

开发支持 QQ 聊天软件的 Wechaty Puppet Provider 模块

学生github账号：anaivebird

导师：李佳芮、李卓桓

QQ puppet仓库地址：

<https://github.com/anaivebird/wechaty-puppet-oicq>

项目背景

- Wechaty 社区目前已经支持微信、Whatsapp、企业微信、飞书等常见流行即时通讯工具，并且能够通过多语言 SDK（比如 Python Wechaty）进行调用。
- QQ 是国内和微信并列的两大聊天软件。
- 我们在本次 Summer 2021 的项目中，Wechaty 希望可以实现对 QQ Chatbot 的支持。
- 通过 Wechaty Puppet 的接口，可以将 QQ 进行 RPA 封装
- 目标是使 `wechaty-puppet-qq` 供 Wechaty 开发者方便接入 QQ 平台，使其成为 Wechaty 可以使用的社区生态模块。

方案描述

- 通过对Wechaty Puppet Provider和wechaty-puppet-mock项目等进行了解和学习，并对已有项目进行梳理
- oicq项目是使用node.js语言实现的一个qq机器人接口库，为我们本次项目实施带来了极大便利
- 实现通过 Wechaty 加载 wechaty-puppet-qq模块，并包装[oicq](#)项目，实现文本消息的收发功能
- 提供一个 examples/ding-dong-bot.ts ，完成“接收到文字消息ding时，自动回复消息dong”等功能。

目前进展

- 通过与导师沟通，并观看Wechaty Puppet Provider 的 workshop 视频，熟悉Wechaty Puppet Provider的业务流程
- 完成了之前掌握不够熟练的nodejs、npm、typescript等内容进行快速的了解和学习，对时间安排做好具体的规划
- 完成了puppet代码的prototyping，参考wechaty-puppet-5g-qq 模块对接口的封装格式，并通过oicq项目实现的qq协议，实现文本消息的收发功能，并且用实际的qq账号扫码登录实践成功。
- 完成了wechaty.js.org网站博客仓库的pull request，了解了git和github的基本操作
- 成功将目前进展提交到了wechaty-puppet-oicq仓库，后续将会把这个仓库提交到wechaty账号下

启动配置

- 将oicq-puppet克隆到本地
- 执行npm install安装依赖
- 在环境变量中设置QQ号
- 输入npm run start运行bot

WECHATY PUPPET OICQ (基于OICQ项目的QQ机器人)

