|  |  |
| --- | --- |
| **Prototype** | |
| **Line** | **Code** |
| **1**  **2**  **3**  **4**  **5**  **6**  **7**  **8**  **9**  **10**  **11**  **12**  **13**  **14**  **15**  **16**  **17**  **18**  **19**  **20**  **21**  **22**  **23**  **24**  **25**  **26**  **27**  **28**  **29**  **30**  **31**  **32**  **33**  **34**  **35**  **36**  **37**  **38**  **39**  **40**  **41**  **42**  **43**  **44**  **45**  **46**  **47**  **48**  **49**  **50**  **51**  **52**  **53**  **54**  **55**  **56**  **57**  **58**  **59**  **60**  **61**  **62**  **63**  **64**  **65**  **66**  **67**  **68**  **69**  **70**  **71**  **72**  **73**  **74**  **75**  **76**  **77**  **78**  **79**  **80**  **81**  **82**  **83**  **84**  **85**  **86**  **87**  **88**  **89**  **90**  **91**  **92**  **93**  **94**  **95**  **96**  **97**  **98**  **99**  **100** | **using** System;  **using** System.IO;  **using** System.Collections.Generic;  **using** System.Runtime.Serialization.Formatters.Binary;  **namespace** Prototype{  [Serializable()]// Serialization is used for the deep copy option  **public abstract class** IPrototype<T>{  **public** T Clone(){  **return** (T)**this**.MemberwiseClone(); //Shallow copy  }  **public** T DeepCopy(){ //Deep Copy  MemoryStream stream = **new** MemoryStream();  BinaryFormatter formatter = **new** BinaryFormatter();  formatter.Serialize(stream, **this**);  stream.Seek(0, SeekOrigin.Begin);  T copy = (T)formatter.Deserialize(stream);  stream.Close();  **return** copy;  }  }  [Serializable()]  // Helper class used to create a second level data structure  **class** DeeperData{  **public string** Data { get; set; }  **public** DeeperData(**string** s){  Data = s;  }  **public override string** ToString(){  **return** Data;  }  }  [Serializable()]  **class** Prototype:IPrototype<Prototype>{  // Content members  **public string** Country { get; set; }  **public string** Capital { get; set; }  **public** DeeperData Language { get; set; }  **public** Prototype(**string** country,**string** capital,**string** language){  Country = country;  Capital = capital;  Language = **new** DeeperData(language);  }  **public override string** ToString(){  **return** Country + "\t\t" + Capital + "\t\t->" + Language;  }  }  **class** PrototypeManager {  **public** Dictionary<**string**, Prototype> prototypes =  **new** Dictionary<**string**, Prototype> {  {"Germany", new Prototype("Germany","Berlin","German")},  {"Italy", new Prototype("Italy","Rome","Italian")},  {"Australia",new Prototype("Australia","Canberra","English")}  };  }  **class** Program{  **static void** Report(**string** s, Prototype a, Prototype b){  Console.WriteLine("\n{0}\nPrototype {1}\nClone {2}",s,a,b);  }  **static void** Main(){  PrototypeManager manager = **new** PrototypeManager();  Prototype c2, c3;  // Make a copy of Australia's data  c2 = manager.prototypes["Australia"].Clone();  Report("Shallow cloning Australia\n===============",  manager.prototypes["Australia"], c2);  // Change the capital of Australia to Sydney  c2.Capital = "Sydney";  Report("Altered Clone's shallow state, prototype unaffected",  manager.prototypes["Australia"], c2);  // Change the language of Australia (deep data)  c2.Language.Data = "Chinese";  Report("Altering Clone deep state: prototype affected \*\*\*\*\*",  manager.prototypes["Australia"], c2);  // Make a copy of Germany's data  c3 = manager.prototypes["Germany"].DeepCopy();  Report("Deep cloning Germany\n============",  manager.prototypes["Germany"], c3);  // Change the capital of Germany  c3.Capital = "Munich";  Report("Altering Clone shallow state, prototype unaffected",  manager.prototypes["Germany"], c3);  // Change the language of Germany (deep data)  c3.Language.Data = "Turkish";  Report("Altering Clone deep state, prototype unaffected",  manager.prototypes["Germany"], c3);  Console.ReadKey();  }  }  } |