**Ejercicios 1 a 3**

1. **Indica cuáles de los siguientes identificadores son válidos en Visual C#, y explica por qué en aquellos que no lo sean (los identificadores vienen separados por comas):**

|  |  |  |
| --- | --- | --- |
| **Identificador** | **¿Es válido?** | **Razonamiento** |
| num | Sí | \_ |
| num\_valido | Sí | \_ |
| num contador | No | Carácter no válido: espacio en blanco |
| declaracion renta | No | Carácter no válido: espacio en blanco |
| declaracion\_renta | Sí | \_ |
| hoy | Sí | \_ |
| mañana | No | Carácter no válido: letra ‘ñ’ |
| ayer | Sí | \_ |
| la\_escalera | Sí | \_ |
| ley22/14 | No | Carácter no válido: símbolo ‘/’ |
| num\_2 | Sí | \_ |
| cadena | Sí | \_ |
| carácter | No | Carácter no válido: letra ‘á’ |
| while | No | Palabra reservada ‘while’ |
| Nombre | Sí | \_ |
| NOMBRE | Sí | \_ |

1. **Seleccionar el tipo de datos más adecuado para los siguientes conceptos.**

|  |  |
| --- | --- |
| **Concepto** | **Tipo de dato** |
| Dirección | Cadena de caracteres (string) |
| Cantidad de un préstamo | Entero (int) si no tiene decimales o Real (double) si tiene decimales |
| Número de teléfono | Cadena de caracteres (string) |
| Tipo de interés | Entero (int) si no tiene decimales o Real (double) si tiene decimales |
| Cumpleaños | Cadena de caracteres (string) |
| DNI | Cadena de caracteres (string) |
| Nombre de una persona | Cadena de caracteres (string) |

1. **Analiza qué valores van tomando las variables tras cada una de estas expresiones. Antes de ello indica de qué tipo deberían ser cada una de las variables y decláralas:**

int num1;

int num2;

int num3;

num1 = 10; // num1 tiene valor 10.

num2 = 20; // num2 tiene valor 20.

num3 = num1 + num2; // num3 tiene valor 30.

num3++; // num3 tiene valor 31.

double r1;

double r2;

double r3;

r1 = num1 / num2; // r1 tiene valor 0.

r1 = (double)num1 / (double)num2; // r1 tiene valor 0,5.

r2 = 2.0; // r2 tiene valor 2.0.

r3 = r2 / r1 – 1; // r3 tiene valor 3.0.

r3 = r2 / (r1 - 1); // r3 tiene valor -4.0.

bool l1;

l1 = num1 > num2; // l1 tiene valor FALSE.

l1 = (num1 > num2) && (num2 <= 20); // l1 tiene valor FALSE.

l1 = (num1 > num2) || (num2 <= 20); // l1 tiene valor TRUE.

l1 = (num1 < num2) || (num2 <= 20) && (num1 == 10); // l1 tiene valor TRUE.

l1 = ((num1 < num2) || (num2 <= 20)) && (num1 != 10); // l1 tiene valor FALSE.

l1 = !(((num1 < num2) || (num2 <= 20)) && (num1 != 10)); // l1 tiene valor FALSE.