



---

でbotを動かしてみよう

---

# BOT作成手順

---

1. Node（ノード）.jsのインストール
  2. Botkit（ボットキット）の導入
  3. Slackでbotを動かそう
  4. オリジナルの返答を追加
-

---

# BOT作成手順

---

1. **Node（ノード）.jsのインストール**
  2. Botkit（ボットキット）の導入
  3. Slackでbotを動かそう
  4. オリジナルの返答を追加
-



---

# Node.jsのインストール

---

Node.jsの公式サイトより  
インストーラをDLしてインストール（全部「はい」でOK）

Node.js公式サイト - <https://nodejs.org/en/download/>



---

# BOT作成手順

---

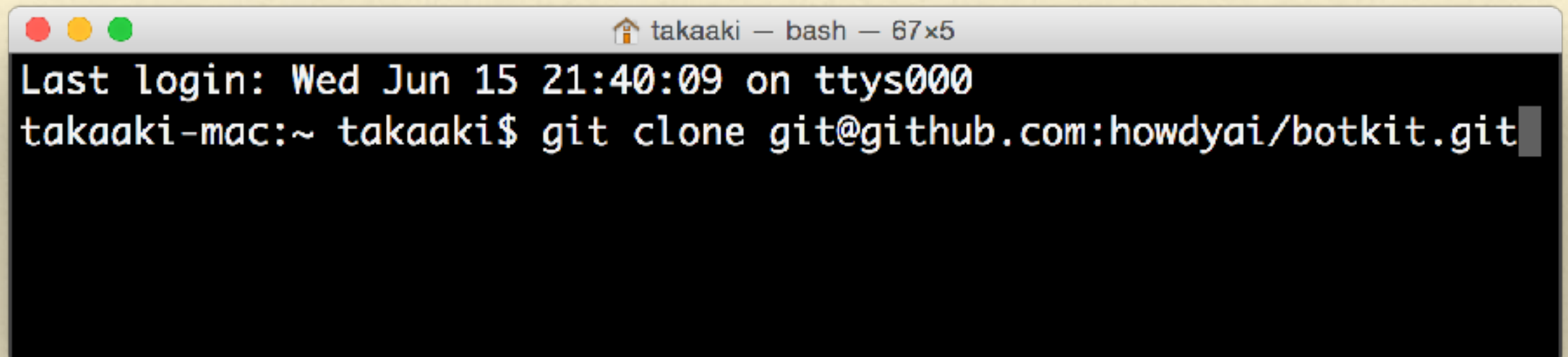
1. Node（ノード）.jsのインストール
  - 2. Botkit（ボットキット）の導入**
  3. Slackでbotを動かそう
  4. オリジナルの返答を追加
-

# Botkitの導入

GitHubのBotkitリポジトリからClone or DLする。

※DLした場合は、好きなフォルダに解凍してください。

<https://github.com/howdyai/botkit>



```
takaaki — bash — 67x5
Last login: Wed Jun 15 21:40:09 on ttys000
takaaki-mac:~ takaaki$ git clone git@github.com:howdyai/botkit.git
```

