



“Toda reforma interior e toda mudança para melhor dependem exclusivamente da aplicação do nosso próprio esforço.”

Immanuel Kant

“Em C, os tipos primitivos são números”

```
char var = 42;  
int var = 42;  
long var = 42;  
float var = 42;  
double var = 42;
```

```
if (valor1 > valor2)  
if (valor1 >= valor2)  
if (valor1 < valor2)  
if (valor1 <= valor2)  
if (valor1 == valor2)  
if (valor1 != valor2)
```



Typedefs

```
typedef tipo_existente tipo_que_eu_quero;  
  
typedef int inteiro;
```



Ponteiros

```
int var = 42;
```

```
int *ptr = 42;
```



Atribuição de ponteiros

```
int var = 42;
```

```
int *ptr = &var;
```



Formas de usos de variáveis

```
int var;  
var = 42;  
var;
```

```
int *ptr;  
ptr = &var;  
ptr;  
*ptr; ptr[0];
```



Arrays vs Ponteiros

```
char ptr_c[] = {48, 50, -117, 100, 100, 40, 0, -1};  
int  *ptr_i = (int *)ptr_c;  
long *ptr_l = (long *)ptr_c;
```

```
char *ptr_c = {48, 50, -117, 100, 100, 40, 0, -1};  
int  *ptr_i = (int *)ptr_c;  
long *ptr_l = (long *)ptr_c;
```

Referência de tipos primitivos



<https://www.gnu.org/software/gnu-c-manual/gnu-c-manual.html#Primitive-Types>

Referência de Typedef



<https://www.gnu.org/software/gnu-c-manual/gnu-c-manual.html#The-typedef-Statement>

Referência de Ponteiros



<https://www.geeksforgeeks.org/c/c-pointers/>



Avalie nosso conteúdo!

