

自动生成规则编码，比如：订单号的生成，前缀+年月日时分秒 时间戳

如果创建一个规则实现？

- 第一步：实现接口 IFillRuleHandler

参考实例：

```
/**
 * 填值规则Demo：生成订单号
 * 【测试示例】
 */
public class OrderNumberRule implements IFillRuleHandler {

    @Override
    public Object execute(JSONObject params, JSONObject formData) {
        String prefix = "CN";
        //订单前缀默认为CN 如果规则参数不为空，则取自定义前缀
        if (params != null) {
            Object obj = params.get("prefix");
            if (obj != null) prefix = obj.toString();
        }
        SimpleDateFormat format = new SimpleDateFormat("yyyyMMddHHmmss");
        int random = RandomUtils.nextInt(90) + 10;
        String value = prefix + format.format(new Date()) + random;
        // 根据formData的值的不同，生成不同的订单号
        String name = formData.getString("name");
        if (!StringUtils.isEmpty(name)) {
            value += name;
        }
        return value;
    }
}
```

- 第二步：在规则管理里面配置

编辑

规则名称: 请输入规则名称

+ 新增 导出

已选择 0 项

| | # | |
|--------------------------|---|--|
| <input type="checkbox"/> | 1 | |
| <input type="checkbox"/> | 2 | |

* 规则名称: 订单号生成

* 规则Code: order_num_rule

* 规则实现类: org.jeecg.modules.online.cgform.rule.OrderNumberRule

规则参数: 请输入规则参数

关闭 确定

- 第三步：通过接口执行规则生成

参数：

接口URL: /sys/fillRule/executeRuleByCode/{ruleCode}

```
/**
 * 通过 ruleCode 执行自定义填值规则
 *
 * @param ruleCode 要执行的填值规则编码
 * @param formData 表单数据, 可根据表单数据的不同生成不同的填值结果
 * @return 运行后的结果
 */
@PutMapping("/executeRuleByCode/{ruleCode}")
public Result executeByRuleCode(@PathVariable("ruleCode") String ruleCode, @RequestBody JSONObject formData) {
    Object result = FillRuleUtil.executeRule(ruleCode, formData);
    return Result.ok(result);
}
```

- 第四步: 批量编码规则生成接口

/sys/fillRule/executeRuleByCode/executeRuleByCodeBatch

具体用法参考:

org.jeecg.modules.system.controller.SysFillRuleController#executeByRuleCodeBatch

```
/**
 * 批量通过 ruleCode 执行自定义填值规则
 *
 * @param ruleData 要执行的填值规则JSON数组:
 * 示例: { "commonFormData": {}, rules: [ { "ruleCode": "xxx", "formData": null } ] }
 * @return 运行后的结果, 返回示例: [{"ruleCode": "order_num_rule", "result": "CN2019111117212984"}]
 */
@PutMapping("/executeRuleByCodeBatch")
public Result executeByRuleCodeBatch(@RequestBody JSONObject ruleData) {
    JSONObject commonFormData = ruleData.getJSONObject("commonFormData");
    JSONArray rules = ruleData.getJSONArray( key: "rules");
    // 遍历 rules , 批量执行规则
    JSONArray results = new JSONArray(rules.size());
    for (int i = 0; i < rules.size(); i++) {
        JSONObject rule = rules.getJSONObject(i);
        String ruleCode = rule.getString( key: "ruleCode");
        JSONObject formData = rule.getJSONObject("formData");
        // 如果没有传递 formData , 就用common的
        if (formData == null) {
            formData = commonFormData;
        }
    }
}
```

其他技巧

JAVA代码里, 如何直接调用生成编码?

```
* @param params 页面配置参数
* @param formData 动态参数
FillRuleUtil.executeRule(String ruleCode, JSONObject formData)
```

前端JS里如何调用生成编码?

可通过 `get` 请求调用后台填值规则接口来实现生成编码。

示例

```
import { getAction } from '@api/manage'

const ruleBaseURL = '/sys/fillRule/executeRuleByCode/'
export default {
  name: 'App',
  data() {
    return {
      // 页面的数据
      model: {},
    }
  }
}
```



```
// 请求的url
url: { orderNum: ruleBaseURL + 'order_num_rule' }
},
methods: {
  getOrderNum() {
    getAction(this.url.rule.orderNum, this.model).then(res => {
      // 执行成功，获取返回的值，并赋到页面上
      if (res.success) {
        this.model.orderNum = res.result
      }
    })
  }
}
}
```

Online表单的JS增强里如何调用生成编码？

和前端JS里一样，可通过 `getAction` 方法来调用接口生成编码

表单设计器 JS 增强里如何调用生成编码？

可以通过 `api.get()` 方法实现获取填值规则的值，获取到之后再通过 `api.setFormData()` 方法赋到页面上。

api的使用方法可见 [表单设计器-->高级操作手册-->JS增强](#) 文档

示例

```
// 获取所有 form data
var formData = api.getFormData() || {}
// 请求后台的填值规则接口地址
var url = '/sys/fillRule/executeRuleByCode/'
var ruleCode = 'order_num_rule'
api.get(url + ruleCode, formData).then(res => {
  // 执行成功，获取返回的值，并赋到页面上
  if (res.success) {
    var value = res.result
    api.setFormData('orderNum', value)
  }
})
```

上一篇：[列表数据权限手工集成示例](#)

下一篇：[Online开发](#)