



Universidad de Costa Rica
Escuela de Ciencias de la Computación e Informática

CI0202 - Laboratorio #1 - 18/Agosto/2017

Identificación de errores de sintaxis, creación de instancias

Objetivos

1. Utilizar la generación intencional de errores de sintaxis para mejorar la efectividad del estudiante al identificarlos.
2. Crear una clase y una instancia de un objeto.

Enunciado

I Parte

El estudiante debe familiarizarse con los diferentes errores de sintaxis que se encontrará. En este ejercicio se pretende que genere intencionalmente los errores de sintaxis más comunes, tales como la falta de un paréntesis, una llave, o palabras reservadas mal escritas entre otros. Además se pretende que observe en cuáles casos no recibirá errores de sintaxis pero el programa no llevará a cabo lo que se solicita.

Para la siguiente lista de preguntas modifique el código de la clase Persona.java para provocar el error, compile el programa y en caso de compilar ejecute el programa. Observe los resultados de cada caso y anótelos. En caso de tener mensajes de error imprima la pantalla de salida del compilador y en caso de compilar pero producir una salida que no es la esperada imprima la pantalla obtenida, e indique en cuáles casos no hay errores y ¿por qué? Deje claro si el programa pudo compilar o no.

Caso 1: ¿Qué pasa si falta una llave { }?

Caso 2: ¿Qué pasa si falta un paréntesis redondo ()?

Caso 3: ¿Qué pasa si falta un paréntesis cuadrado []?

Caso 4: ¿Qué pasa si faltan comillas dobles " en un texto?

Caso 5: ¿Qué pasa si sólo se cambia lo que está entre las comillas?

Caso 6: ¿Qué pasa si falta o está mal escrita una palabra reservada como: public, class, void, static, main, new?

Caso 7: ¿Qué pasa si se omite o se escribe en minúscula un identificador como: String, System?

Caso 8: ¿Qué pasa si se cambia consistentemente todas las apariciones de un nombre identificador como: nombre, edad, setNombre, entre otros, los cuales fueron asignadas por el creador del programa?

Caso 9: ¿Qué pasa si se cambia el nombre de la clase pero no se cambia el nombre del archivo que la contiene?

Caso 10: ¿Qué pasa si se cambia el nombre de la clase y luego se hace corresponder con el nombre del archivo que la contiene?

Caso 11: ¿Qué pasa si se cambia solo la primera aparición de la variable nombre y se sustituye por apellido?

Caso 12: ¿Qué pasa si se cambia solo la primera aparición del nombre del método llamado setNombre y se sustituye por setApellido?

Caso 13: ¿Qué pasa si se elimina el símbolo de suma + en el método imprimir en Persona?

Caso 14: Modifique el método setNombre para que quede de la siguiente forma:

```
public void setNombre(String nombreParam)
{
    nombre = nombreParam;
}
```

Caso 15: Modifique el método get Nombre para que quede de la siguiente forma:

```
public String getNombre()
{
    return this.nombre;
}
```

Caso 16: Modifique el método getNombre para que quede de la siguiente forma:

```
public String getNombre ()
```

```
{  
  
    this.nombre;  
  
}
```

Caso 17: Que sucede si se añade un ; luego de la declaración del método y antes del cuerpo del mismo (justo después del paréntesis que cierra los parámetros que recibe el método y antes de la llave) tal y como se muestra a continuación.

```
public int getEdad ();  
  
{  
  
    return edad;  
  
}
```

II Parte

2.1 Defina su propio tipo Elemento. Para ello especifique una clase Elemento, en el archivo Elemento.java, que tiene los siguientes atributos o propiedades: masa atómica (float), símbolo químico (string), nombre (string), número atómico (int) y electronegatividad (float).

Adicionalmente programe los métodos set y get para cada uno de los atributos anteriores.

2.2 Cree un programa en el archivo ProgramaQuimica.java que instancia dos objetos Elemento.

Al primero asígnele los valores:

Masa: 1.00794,
Símbolo: "H"
Nombre: "Hidrógeno"
Número atómico: 1
Electronegatividad: 2.20

Al segundo objeto agréguele los valores 55.845, "Fe", "Hierro", 26 y 1.83.

Recuerde que para asignar los valores, deberá llamar a los métodos de tipo set programados dentro de la clase Elemento.

Utilizando los métodos get y la instrucción System.out.println(), imprima en consola los datos de las instancias que acaba de crear.

Forma de entrega de la solución

El estudiante debe presentar una explicación de los resultados obtenidos en cada caso acompañado de los posibles mensajes de error.

Para la segunda parte, deberá entregar los archivos con extensión .java.

Los archivos deberán ser entregados en el sitio web Schoology, en la asignación: Laboratorio 1

Fecha de entrega máxima: 22 de agosto del 2017 a las 7 a.m.