

Kimi

1. Explica los Tipos definidos en flowgorithm

Entero

Real

Cadena

Booleano



En Flowgorithm los **tipos definidos** son solo **cuatro primitivos**.

A continuación tienes su **descripción, rango y ejemplos** para que el alumno los use sin confusiones:

1. Entero (`Integer`)

- **Qué es:** Números **sin decimales**, positivos o negativos.
- **Rango aproximado:** $-2\ 147\ 483\ 648 \dots +2\ 147\ 483\ 647$ (32 bits).
- **Ejemplos válidos:** `0` , `-15` , `42` .
- **Uso típico:** contadores, índices, edades, cantidades.

2. Real (`Real`)

- **Qué es:** Números **con decimales** (coma flotante).
 - **Precisión:** \approx 7 decimales válidos (simple precisión IEEE-754).
 - **Ejemplos válidos:** `3.1416` , `-0.005` , `2.0` .
 - **Uso típico:** promedios, áreas, porcentajes, funciones matemáticas.
-

3. Cadena (`String`)

- **Qué es:** Secuencia de **caracteres** (texto).
 - **Longitud máxima:** 2 GB (prácticamente ilimitada para tareas escolares).
 - **Ejemplos válidos:** `"Hola"` , `""` , `"A"` .
 - **Uso típico:** nombres, mensajes, rutas de archivo, menús.
-

4. Booleano (`Boolean`)

- **Qué es:** Solo **dos valores:** verdadero o falso.
- **Literales:** `True` , `False` .
- **Uso típico:** banderas, condiciones, estados de control.



Ejemplo rápido en Flowgorithm

pseudocode

📄 复制

```
Entero: edad
Real: promedio
Cadena: nombre
Booleano: esMayor

edad ← 18
promedio ← 8.5
nombre ← "Ana"
esMayor ← (edad >= 18)
```

Resultado en consola:

📄 复制

```
Ana tiene 18 años y promedio 8.5
¿Es mayor de edad? True
```

2. ¿Existe el concepto de declaración o definición en flowgorithm?



Sí existe, pero **Flowgorithm lo hace invisible**: cuando arrastras el símbolo "Declare" (o cuando exportas a Python/Java) **estás declarando** la variable; no hay bloque separado de "definición" porque el lenguaje es **no explícito** y **único-bloque**.

¿Qué ocupa entonces?

- **Declaración**: elegir nombre, tipo y (opcional) valor inicial en la ventana del símbolo **Declare**.
- **Definición implícita**: Flowgorithm **crea la variable** en ese momento y la inicializa (0, 0.0, "", False) si no le das valor.

Ejemplo visual

1. Arrastras **Declare** → ventana:

复制

Name: edad
Type: Integer
Array: False
Initial Value: (vacío)

↓

2. Al dar *Next* Flowgorithm **declara y define** en un solo paso:

Python 复制

```
# código exportado  
edad = 0
```

Conclusión

No hay bloque separado "*definir*": **declarar = crear + inicializar** en Flowgorithm.

3. Escribe un programa flowgorithm que realice lo siguiente:

- En la función principal escribir el comentario: "Entero e, Real r, Cadena c, Booleano b"
- Le asigne un valor entero a la variable 'e', le asigne un valor real a la variable 'r', una cadena a la variable 'c' y un valor booleano a la variable 'b'



3. Escribe un programa flowgorithm que realice lo siguiente:

a) En la función principal escribir el comentario: "Entero e, Real r, Cadena c, Booleano b"

b) Le asigne un valor entero a la variable 'e', le asigne un valor real a la variable 'r', una cadena a la variable 'c' y un valor booleano a la variable 'b'



Aquí tienes el contenido completo del archivo `tipos_demo.fprg` .

Guárdalo y ábrelo en Flowgorithm 4.5; ejecuta paso a paso para ver los valores en la ventana *Watch*.

```
<?xml version="1.0"?>
```

```
<flowgorithm fileversion="4.5">
  <algorithm name="tipos_demo">
    <attributes>
      <attribute key="isMain" value="true"/>
    </attributes>
    <function name="Main" type="None">
      <parameters/>
      <body>
        <declare name="e" type="Integer" array="False"/>
        <declare name="r" type="Real" array="False"/>
        <declare name="c" type="String" array="False"/>
        <declare name="b" type="Boolean" array="False"/>
        <comment text="Entero e, Real r, Cadena c, Booleano b"/>
        <assign variable="e" expression="25"/>
        <assign variable="r" expression="12.5"/>
        <assign variable="c" expression="&quot;Hola Flowgorithm&quot;"/>
        <assign variable="b" expression="True"/>
        <output expression="&quot;e = &quot; & & e" newline="True"/>
        <output expression="&quot;r = &quot; & & r" newline="True"/>
        <output expression="&quot;c = &quot; & & c" newline="True"/>
        <output expression="&quot;b = &quot; & & b" newline="True"/>
      </body>
    </function>
  </algorithm>
</flowgorithm>
```

