Const keyword

The const keyword is used to declare that a variable, function, or object is immutable. i.e, its value cannot be changed after initialization.

for Ex.

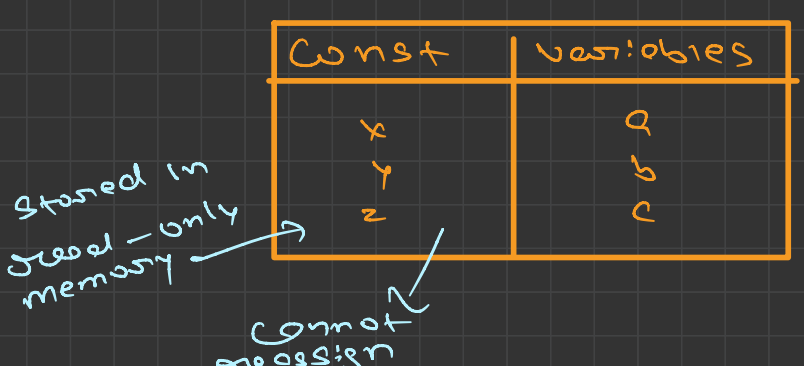
```
const int x = 5; ← cannot modify the value of x later in program
```

```
x = 7; ✗
```

Any attempt to modify the value will result Compilation error.

You can declare a function as const, which means that it doesnot modify the state of the object it is called on.

Compilers may be able to store const variables in read-only memory, which can result in faster access time.



```
const int x = 10;
```

`x = 12;` → Error ⇒ Expression must be a modifiable lvalue =?

lvalue ⇒ variable having memory location

Eg ⇒ `int x, char y.`

rvalue ⇒ variable doesnot value memory location

Eg ⇒ `const int x, int &a = b`

↓

no memory location
a is a name of b

How to change Const variables

```
const int x = 10;

int *p = &x;
*p = 5;
cout << x;
```

Works in old compilers

Const with Pointers

```
const int *a = new int(2);
cout << *a;
```

initialize here

```
*a = 20;
cout << *a;
```

throw error

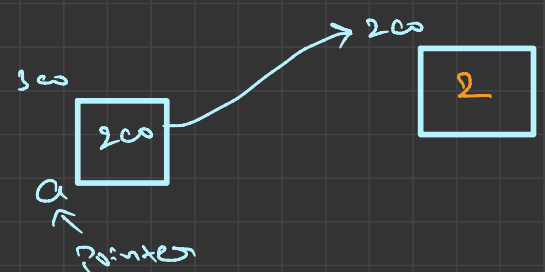
we cannot change the data by dereferencing the pointer

Both are same

const data, non-const pointer

```
int const *a = new int(2);
```

```
Const int *a = new int;
```



→ data/content of a pointer is constant

→ pointer itself can be reassigned

```
const int *a = new int(2);
cout << *a;
```
⇒ 2

```
int b = 20;
a = &b;
cout << *a;
```
⇒ 20

we change the pointer to point different memory location

Pointer itself can be reassigned

Const Pointer, but non-const pointer

```
int *const a = new int(2);
cout << *a;        => 2

*a = 20;
cout << *a;        => 20
```

```
int b = 30;
a = &b;
cout << *a;
```

} Throw error because Pointer is constant

Const pointer, const data

```
const int *const a = new int(10);
cout << *a;
```

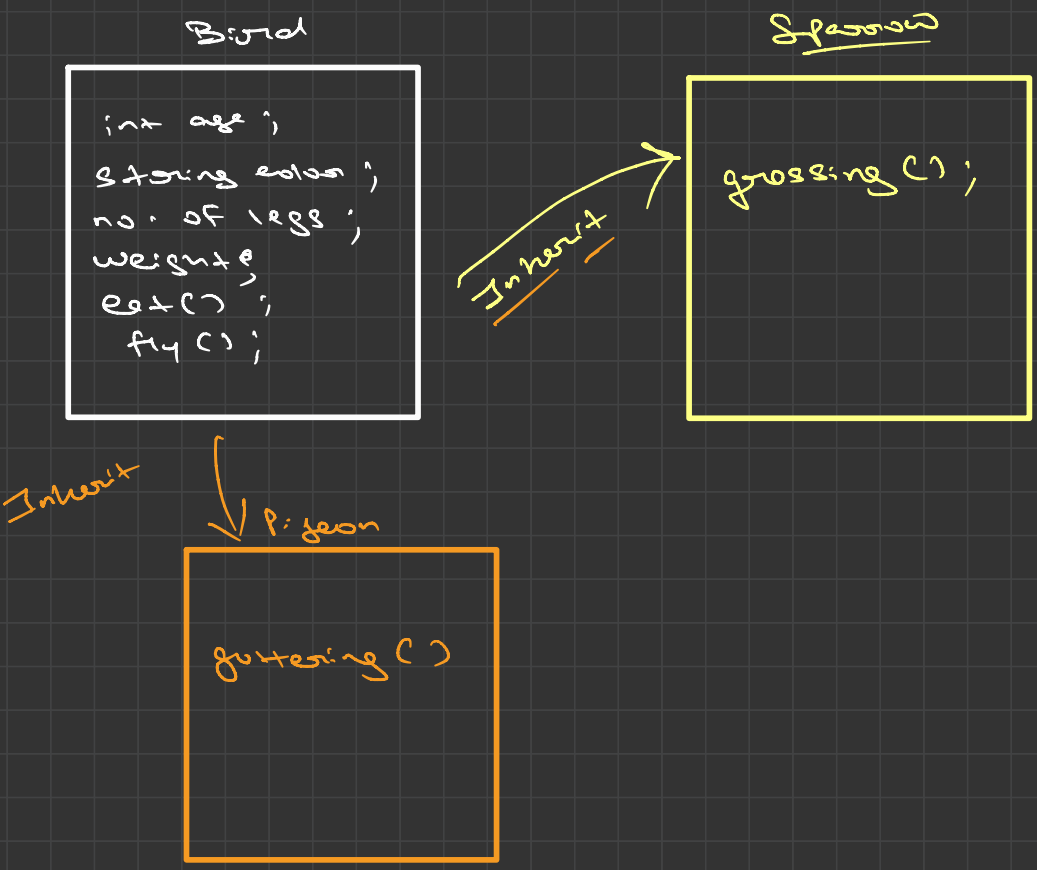
```
*a = 30;
```

```
int b = 10;
a = &b;
```

} Throw error both are constant

If I write const before '*', then content will be const

Const with functions / Class methods


Bird
int age ;
String color ;
no. of legs ;
weight ;
eat() ;
fly() ;
Sparrow
grossing() ;
Inherit
Inherit
Pigeon
guttering()

Syntax

class Child_name : ..... Parent_name {

mode of inheritence

};