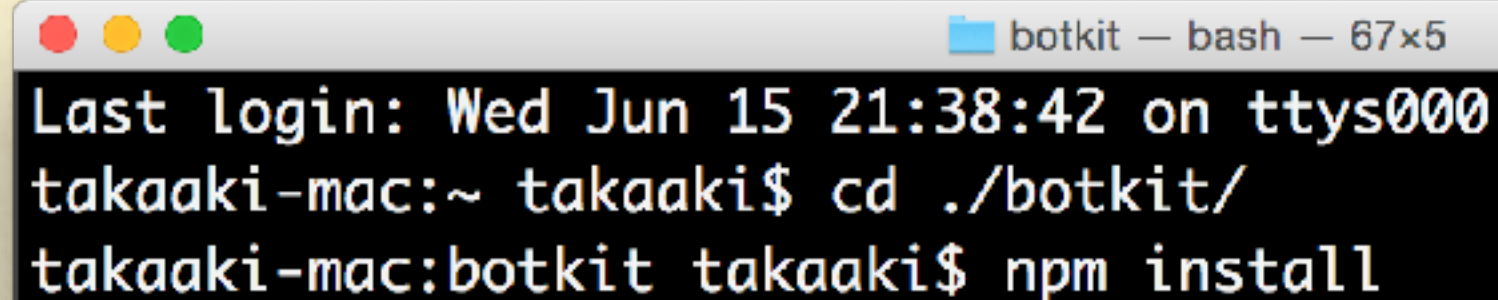


---

# Botkitの導入

---

Botkitを入れたディレクトリに移動し、  
必要なモジュールをインストールします。

A terminal window with a title bar that says "botkit — bash — 67x5". The window has three colored window control buttons (red, yellow, green) on the left. The terminal text shows the user navigating to the botkit directory and installing dependencies with npm.

```
Last login: Wed Jun 15 21:38:42 on ttys000
takaaki-mac:~ takaaki$ cd ./botkit/
takaaki-mac:botkit takaaki$ npm install
```

---

# BOT作成手順

---

1. Node（ノード）.jsのインストール
  2. Botkit（ボットキット）の導入
  3. **Slackでbotを動かそう**
  4. オリジナルの返答を追加
-



---

# Slackでbotを動かそう

---

以下のサイトへ飛んで、botを作成する。（要サインイン）

<https://lets-make-a-bot.slack.com/apps/new/A0F7YS25R-bots>

～入力するもの～

- ・ bot名：@[苗字(ローマ字)]-bot

例) @matsunaga-bot ※苗字が被る人は名前で

---

# Slackでbotを動かそう



## Bots

Connect a bot to the Slack Real Time Messaging API.

Run code that listens and posts to your Slack team just as a user would.

### Username

Start by choosing a username for your bot

↓@[苗字]-botを入力

@matsunaga-bot

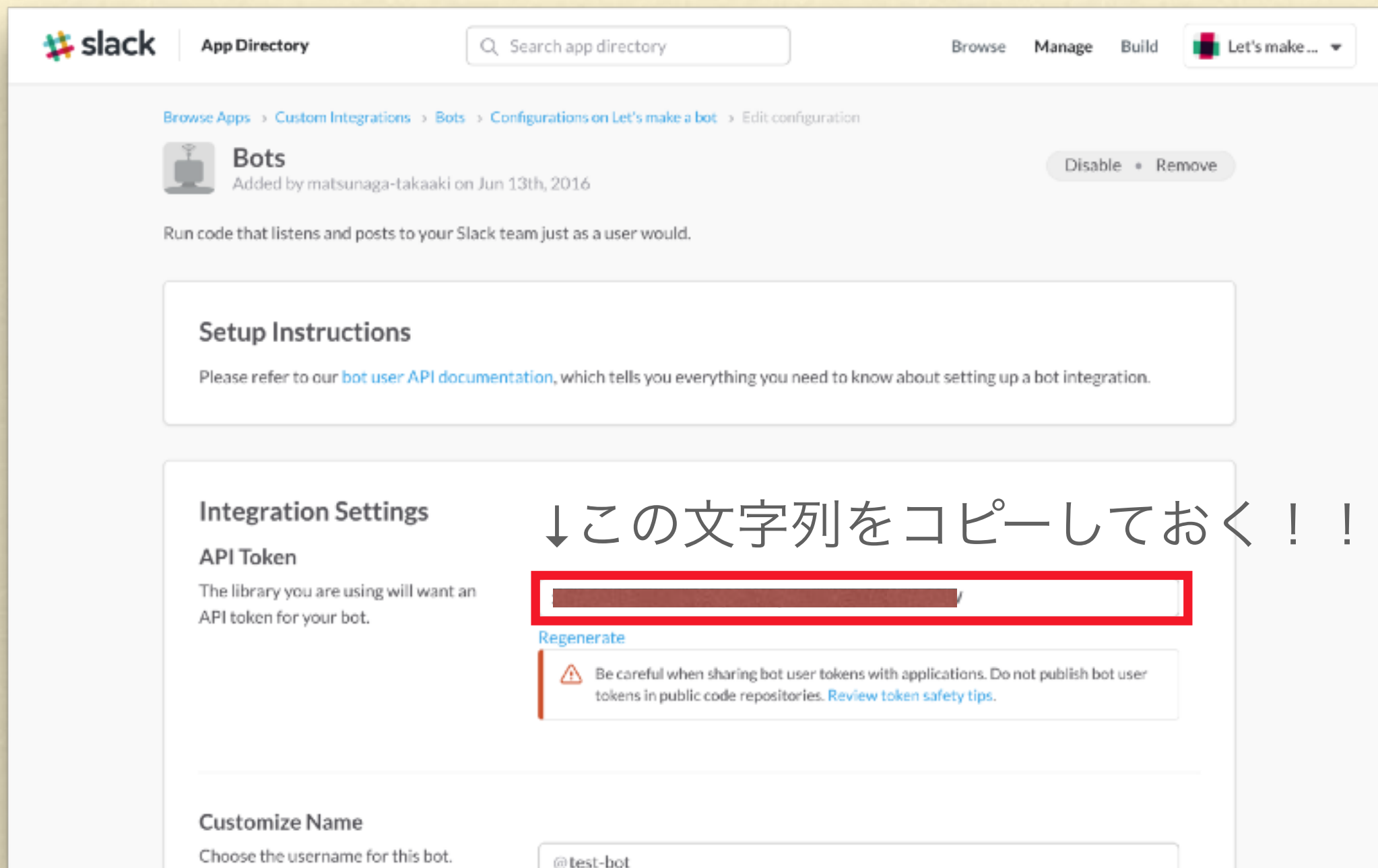
Username must be all lowercase. They cannot be longer than 21 characters and can only contain letters, numbers, periods, hyphens, and underscores. Most people choose to use their first name, last name, nickname, or some combination of those with initials.

