# Introduction

## Tuples

In C#, a tuple is a data structure which organizes data, holding two or more fields of any type.

A tuple is typically created by placing 2 or more expressions separated by commas, within a set of parentheses.

string boast = "All you need to know";

bool success = !string.IsNullOrWhiteSpace(boast);

(bool, int, string) triple = (success, 42, boast);

A tuple can be used in assignment and initialization operations, as a return value or a method argument.

Fields are extracted using dot syntax. By default, the first field is Item1, the second Item2, etc. Non-default names are discussed below.

// initialization

(int, int, int) vertices = (90, 45, 45);

// assignment

vertices = (60, 60, 60);

// return value

(bool, int) GetSameOrBigger(int num1, int num2)

{

return (num1 == num2, num1 > num2 ? num1 : num2);

}

// method argumentint Add((int, int) operands)

{

return operands.Item1 + operands.Item2;

}

Field names Item1 etc. do not make for readable code. The code below shows 2 ways to name the fields of tuples. Note also, in the code below, that var can be used with tuples and the type inferred. This works equally well for tuples with named and unnamed fields.

// name items in declaration

(bool success, string message) results = (true, "well done!");

bool mySuccess = results.success;

string myMessage = results.message;

// name items in creating expression

var results2 = (success: true, message: "well done!");

bool mySuccess2 = results2.success;

string myMessage2 = results2.message;

# Instructions

This exercise has you analyze phone numbers.

You are asked to implement 2 features.

Phone numbers passed to the routines are guaranteed to be in the form NNN-NNN-NNNN e.g. 212-515-9876 and non-null.

## 1. Analyze a phone number

Your analysis should return 3 pieces of data

1. An indication of whether the number has a New York dialing code ie. 212 as the first 3 digits
2. An indication of whether the number is fake having 555 as a prefix code in positions 5 to 7 (numbering from 1)
3. The last 4 digits of the number.
4. Implement the (static) method PhoneNumber.Analyze() to produce the phone number info.
5. PhoneNumber.Analyze("631-555-1234");// => (false, true, "1234")

## 2. Detect if a phone number has a fake prefix code (555)

Implement the (static) method PhoneNumber.IsFake() to detect whether the phone number is fake using the phone number info produced in task 1.

PhoneNumber.IsFake(PhoneNumbers.Analyze("631-555-1234"));

// => true