Powered By Wechaty

npm package

0.4.5

npm

tag not found

运行方法

配置系统环境变量

1. WECHATY_PUPPET_OICQ_QQNUMBER: 要登录的QQ号

安装依赖

将代码clone到本地, 执行npm install

运行示例代码

```
npm run start
```

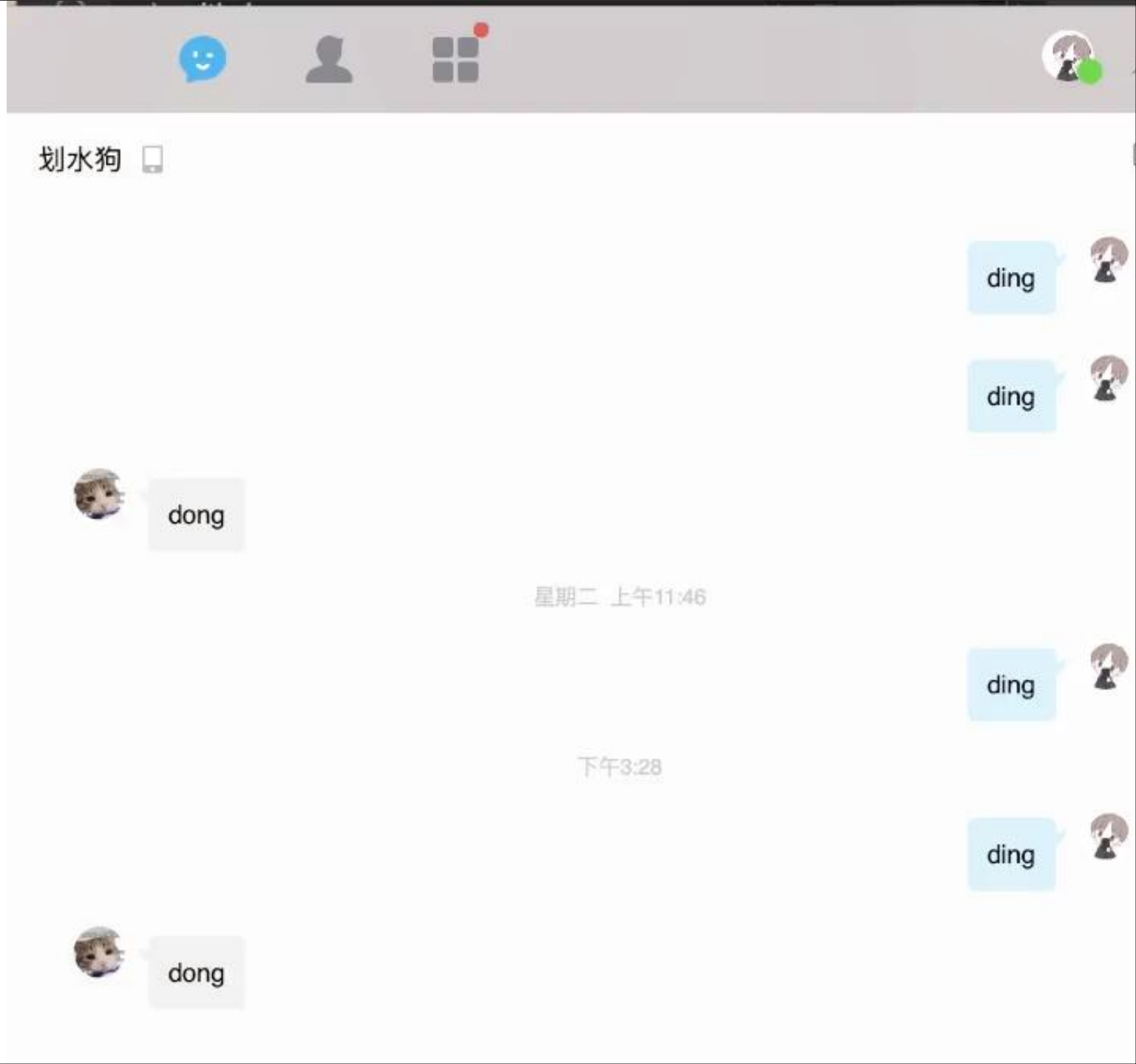
按照提示完成扫码完成后按下回车, 即可运行示例机器人

扫码完之后成功登录

PROBLEMS	OUTPUT	TERMINAL	DEBUG CONSOLE
		[2021-08-15T15:49:15.117] [MARK]	[Android:1962099319] - View Changelogs: https://github.com/t
		[2021-08-15T15:49:15.117] [MARK]	[Android:1962099319] - -----
		[2021-08-15T15:49:15.119] [MARK]	[Android:1962099319] - 正在探索可用服务器...
		Puppet Version: 0.0.1	
		Please wait... I'm trying to login in...	
			I
		[2021-08-15T15:49:15.358] [MARK]	[Android:1962099319] - connecting to 109.244.153.175:8080
		[2021-08-15T15:49:15.373] [MARK]	[Android:1962099319] - 109.244.153.175:8080 connected
		[2021-08-15T15:49:15.602] [MARK]	[Android:1962099319] - Welcome, 划水狗 ! 初始化资源...
		[2021-08-15T15:49:15.767] [MARK]	[Android:1962099319] - 加载了9个好友, 3个群。
		[2021-08-15T15:49:15.767] [MARK]	[Android:1962099319] - 初始化完毕, 开始处理消息。