Add bot integration

←入力したらクリック

By creating a bot integration, you agree to the [Slack API Terms of Service](#).

# Slackでbotを動かそう



slack App Directory Search app directory Browse Manage Build Let's make...

Browse Apps > Custom Integrations > Bots > Configurations on Let's make a bot > Edit configuration

**Bots**  
Added by matsunaga-takaaki on Jun 13th, 2016

Disable • Remove

Run code that listens and posts to your Slack team just as a user would.

**Setup Instructions**  
Please refer to our [bot user API documentation](#), which tells you everything you need to know about setting up a bot integration.

**Integration Settings**

**API Token**  
The library you are using will want an API token for your bot.

↓この文字列をコピーしておく！！

Regenerate

⚠ Be careful when sharing bot user tokens with applications. Do not publish bot user tokens in public code repositories. [Review token safety tips.](#)

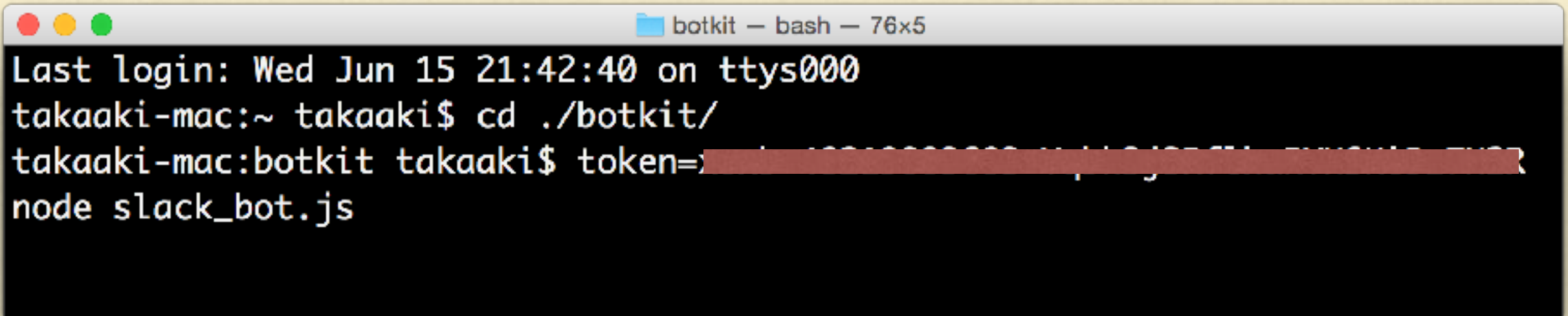
**Customize Name**  
Choose the username for this bot.

