



链滴

HttpServletRequest 中直接获取实体对象

作者: [Lethew](#)

原文链接: <https://ld246.com/article/1478857907642>

来源网站: [链滴](#)

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

组件功能：从HttpServletRequest中获取数据并填充到具体的实体对象

注：该方法目前应该无法获取json数据进行填充

核心方法：coverDomainFromRequest

```
/**
```

- 从request中获取实体对象
- @param e 用于装载数据的空实体对象
- @param request 当前的请求对象
- @param <E> 实体对象类型
- @throws IllegalAccessException 数据转换装载异常时抛出

```
*/
```

```
private <E extends BaseDomain> void coverDomainFromRequest(E e,HttpServletRequest request) throws IllegalAccessException {
```

```
//获取实体类型
```

```
Class<?> clazz = e.getClass();
```

```
//遍历类型成员变量并填充数据
```

```
for (Field f: clazz.getDeclaredFields()) {
```

```
//从request中获取数据
```

```
String value = request.getParameter(f.getName());
```

```
//数据判空
```

```
if (value == null){
```

```
    continue;
```

```
}
```

```
//非空情况下进行数据填充
```

```
f.setAccessible(true);
```

```
f.set(e, ConvertUtil.castFromObject(value, f.getType()));
```

```
}
```

```
}
```

使用到的方法：ConvertUtil.castFromObject

```
public static Object castFromObject(Object obj, Class<?> destType){
```

```
if (obj == null){
```

```
return null;
```

```
}
```

```
else if (obj.getClass().equals(destType)){
```

```
return obj;
```

```
}
```

```
Converter converter = converters.get(destType);  
if (converter != null){  
    return converter.convert(obj);  
}
```

```
String typeName = obj.getClass().getCanonicalName();  
if (typeName.equals("java.lang.String")){  
    return castFromString((String)obj, destType);  
}  
else if (typeName.equals("boolean") || typeName.equals("java.lang.Boolean")){  
    return castFromString(Boolean.toString((Boolean)obj), destType);  
}  
else if (typeName.equals("int") || typeName.equals("java.lang.Integer")){  
    return castFromString(Integer.toString((Integer)obj), destType);  
}  
else if (typeName.equals("short") || typeName.equals("java.lang.Short")){  
    return castFromString(Short.toString((Short)obj), destType);  
}  
else if (typeName.equals("long") || typeName.equals("java.lang.Long")){  
    return castFromString(Long.toString((Long)obj), destType);  
}  
else if (typeName.equals("double") || typeName.equals("java.lang.Double")){  
    return castFromString(Double.toString((Double)obj), destType);  
}  
else if (typeName.equals("float") || typeName.equals("java.lang.Float")){  
    return castFromString(Float.toString((Float)obj), destType);  
}  
else if (typeName.equals("java.math.BigInteger")){  
    return castFromString(((java.math.BigInteger)obj).toString(), destType);  
}  
else if (typeName.equals("java.math.BigDecimal")){  
    return castFromString(((java.math.BigDecimal)obj).toString(), destType);  
}  
else if (typeName.equals("java.sql.Date")){  
    return castFromString(((java.sql.Date)obj).toString(), destType);  
}  
else if (typeName.equals("java.sql.Time")){  
    return castFromString(((java.sql.Time)obj).toString(), destType);  
}  
else if (typeName.equals("java.sql.Timestamp")){  
    return castFromString(((java.sql.Timestamp)obj).toString(), destType);  
}  
else{  
    return obj;  
}  
}
```