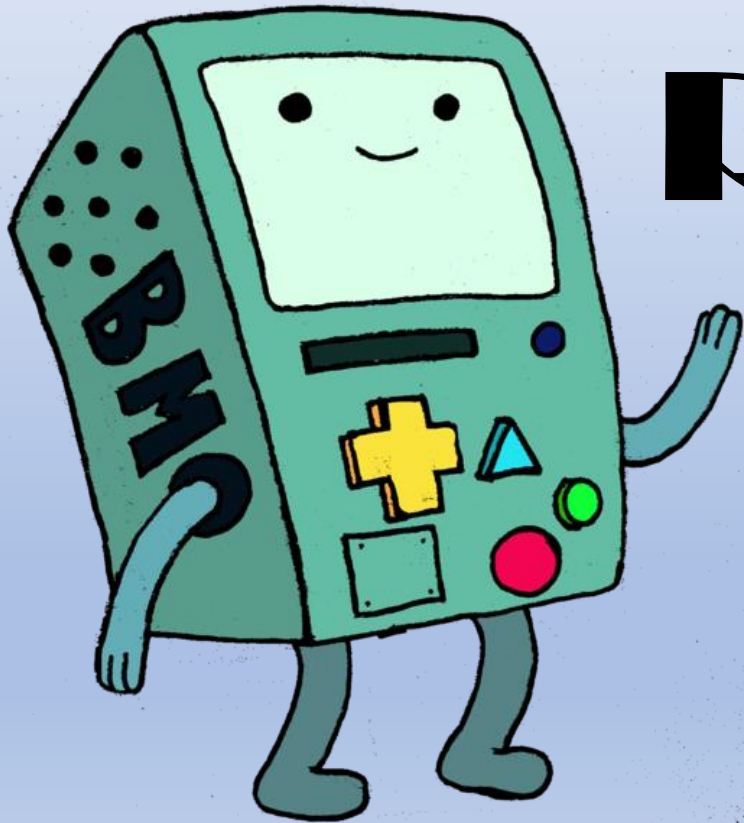


# Seguimos con ...

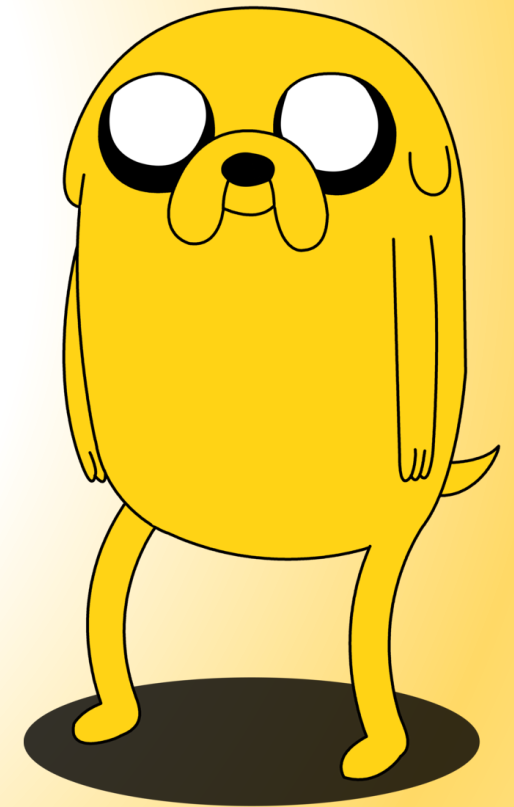


# REGISTROS

# **Antes de entrar en el tema volvemos a pensar en abstracción**

**Si tuvieran que describir a un perrito  
¿En cosas pensarían?**

- ❖ **Características importantes de un perrito.**
- ❖ **Dependen de a quién le pregunte (no es lo mismo preguntárselo a un veterinario que a un nene).**
- ❖ **Va a pasar lo mismo cuando tratemos de resolver problemas con código.**
- ❖ **Los aspectos importantes de nuestros elementos van a depender del problema.**



# ¿Como podríamos llevar esta abstracción a código?



Vamos a poder realizarlo mediante  
la creación de nuestros propios  
tipos de datos

Esto va a ser posible mediante los tipos de datos `STRUCT` O `REGISTROS`  
Para poder implementarlo vamos a usar la siguiente sintaxis:

```
typedef struct nombre {  
    tipo_de_dato variable1;  
    tipo_de_dato variable2;  
    tipo_de_dato variableN;  
} nombre_t;
```

