swagger: “2.0” 指定swagger的版本号

info: 关于该api的信息，版本和标题

host： 表示你的ip地址或是域名

paths: 路由地址

post: 请求方法

tags: 命名空间

summary: 描述信息

consumes：消费格式‘application/json’和‘application/xml’

produces：生产格式‘application/json’和‘application/xml’

parameters：参数

name： 参数名字

description：描述

required：是否必要

in：属于哪种参数 body, header, formData, query, path

body只能有一个，body里的参数需要以model的形式通过schema放在里面

schema:

$ref: “#/definitions/Auth”

$ref: 把model 在definitions里的Auth当作参数放入body中。

model Auth的写法，会放到yaml的最后

responses： 状态码

string integer

Map<String, Object> params = new HashMap<>();

params.put("limit", limit);

params.put("offset", offset);

params.put("typeName", typeName);

params.put("name", name);

params.put("level", level);

params.put("type", type);

- name: "flow"

in: "body"

required: true

schema:

$ref: "#/definitions/Flow"

params.put("offset", offset \* limit);

PageInfo<?> pageInfo = PageInfoUtils.setPropertByPageInfo(offset, limit,total, jobList);

JobDirectionTypeDTO:

type: "object"

JobDirectionTypeDTOList:

type: array

items:

$ref: '#/definitions/JobDirectionTypeDTO'

HttpServletRequest req = ((ServletRequestAttributes) RequestContextHolder.getRequestAttributes()).getRequest();

HttpServletResponse response = ((ServletRequestAttributes) RequestContextHolder.*getRequestAttributes*()).getResponse();

ids.toArray(new String[ids.size()])

Mockito.when(sysPropertiesService.getByAlias("menu\_contest")).thenReturn(sysPropertiesDo);

Mockito.when(sysPropertiesService.getByAlias("menu\_vulnerability")).thenReturn(sysPropertiesDo);

Mockito.when(sysPropertiesService.getByAlias("forum\_center")).thenReturn(sysPropertiesDo);