



链滴

springmvc 在不给定注解的情况下，参数是怎样绑定的？

作者：[714593351](#)

原文链接：<https://ld246.com/article/1452062085238>

来源网站：[链滴](#)

许可协议：[署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)


```

else if (CookieValue.class.isInstance(paramAnn)) {
    CookieValue cookieValue = (CookieValue) paramAnn;
    cookieName = cookieValue.value();
    required = cookieValue.required();
    defaultValue = parseDefaultValueAttribute(cookieValue.defaultValue());
    annotationsFound++;
}
else if (PathVariable.class.isInstance(paramAnn)) {
    PathVariable pathVar = (PathVariable) paramAnn;
    pathVarName = pathVar.value();
    annotationsFound++;
}
else if (ModelAttribute.class.isInstance(paramAnn)) {
    ModelAttribute attr = (ModelAttribute) paramAnn;
    attrName = attr.value();
    annotationsFound++;
}
else if (Value.class.isInstance(paramAnn)) {
    defaultValue = ((Value) paramAnn).value();
}
else if (paramAnn.annotationType().getSimpleName().startsWith("Valid")) {
    validate = true;
    Object value = AnnotationUtils.getValue(paramAnn);
    validationHints = (value instanceof Object[] ? (Object[]) value : new Object[] {value});
}
}

if (annotationsFound > 1) {
    throw new IllegalStateException("Handler parameter annotations are exclusive choices
- " +
        "do not specify more than one such annotation on the same parameter: " + hand
erMethod);
}

if (annotationsFound == 0) { // 若没有发现注解
    Object argValue = resolveCommonArgument(methodParam, webRequest); //判断W
bRequest是否可赋值给参数
    if (argValue != WebArgumentResolver.UNRESOLVED) {
        args[i] = argValue;
    }
    else if (defaultValue != null) {
        args[i] = resolveDefaultValue(defaultValue);
    }
    else {
        Class<?> paramType = methodParam.getParameterType();
        if (Model.class.isAssignableFrom(paramType) || Map.class.isAssignableFrom(paramT
pe)) {
            if (!paramType.isAssignableFrom(implicitModel.getClass())) {
                throw new IllegalStateException("Argument [" + paramType.getSimpleName()
+ "] is of type " +
                    "Model or Map but is not assignable from the actual model. You may need
to switch " +
                    "newer MVC infrastructure classes to use this argument.");
            }
        }
    }
}

```

```

        args[i] = implicitModel;
    }
    else if (SessionStatus.class.isAssignableFrom(paramType)) {
        args[i] = this.sessionStatus;
    }
    else if (HttpEntity.class.isAssignableFrom(paramType)) {
        args[i] = resolveHttpRequest(methodParam, webRequest);
    }
    else if (Errors.class.isAssignableFrom(paramType)) {
        throw new IllegalStateException("Errors/BindingResult argument declared " +
            "without preceding model attribute. Check your handler method signature!")
    }
    else if (BeanUtils.isSimpleProperty(paramType)) { // 判断是否参数类型是否是简单类型
        // 若是在使用@RequestParam方式来处理, 否则使用@ModelAttribute方式处理
        paramName = "";
    }
    else {
        attrName = "";
    }
}

if (paramName != null) {
    args[i] = resolveRequestParam(paramName, required, defaultValue, methodParam, webRequest, handler);
}
else if (headerName != null) {
    args[i] = resolveRequestHeader(headerName, required, defaultValue, methodParam, webRequest, handler);
}
else if (requestBodyFound) {
    args[i] = resolveRequestBody(methodParam, webRequest, handler);
}
else if (cookieName != null) {
    args[i] = resolveCookieValue(cookieName, required, defaultValue, methodParam, webRequest, handler);
}
else if (pathVarName != null) {
    args[i] = resolvePathVariable(pathVarName, methodParam, webRequest, handler);
}
else if (attrName != null) {
    WebDataBinder binder =
        resolveModelAttribute(attrName, methodParam, implicitModel, webRequest, handler);
    boolean assignBindingResult = (args.length > i + 1 && Errors.class.isAssignableFrom(paramTypes[i + 1]));
    if (binder.getTarget() != null) {
        doBind(binder, webRequest, validate, validationHints, !assignBindingResult);
    }
    args[i] = binder.getTarget();
    if (assignBindingResult) {
        args[i + 1] = binder.getBindingResult();
        i++;
    }
}
}

```

