

Smart Pointers

[auto memory
management]



garbage collector

→ Can we directly init a
variable in heap in C++??

↳ ^{no,} we have to always use pointers

free / delete → they are working on ptrs.

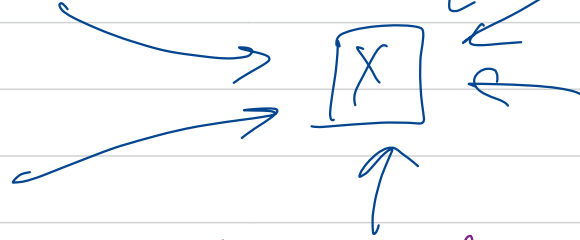
- ① Unique Pointer
 - ② Shared Pointer
 - ③ Weak Pointer
- 3 types of Smart Pointers

if we have created an object using unique-pts,
then no other pointer can point to it.

shared-pointer gives you access to other objects

→ Destructor → Storcks → this is a function which
is called finally before the storck is
removed from memory?

Smart ptrs → reference count → 0 → destroy it



weak ptr → doesn't maintain reference count

~~dangling~~

↖
↙

weak ptr
Solve the

problem of a dayly ptr

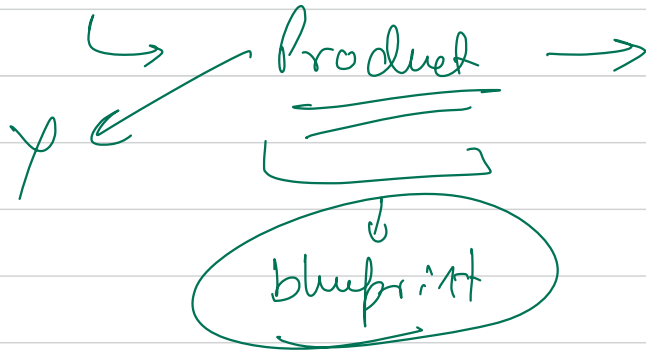
empired()