@test-bot



# Slackでbotを動かそう

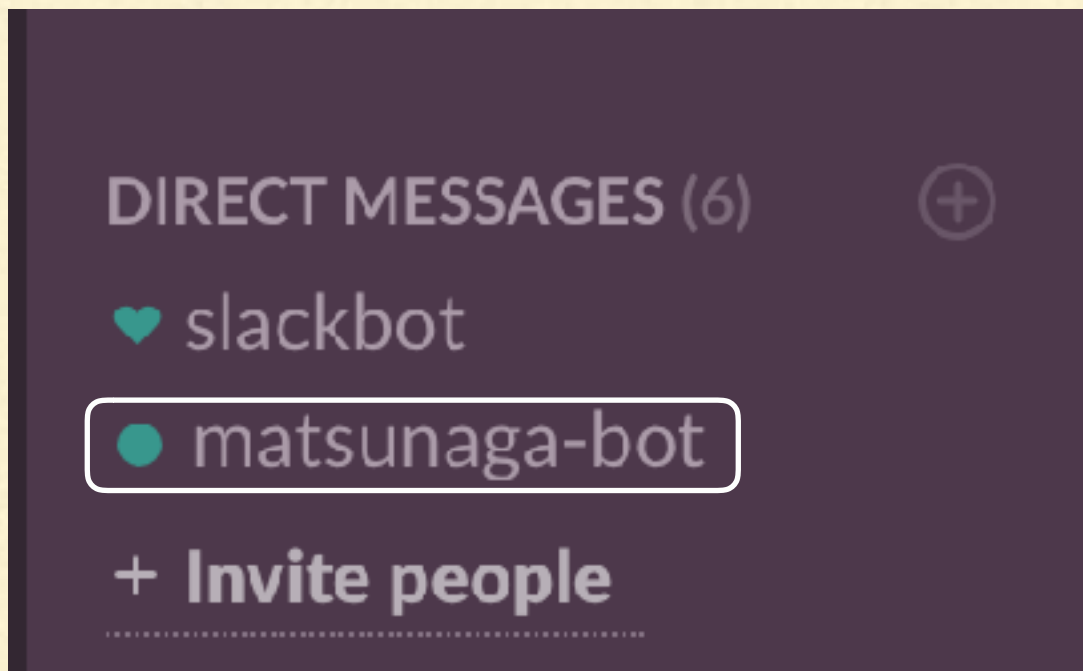
実際に動かしてみよう！！

A terminal window titled 'botkit - bash - 76x5' with a dark background and light text. The window shows the following text: 'Last login: Wed Jun 15 21:42:40 on ttys000', 'takaaki-mac:~ takaaki\$ cd ./botkit/', 'takaaki-mac:botkit takaaki\$ token=[REDACTED]', and 'node slack\_bot.js'. The token is obscured by a red rectangular box.

```
botkit - bash - 76x5
Last login: Wed Jun 15 21:42:40 on ttys000
takaaki-mac:~ takaaki$ cd ./botkit/
takaaki-mac:botkit takaaki$ token=[REDACTED]
node slack_bot.js
```

```
$ token=[コピーした文字列] node slack_bot.js
```

