using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace Method\_Stubs

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void buttonPress\_Click(object sender, EventArgs e)

{

}

public int sum(int num1, int num2)

{

return num1 + num2;

}

public double average(double num1, double num2, double num3, double num4, double num5)

{

return (num1 + num2 + num3 + num4 + num5) / 5;

}

public int randomSum()

{

Random intNum = new Random();

int ranNumb1 = intNum.Next();

int ranNumb2 = intNum.Next();

return ranNumb1 + ranNumb2;

}

public bool divisibleByThree(int num1, int num2, int num3)

{

int sum = num1 + num2 + num3;

if (sum % 3 == 0)

{

return true;

}

else

{

return false;

}

}

public string lessString(string str1, string str2)

{

if (str1.Length > str2.Length)

{

return str2;

}

else

{

return str1;

}

}

public double largestDoubleVal(double[] doubleValNums)

{

double largestDouble = 0.0;

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

{

if (doubleValNums[i] > largestDouble)

{

largestDouble = doubleValNums[i];

}

}

return largestDouble;

}

public int[] generateIntArray()

{

int[] intArray = new int[50];

Random intNums = new Random();

for (int i = 0; i < 50; i++)

{

int randomIntVal = intNums.Next();

intArray[i] = randomIntVal;

}

return intArray;

}

public bool compareValues(bool bool1, bool bool2)

{

if (bool1 == bool2)

{

return true;

}

else

{

return false;

}

}

public double productReturn(int val1, double val2)

{

double product = val1 \* val2;

return product;

}

public int array2dAverage(int[,] first2dArray)

{

const int ROW = 5;

const int COL = 10;

int[,] values = new int[ROW, COL];

int entries = 0;

for (int row = 0; row < ROW; row++)

{

for (int col = 0; col < COL; col++)

{

entries += first2dArray[row, col];

}

}

int average = (entries) / (ROW \* COL);

return average;

}

}

}