↳ any piece of code that is common for all

products

features

100%



name
desc
cost
discount

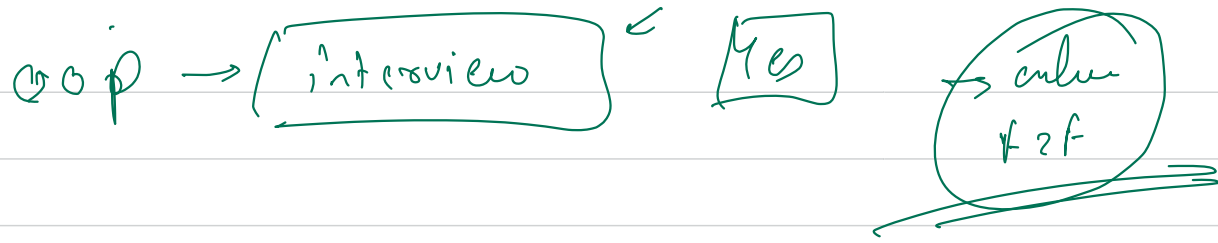
shirt

object

some real life entity

macbook

iphone

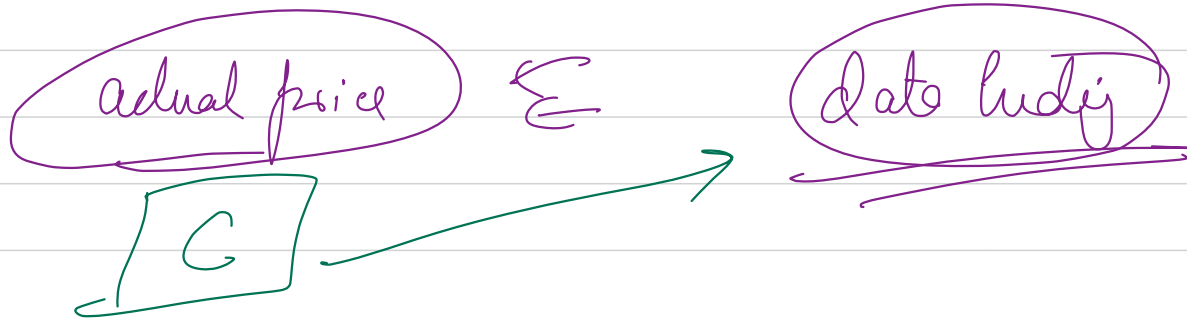


objects \rightarrow real life entities / also called as instances

Class \rightarrow is a blue print of a real life

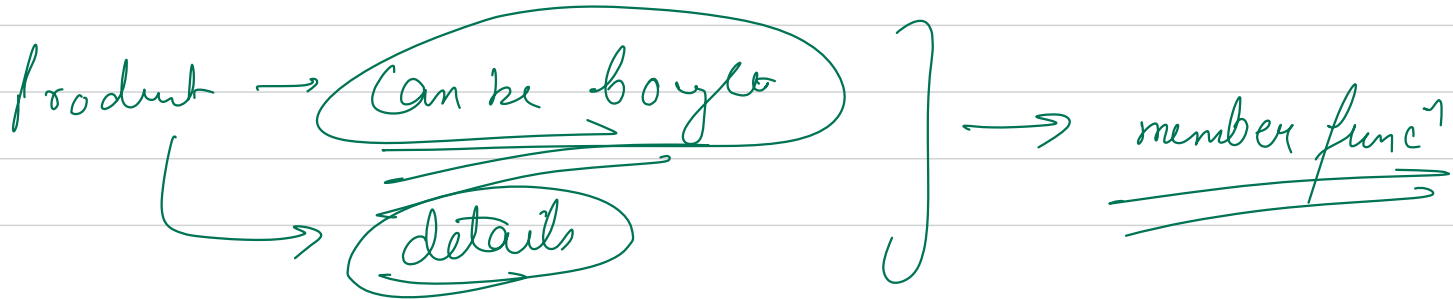
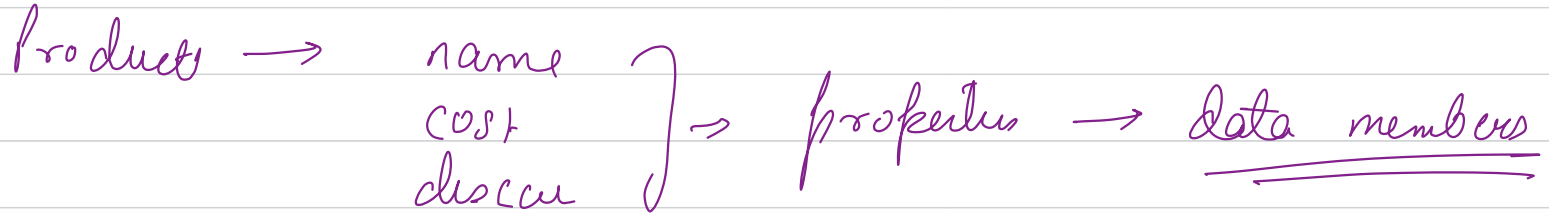
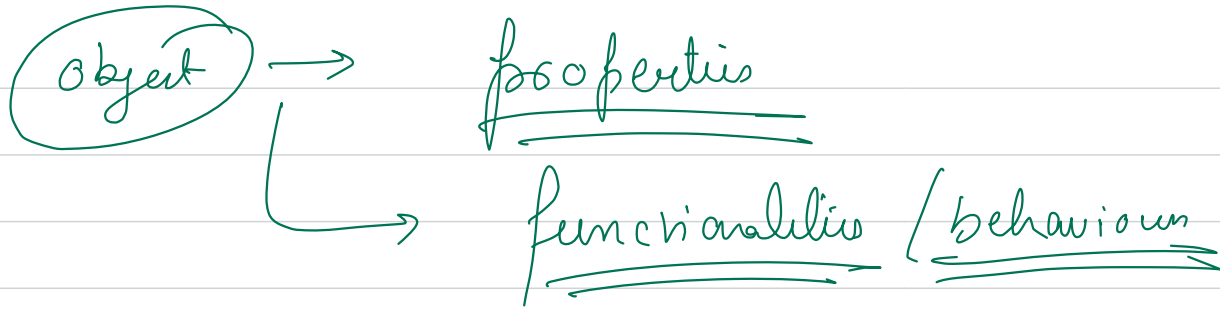
Object

