Controller작성하기

```xml
<bean class="org.springframework.web.servlet.handler.BeanNameUrlHandlerMapping">
    <property name="order" value="1" />
</bean>
                <!-- controller 직접생성및  urlmapping -->
<bean name="/hello.ok" class="com.acorn.cont.AController" />
<bean name="/hello2.ok" class="com.acorn.cont.BController" />
```
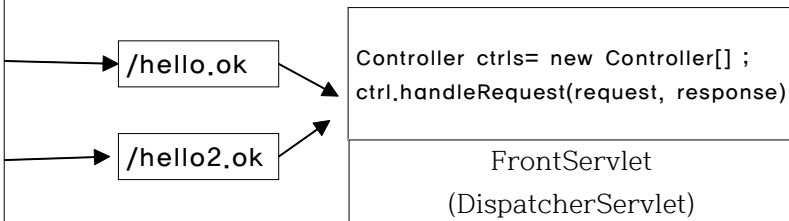
```java
public class BController  implements Controller {

        @Override
        public ModelAndView handleRequest(HttpServletRequest request, HttpServletResponse response)
throws Exception {
                // TODO Auto-generated method stub

                ModelAndView mv = new ModelAndView("view2");   // view이름
                mv.addObject("data", "spring hello!");     //model
                return mv;
        }

}
```

xml설절할 때의 url와 controller매핑방식 (클래스단위로 매핑)

```
/hello.ok  ──►
                    Controller ctrls= new Controller[] ;
                    ctrl.handleRequest(request, response)
/hello2.ok ──►
                    ─────────────────────────────
                           FrontServlet
                        (DispatcherServlet)
```

```java
public class AController implements Controller{
  @Override
  public ModelAndView  handleRequest( ){
  }
}
```

```java
public class BController implements Controller{
  @Override
  public ModelAndView  handleRequest( ){
  }
}
```

```xml
<bean name="/hello.ok"  class="com.acorn.web.AController" />
<bean name="/hello2.ok"  class="com.acorn.web.BController" />
```

■ 어노테이션으로 매핑하는 방식 (매서드 단위로 url매핑함)

```java
@Controller
public class CController {

        @GetMapping("/hello.ok")
        public String  method1(Model m) {
                m.addAttribute("drama", "모범택시2");
                return "view1";
        }

        @GetMapping("/hello2.ok")
        public String  method2(Model m) {
                m.addAttribute("name", "도기&고은");
                return "view2";
        }

}
```