OBJECT POOLING

DANIEL NAEL
OBJECT POOLING IMPROVES PERFORMANCE AND MEMORY USE BY REUSING OBJECTS FROM A FIXED POOL INSTEAD OF ALLOCATING AND FREEING THEM INDIVIDUALLY.
A lot of bullets
A lot of effects
A lot of enemies
A lot of cats
...
When you spawn something:
- Avoid asking the OS for memory **at runtime**
- That is what loading screens are for.
HOW IT WORKS?
1) Instantiate a lot of objects before the start.
2) Save your objects in an array (pool).
3) Hide the objects you are not using.
4) If you need a new object, ask the pool for it.
5) If the object is done, return it to the pool.
6) If the pool is empty, create more (quietly)
1) Instantiate a lot of objects before the start.
2) **Save your objects in an array (pool).**
3) Hide the objects you are not using.
4) If you need a new object, ask the pool for it.
5) If the object is done, return it to the pool.
6) If the pool is empty, create more (quietly)
1) Instantiate a lot of objects before the start.
2) Save your objects in an array (pool).
3) **Hide the objects you are not using.**
4) If you need a new object, ask the pool for it.
5) If the object is done, return it to the pool.
6) If the pool is empty, create more (quietly)
1) Instantiate a lot of objects before the start.
2) Save your objects in an array (pool).
3) Hide the objects you are not using.
4) **If you need a new object, ask the pool for it.**
5) If the object is done, return it to the pool.
6) If the pool is empty, create more (quietly)
1) Instantiate a lot of objects before the start.
2) Save your objects in an array (pool).
3) Hide the objects you are not using.
4) If you need a new object, ask the pool for it.
5) If the object is done, return it to the pool.
6) If the pool is empty, create more (quietly)
1) Instantiate a lot of objects before the start.
2) Save your objects in an array (pool).
3) Hide the objects you are not using.
4) If you need a new object, ask the pool for it.
5) If the object is done, return it to the pool.
6) If the pool is empty, create more (quietly)
HOW TO IMPLEMENT?
THERE IS A SIMPLE WAY
Alive list

Dead list

A B C D E F

E F
C dies

Alive list:
- A
- B
- D

Dead list:
- E
- F
- C
E was asked

Alive list

Dead list

A B D E

F C
THERE IS A BETTER WAY
C dies

A  B  C  D  E  F

swap
C dies

A  B  D  C  E  F

swap
C dies

decrease index
New object was asked

A B D C E F

increase index
TODO

- Read the **Object Pooling** chapter in CGLearn and Game Programming Patterns book
- Do the **Object Pooling** task in CGLearn **IN TEAMS**