|  |
| --- |
| **ChainOfResponsibility** |
| **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** | **using** System;**namespace** ChainOfResponsibility { **interface** IHandler { **string** HandleRequest(**int** data); } **class** ConcreateHandler : IHandler { IHandler next; **int** id; **public int** Limit { get; set; } **public** ConcreateHandler(**int** id, IHandler handler) { **this**.id = id; Limit = id \* 1000; next = handler; } **public string** HandleRequest(**int** data) { **if** (data < Limit) **return** "Request for " + data + " handled at level " + id; **else if** (next != **null**) **return** next.HandleRequest(data); **else return** ("Request for " + data + " handled BY DEFAULT at level " + id); } } **class** Program { **static void** Main() { IHandler start = **null**; **for** (**int** i = 5; i > 0; i--) {//Forming the chain Console.WriteLine("Handler " + i + " deals up to a limit of " + i \* 1000); start = **new** ConcreateHandler(i, start); } //Test it **int**[] a = { 50, 2000, 1500, 10000, 175, 4500 }; **foreach** (**int** i **in** a) Console.WriteLine(start.HandleRequest(i)); Console.ReadKey(); } }}/\* --------------------- OUTPUT ----------------------Handler 5 deals up to a limit of 5000Handler 4 deals up to a limit of 4000Handler 3 deals up to a limit of 3000Handler 2 deals up to a limit of 2000Handler 1 deals up to a limit of 1000Request for 50 handled at level 1Request for 2000 handled at level 3Request for 1500 handled at level 2Request for 10000 handled BY DEFAULT at level 5Request for 175 handled at level 1Request for 4500 handled at level 5------------------------------------------------------\*/ |