\rightarrow a huge upperhand in any business logic



float f = 3.14; ← ✖
primitive data type

Cardekho.com → vehicle → cost → type float



Constructor



it creates

a new

^{Object}
instance

of a class

dy and f

Constructors

visibility

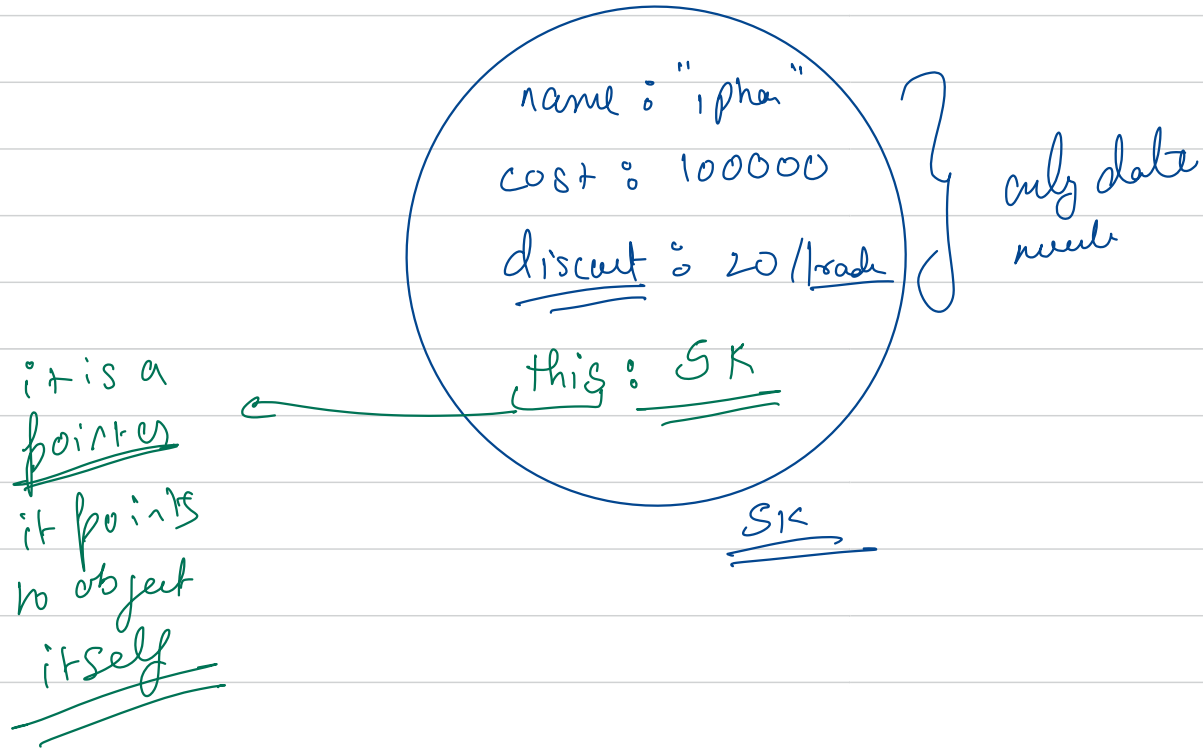
① private → default

② public

③ protected]

→ inheritance

what happens when we create an object: ??



if accessing via a pointer (\rightarrow)

ptr \rightarrow name

if accessing by object (.)

Obj. name

only diff b/w struct & classes is that everythg in
struct is public ^{by default} but we can have custom
visiblity modifiers in classes.

We can have private & protected in structs also