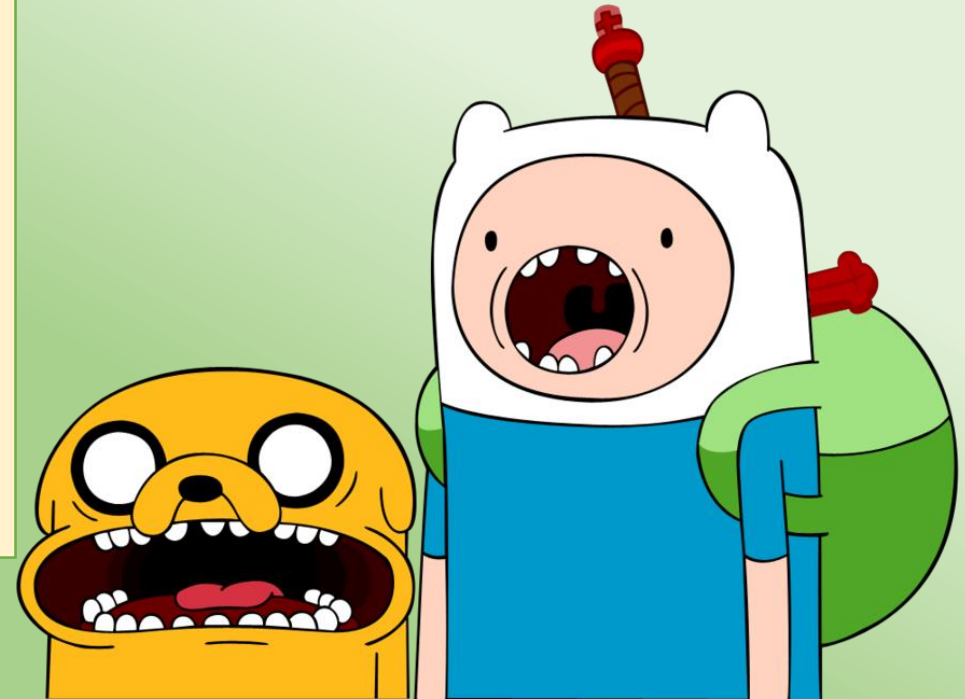
Lo que esta dentro del struct  
le diremos campos internos  
a los cuales podremos acceder.

# ¿Como podemos acceder a los campos internos?

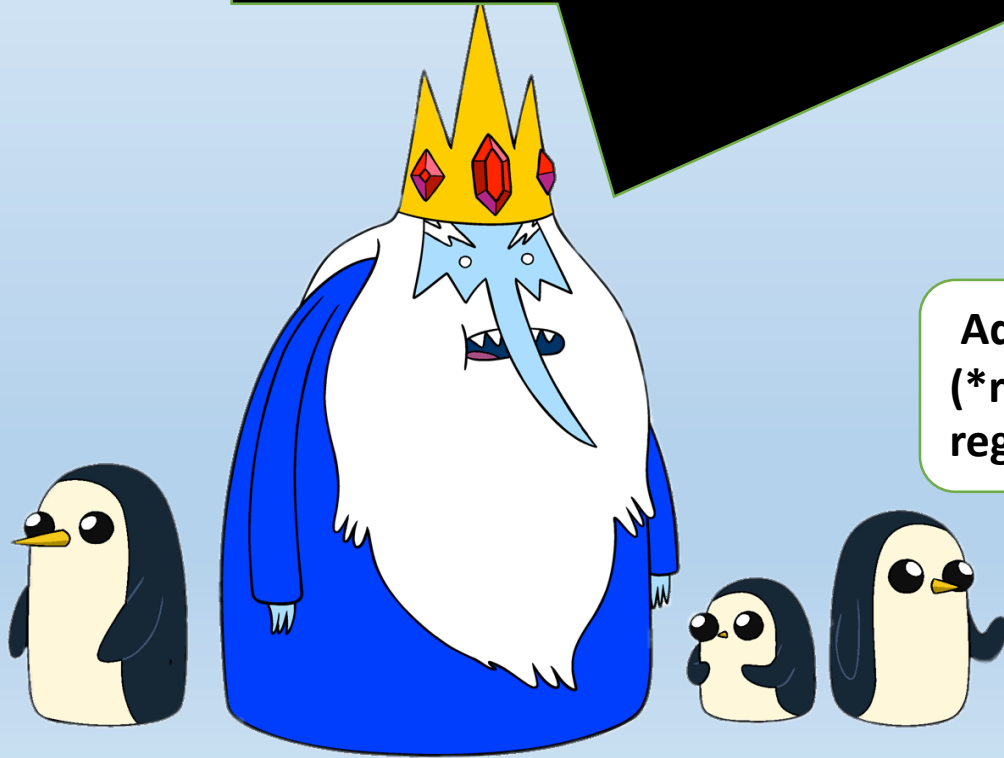
Para acceder a un campo usamos el “operador punto”.

Ejemplo:

```
variable nombre;  
variable.campo = valor;
```



**Es importante saber que los registros se pasan por valor. Si queremos modificarlo adentro de una**



Además en vez de hacer  
(\*registro).campo podemos hacer  
registro->campo y es lo mismo

**¿PREGUNTAS?**

