  Console.WriteLine("Enter 1st array value");  data[i, j] = Convert.ToInt32(Console.ReadLine());  }  }  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++) { }  }  int[,] data2 = new int[2, 2];  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  Console.WriteLine("Enter 2nd array value");  data2[i, j] = Convert.ToInt32(Console.ReadLine());  }  }  for (int i = 0; i < 2; i++)  {  for (int j = 0; j < 2; j++)  {  Console.Write(data[i, j] \* data2[i, j] + " ");  }  Console.Write("\n");  }  Console.ReadLine();  }  }  } |
| Output: |
|  |

|  |
| --- |
| 7. What is a jagged array What is the benefit of jagged array |
| Jagged array is array of array such that member arrays can be of different sizes. In other words, the length of each array index can differ. The elements of Jagged Array are reference types and initialized to null by default. Jagged Array can also be mixed with multidimensional arrays. Here, the number of rows will be fixed at the declaration time, but you can vary the number of columns. |

|  |
| --- |
| 8. WACP to declare a jagged array and print values |
| code: |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day13\_project7  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: Bhanu rama krishna Prakash jakkamsetti  \* purpose: jagged array and print values  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  char[][] names = new char[3][];  names[0] = new char[] { 'j', 'k' };  names[1] = new char[] { 'b', 'h', 'a', 'n', 'u' };  names[2] = new char[] { 'p', 'r', 'a', 'k', 'a', 's', 'h' };  for (int i = 0; i < 3; i++)  {  for (int j = 0; j < names[i].Length; j++)  {  Console.Write(names[i][j]);  }  Console.Write("\n");  Console.ReadLine();  }  }  }  } |
| Output: |
|  |

|  |
| --- |
| 9. What is Recursion |
| A recursive method is a method which calls itself again and again on basis of few statements which need to be true. |

|  |
| --- |
| 10 . WACP to illustrate usage of Recursion. |
| Code: using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day\_13\_project\_9  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: **Bhanu rama krishna Prakash jakkamsetti**  \* purpose: write recursion  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {      static void Main(string[] args)  {  int fact = 1;  Console.WriteLine("Enter a number:");  int num = int.Parse(Console.ReadLine());  for (int i = 1; i <=num; i++)  {  fact = fact\*i;  }  Console.WriteLine($"fact of given number is:{fact}");  Console.ReadLine();    }  }  } |
| Output: |

|  |
| --- |
| Stack |
| * A stack is a last in, first –out collection of objects. It is used when you need last in, first out access to terms.it is both genic and non-generic type of collection * When you add an item in the list, it is called push. * when you remove an item, it is called pop. |
| Queue |
| * A queue is a first in first out collection of objects. * When you add an item in the list, it is called enqueue. * when you remove an item, it is called dequeue. |

|  |
| --- |
| 11.WACP to illustrate usage of Stack<> |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day13\_project8  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: Bhanu rama krishna Prakash jakkamsetti  \* purpose: stack  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  Stack<int> data = new Stack<int>();  data.Push(10);  data.Push(9);  data.Push(60);  data.Push(14);  Console.WriteLine(data.Count);  Console.WriteLine(data.Pop());  Console.WriteLine(data.Count);    Console.ReadLine();  }  }  } |
| Output: |
|  |

|  |
| --- |
| 12. WACP to illustrate usage of Queue<> |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Day13\_project9  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author: Bhanu rama krishna Prakash jakkamsetti  \* purpose: queue  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  internal class Program  {  static void Main(string[] args)  {  Queue<int> data = new Queue<int>();  data.Enqueue(64);  data.Enqueue(25);  data.Enqueue(54);  Console.WriteLine(data.Count);  Console.WriteLine(data.Dequeue());  Console.WriteLine(data.Count);  Console.ReadLine();  }  }  } |
| Output: |
|  |