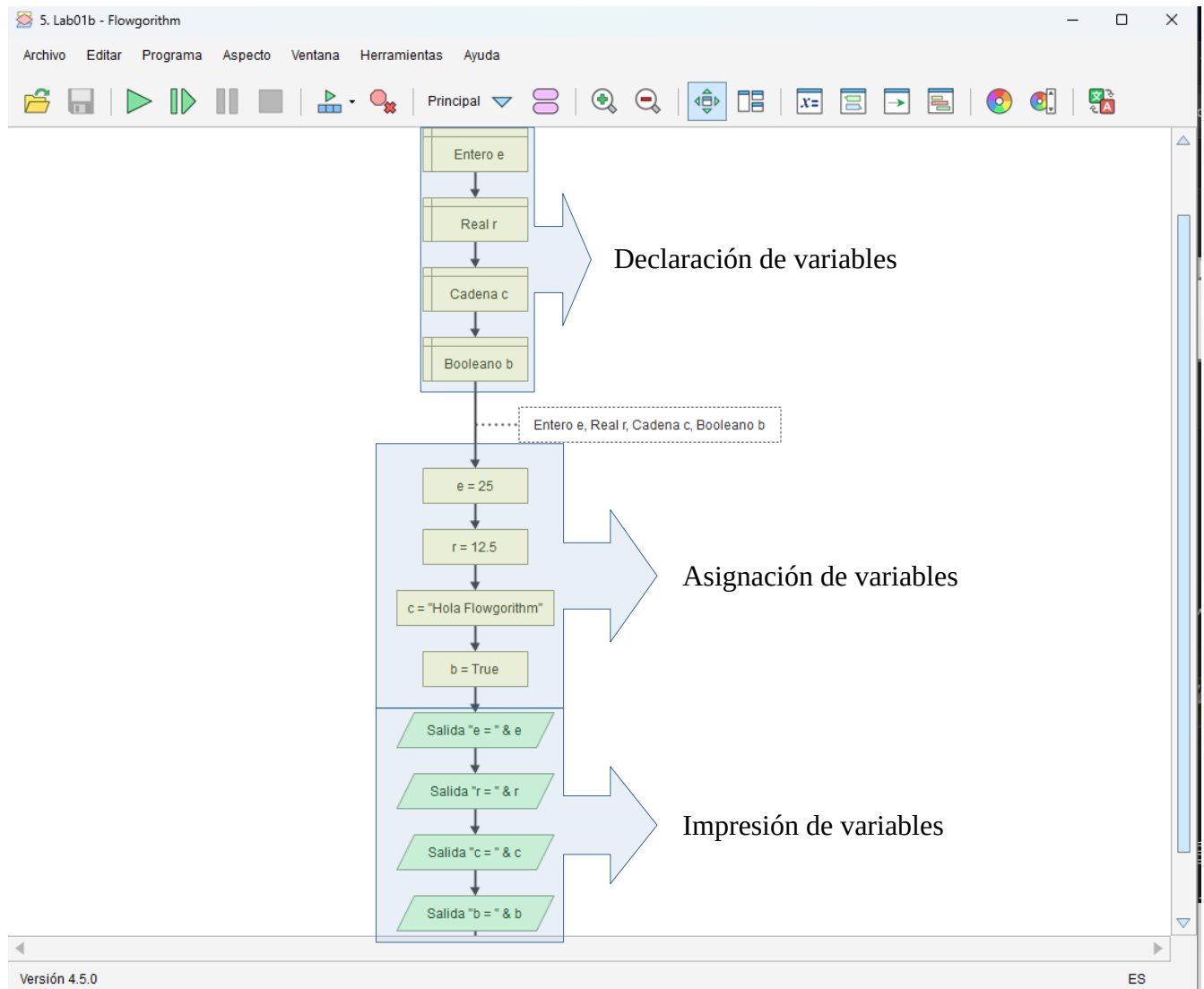
Salida en consola:

复制

```
e = 25  
r = 12.5  
c = Hola Flowgorithm  
b = True
```



Intercambiar la sección <body> de un nuevo proyecto por la sección <body> del código anterior y luego abrir el código modificado con flowgorithm



Ejecución

