

## ¿Cómo funcionan los modificadores privados, públicos y protegidos en Ruby?

En Ruby, los modificadores de acceso (`private`, `public` y `protected`) controlan la visibilidad de los métodos en una clase. Estos modificadores determinan qué métodos son accesibles desde diferentes partes del programa. cuando declaramos un método si no indicamos alguno de estos modificadores se define de tipo `public`.

- **Private:** Cuando un método se declara como privado, solo puede ser invocado por otros métodos dentro de la misma clase. Estos métodos no pueden ser llamados desde fuera de la clase, ni siquiera por objetos de la misma clase. Para declarar un método como privado, se utiliza la palabra clave `'private'`.

### Ejemplo sintaxis:

```
class Modificadores

  private

  def private_Ruby
    # Código del método privado
  end
end
```

- **Public:** Los métodos públicos son accesibles desde cualquier parte del programa, ya sea desde dentro de la clase o desde fuera. Por defecto, todos los métodos en Ruby son públicos a menos que se correspondan a lo contrario. No es necesario utilizar una palabra clave para declarar un método como público.

### Ejemplo de sintaxis:

```
class Modificadores

  def public_Ruby
    # Código del método público
  end
end
```

- **Protected:** Los métodos protegidos solo pueden ser invocados por objetos de la misma clase o de una subclase. Estos métodos no pueden ser llamados desde fuera de la clase o subclase. Para declarar un método como protegido, se utiliza la palabra clave 'protected'.

**Ejemplo sintaxis:**

```
class Modificadores

  protected

  def protected_Ruby

    # Código del método protegido

  End

End
```

**¿Cómo puedo hacerlo para establecer el método de inicializar, para que funcione recibiendo argumentos?**

```
class Metodo

  def initialize(arg1, arg2 = "valor agregado")

    @argumento1 = arg1

    @argumento2 = arg2

  End

End

# Crear objeto con argumento opcional

Objeto1 = Metodo.new("valor1", "valor2")

puts objeto1
```

En Ruby, puede definir el método de inicialización (initialize) para aceptar argumentos opcionales utilizando argumentos con valores predeterminados. Esto te permite llamar al método de inicialización con o sin argumentos