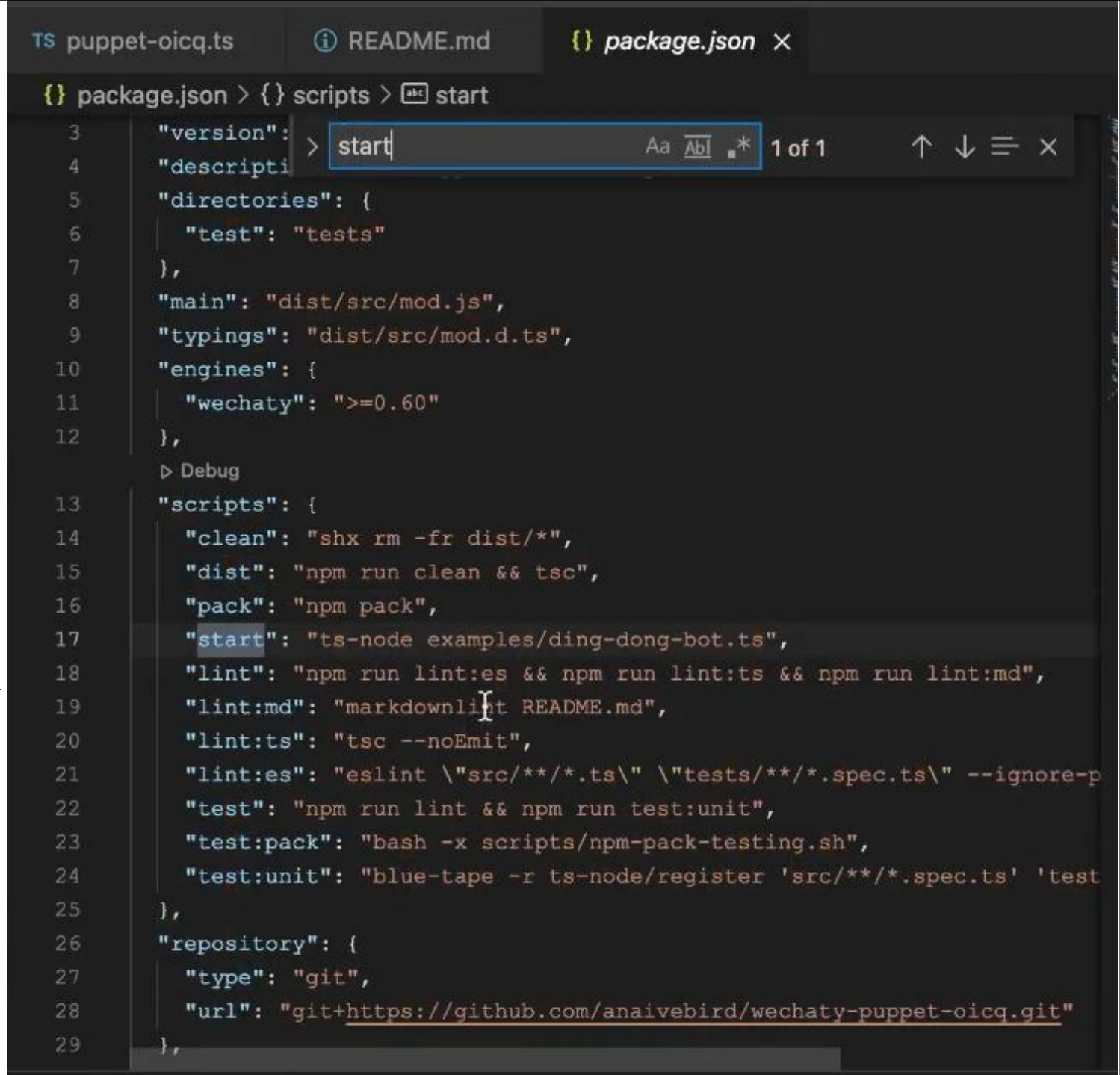
给机器人发消息

- 给机器人发送ding的消息
- 若机器人启动成功
- 则会回复dong的消息



npm run start

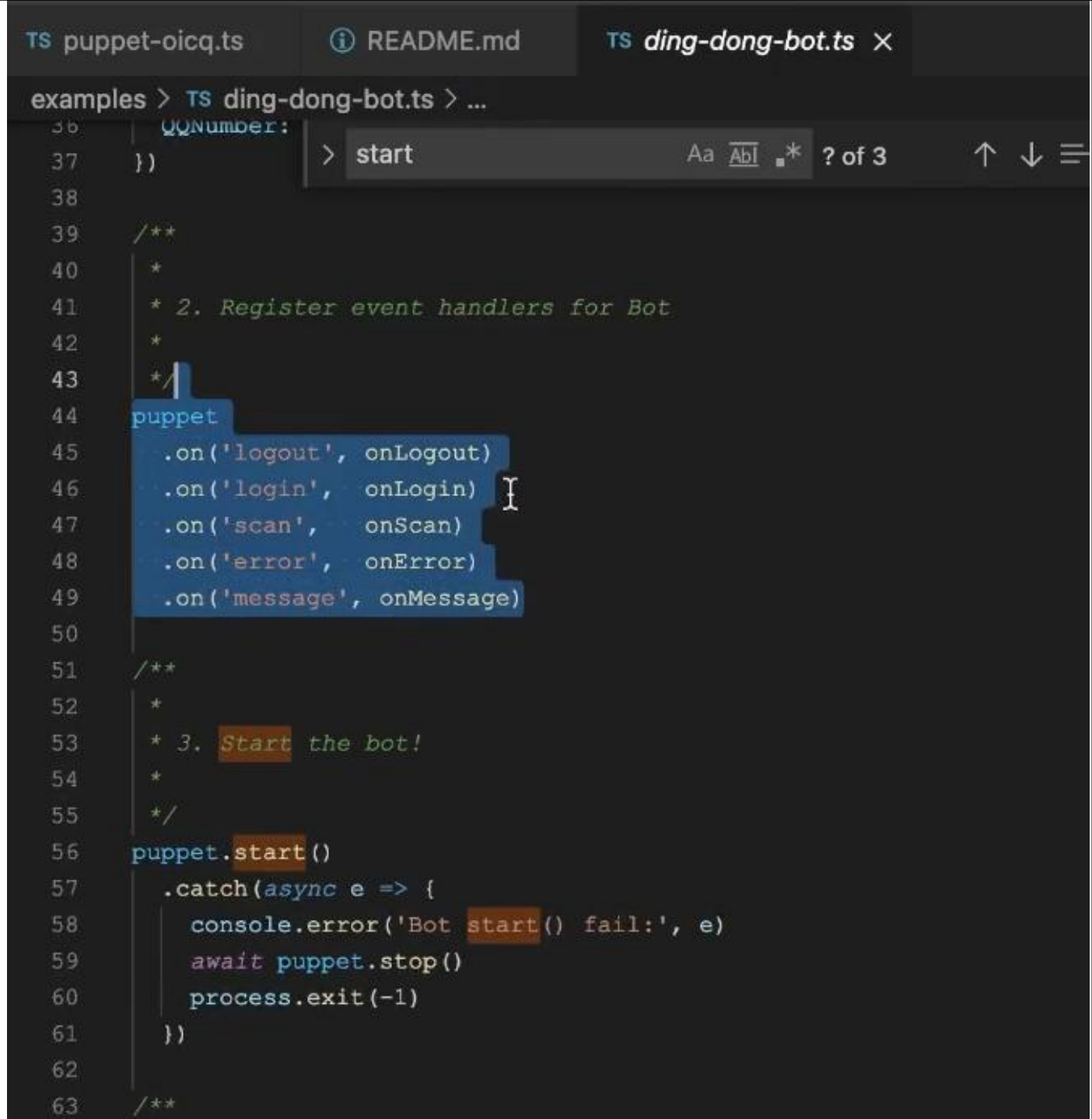
- node.js小知识
- 在package.json中
- 定义了npm run start作用
- 将会运行ding-dong-bot.ts
- 其他脚本也一样在这里定义



```
TS puppet-oicq.ts  README.md  {} package.json X
{} package.json > {} scripts > start
3  "version": "1.0.0"
4  "description": "A puppet-oicq bot"
5  "directories": {
6    "test": "tests"
7  },
8  "main": "dist/src/mod.js",
9  "typings": "dist/src/mod.d.ts",
10 "engines": {
11   "wechaty": ">=0.60"
12 },
13 "scripts": {
14   "clean": "shx rm -fr dist/*",
15   "dist": "npm run clean && tsc",
16   "pack": "npm pack",
17   "start": "ts-node examples/ding-dong-bot.ts",
18   "lint": "npm run lint:es && npm run lint:ts && npm run lint:md",
19   "lint:md": "markdownlint README.md",
20   "lint:ts": "tsc --noEmit",
21   "lint:es": "eslint \"src/**/*.ts\" \"tests/**/*.spec.ts\" --ignore-p",
22   "test": "npm run lint && npm run test:unit",
23   "test:pack": "bash -x scripts/npm-pack-testing.sh",
24   "test:unit": "blue-tape -r ts-node/register 'src/**/*.spec.ts' 'test",
25 },
26 "repository": {
27   "type": "git",
28   "url": "git+https://github.com/anaivebird/wechaty-puppet-oicq.git"
29 },
```


事件是如何绑定的

- puppet.on(事件名, 函数名)
- 用这种方法可以
- 当puppet收到事件的时候
- 对应事件处理函数被调用
- 事件处理函数绑定是
- ding-dong-bot.ts的主要任务



```
TS puppet-oicq.ts  README.md  TS ding-dong-bot.ts X
examples > TS ding-dong-bot.ts > ...
36  QQNumber:
37  })
38
39  /**
40   *
41   * 2. Register event handlers for Bot
42   *
43   */
44  puppet
45    .on('logout', onLogout)
46    .on('login', onLogin)
47    .on('scan', onScan)
48    .on('error', onError)
49    .on('message', onMessage)
50
51  /**
52   *
53   * 3. Start the bot!
54   *
55   */
56  puppet.start()
57    .catch(async e => {
58      console.error('Bot start() fail:', e)
59      await puppet.stop()
60      process.exit(-1)
61    })
62
63  /**
```