

In the above diagram the basic working of the object pool design pattern is explained. While creating object pool we follow below rules:

* We create[**Singleton**](http://codepumpkin.com/?p=398)class to have object pool and create private array to hold objects in it.
* Create acquireReusable() and releaseResusable() methods in the class. In addition to above methods, we can have also have setMaxPoolSize() method to set the size of the pool.

The objects created in the **object pool** have lifecycle as **creation, validation and destroy.**

**Advantage of Object Pool Design Pattern:**

* Most effective when creation of object is costly.
* It boosts the performance of the application significantly.
* It can also provide the limit for the maximum number of objects that can be created.
* It manages the connections and provides a way to reuse and share them.

**Object Pool Implementation:**

To create Object Pool simple things that we need to do is:

* We need a pool(Collection) to store large/heavy objects.
* An interface to get the object from pool, returned the object and validation of the object.

Database connection object creation is costly as it involves database specific driver loading and many other things. So object pool is commonly used for creating pool of connections which can be directly served when required. Which makes the execution of the application faster.