```java
 1  import java.util.ArrayList;
 2
 3  abstract class MenuElement {
 4    public String name;
 5    public String url;
 6
 7    MenuElement(String name, String url) {
 8      this.name = name;
 9      this.url = url;
10    }
11
12    public void add(MenuElement component) {
13      throw new UnsupportedOperationException();
14    }
15
16    abstract void displayMenu();
17  }
18
19  class MenuItem extends MenuElement {
20    public MenuItem(String name, String url) {
21      super(name, url);
22    }
23
24    public void displayMenu() {
25      System.out.println(this.name + " : " + this.url);
26    }
27  }
28
29  class Menu extends MenuElement {
30    ArrayList<MenuElement> subMenus = new ArrayList<>();
31
32    public Menu(String name, String url) {
33      super(name, url);
34    }
35
36    public void add(MenuElement menuElement) {
37      this.subMenus.add(menuElement);
38    }
39
40    public void displayMenu() {
41      System.out.println(this.name + " : " + this.url + "\n");
42      this.subMenus.forEach(MenuElement::displayMenu);
43    }
44
45    class Client {
46      public static void main(String[] args) {
47        MenuElement allTutorials = new Menu("Tutorials", "/Tutorials");
48        MenuElement spring = new Menu("Spring", "/Spring");
49        MenuElement versioning = new Menu("Version Management", "/versioning");
50        MenuElement java = new MenuItem("Java", "/Java");
51
52        allTutorials.add(spring);
53        allTutorials.add(versioning);
54        allTutorials.add(java);
55        spring.add(new MenuItem("Spring Core", "/core"));
56        spring.add(new MenuItem("Spring Boot", "/boot"));
57      }
58    }
59  }
```

Pattern STRUTTURALI	Pattern CREAZIONALI	Pattern COMPORTAMENTALI
☐ Adapter ☐ Decorator ☑ Composite	☐ Abstract Factory ☐ Singleton	☐ Observer ☐ State ☐ Visitor ☐ Strategy