

# 基本信息

- 接口地址  
http://127.0.0.1:20030/createNewProblem
- 请求方法  
POST
- 请求格式  
application/json
- 响应格式  
application/json

# 请求参数

## 参数结构

```
{
  "OJ": "string",
  "problemName": "string",
  "samples": [
    {
      "id": 1,
      "input": "string",
      "output": "string",
      "timeLimit": 1000
    }
  ]
}
```

## 参数说明

参数名	类型	必填	说明
OJ	string	是	OJ 名称， 取值范围： ["Luogu", "Hetao", "atcoder", "codeforces"]
problemName	string	是	题目名称，长度限制： ≤ 32

参数名	类型	必填	说明
samples	array	是	样例组数组，至少包含 1 个样例
id	integer	是	样例组编号，从 1 开始，需保证唯一
input	string	是	样例输入数据
output	string	是	样例输出数据
timeLimit	integer	否	时间限制（毫秒），默认 1000 ，必须为正整数

## 请求示例

### curl

```
curl -X POST http://127.0.0.1:20030/createNewProblem \
-H "Content-Type: application/json" \
-d '{
  "OJ": "Luogu",
  "problemName": "P1001 A + B Problem",
  "samples": [
    {
      "id": 1,
      "input": "1 2",
      "output": "3",
      "timeLimit": 1000
    },
    {
      "id": 2,
      "input": "100 200",
      "output": "300",
      "timeLimit": 2000
    }
  ]
}'
```

# JavaScript Fetch

```
fetch("http://127.0.0.1:20030/createNewProblem", {
  method: "POST",
  headers: {
    "Content-Type": "application/json"
  },
  body: JSON.stringify({
    OJ: "Luogu",
    problemName: "P1001 A + B Problem",
    samples: [
      { id: 1, input: "1 2", output: "3", timeLimit: 1000 },
      { id: 2, input: "100 200", output: "300", timeLimit: 2000 }
    ]
  })
})
.then(res => res.json())
.then(data => console.log(data))
.catch(err => console.error(err));
```

## 响应参数

### 成功响应

```
{
  "code": 200,
  "message": "Problem created successfully"
}
```

### 失败响应

当请求未成功时，会在响应中额外返回 `invalidField` 字段，指示具体出错的参数。

```
{
  "code": 400,
  "message": "Invalid parameters: problemName too long.",
  "invalidField": "problemName"
}
```

另一个示例：样例数组为空

```
{
  "code": 400,
  "message": "Invalid parameters: samples must not be empty.",
  "invalidField": "samples"
}
```

## 状态码说明

状态码	说明
200	请求成功，题目已创建
400	参数错误，响应中会包含 invalidField
500	内部错误