



目的・範囲: <i>Objective & Scope</i>	オブジェクトのクローンを作る
分類名: <i>Classification Name</i>	Prototype pattern(プロトタイプパターン)
著作者: <i>Author</i>	GoF
実施日: <i>Enforcement day</i>	
バージョン: <i>Version</i>	
初版発行日: <i>Original Release</i>	
現行版発行日: <i>Current Release</i>	
キーワード: <i>Key Words</i>	オブジェクトのコピー
背景情報: <i>Background</i>	

◇目的

そのクラスのコピー（クローン）を低コストで作る。

◇効果

手軽にコピーしたいオブジェクトのコピーができる。

◇背景

わかりません

◇プロトタイプパターンの実際のコードと考え方

```
class Clone
  # オブジェクト生成時呼び出し
  def initialize
    @cloneList = Hash.new
  end
  def addCloneList(cloneListName,cloneObjet)
    @cloneList[cloneListName] = cloneObjet
  end
  def createClone(cloneListName)
    obj = @cloneList[cloneListName]
    # 最後の値はreturnされます
    obj.dup_
  end
end

class ViewString
  def initialize(string)
    @viewStr = string
  end
  def view
    puts @viewStr
  end
  def dup_
    self.dup
  end
end

clone = Clone.new
clone.addCloneList('hello',ViewString.new('hello'))
clone.addCloneList('world',ViewString.new('world'))

hello = clone.createClone('hello')
world = clone.createClone('world')
```

hello.view
world.view

Hashにコピーしたいオブジェクトを名前を付けて保存する
そうするとClone1つからさまざまなクローンを作れるようになる

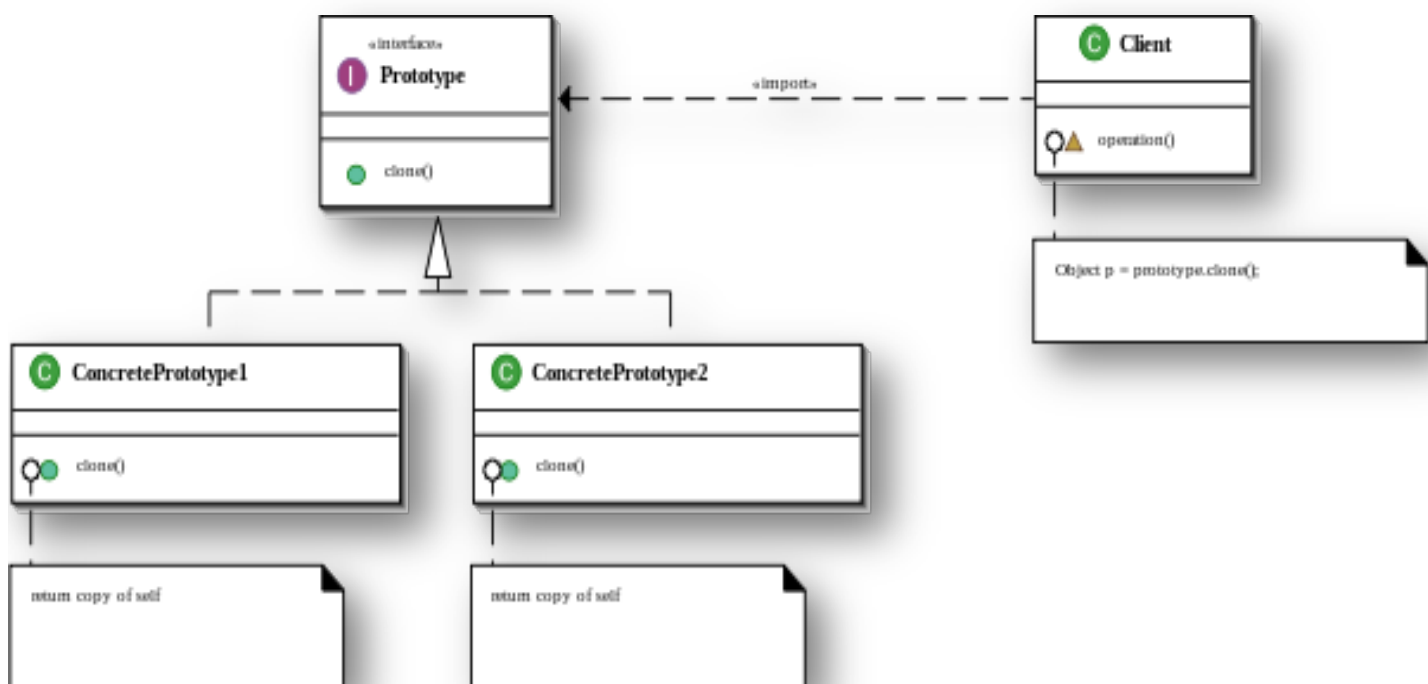
Cloneクラスで管理したくないなら
dup_メソッドを単品で呼び出すだけでできる
こちらへんは言語により、もともとコピーできる機能が存在する言語もあるので調べること。
今回のRubyはself.dupで呼び出し元のオブジェクトをコピーする
dupとcloneの二つのメソッドがあるが、使い分けること(dupは弱いコピー cloneは強いコピー)

◇問題点の改善

コピーを作るだけなのでメモリの節約にはつながらない。改善できない。
それ以外の問題点はおそらく、そのクラスの実装の方法だと思う、鉄のクラスがあってそこからどんな形にするか
getCloneとかそんなメソッドを作っておいて、引数にパイプ、鉄筋などの鉄で作られる製品の名前を入れてやるのか
それともパイプ、鉄筋クラスを作りインターフェースにgetCloneとかをやっておけるか
結局のところ最初の話合いで決まるので、話し合うことが改善策だと思う。

◇プロトタイプパターンのまとめ

そのクラスのコピーを低コストで作り出す。
このデザインパターンを作るときは話し合い、形を決める。
クラスのコピーを作るのでメモリの節約にはならない。
オブジェクトの共有(フライトウェイトパターン)ではないことに注意する。



◇注意

コピーを簡単に作るためのデザインなのでメモリの節約にはならない。
キャストが必要になる場合もある。(Javaとか型がきっちりしてるものなら)
わりと検索すると、もうデフォルトでクローンを行ってくれる言語も多いので調べてみることに。

◇総括

そのオブジェクトのコピーを作る。

用語	説明
キャスト	プログラマが意図的に行う型変更です。例:String hoge = "hoge"; int hogeInt = (int)hoge;
Hash	Hash、Map、連想配列などと呼ばれることもある。例:Hash[name:"hoge"]; Hash["name"] (出力"hoge")
フライウェイトパター	こちらはオブジェクトのコピーではなく共有をするためのデザインパターン

※引用文献e-words、WikiPedia