# Slackでbotを動かそう

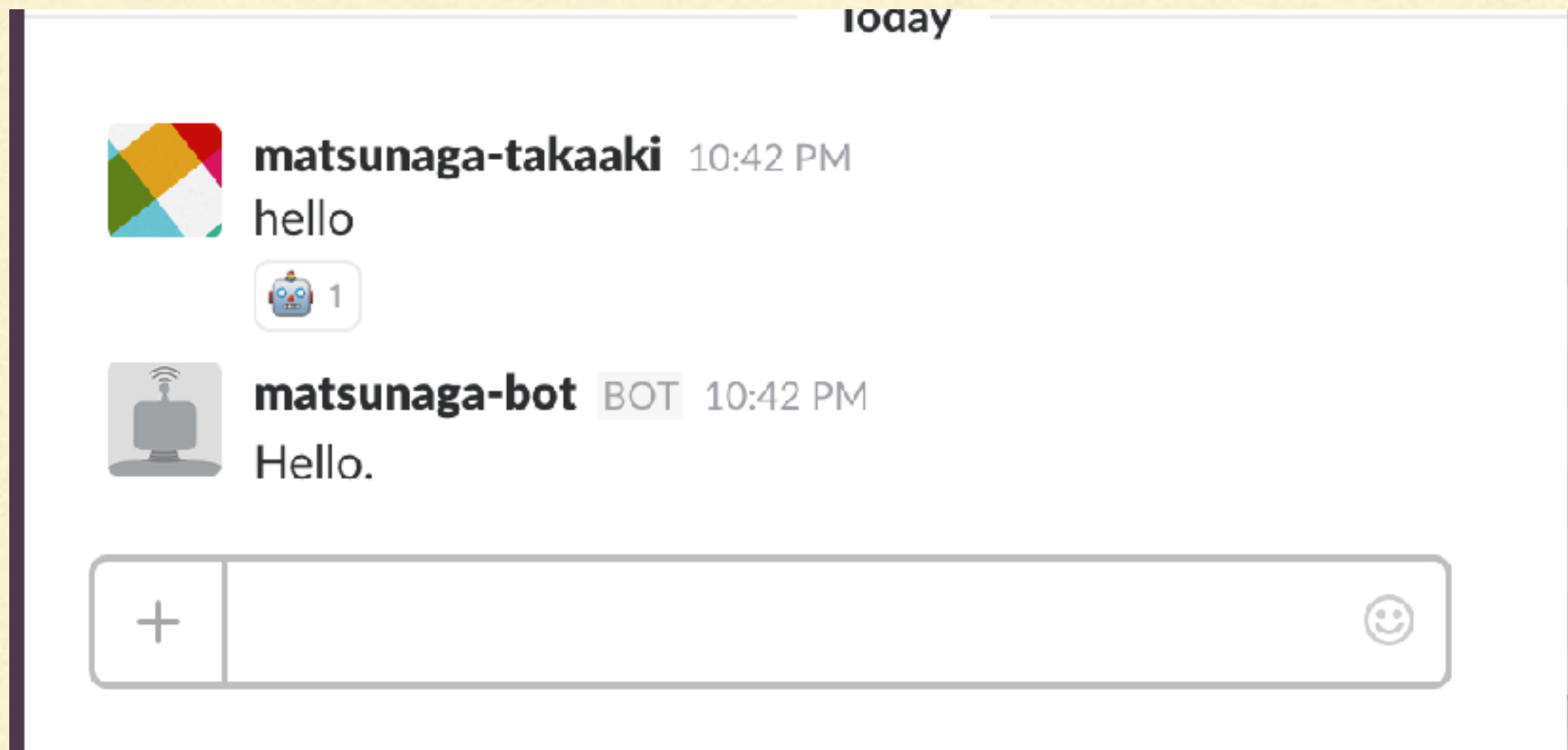


```
botkit — node — 57x30
debug: PONG received
debug: PONG received
debug: No handler for tick
debug: No handler for tick
debug: No handler for tick
debug: No handler for tick
debug: PING sent
debug: No handler for tick
debug: PONG received
debug: PONG received
debug: No handler for tick
debug: No handler for tick
debug: No handler for tick
debug: No handler for tick
debug: PING sent
debug: No handler for tick
debug: PONG received
debug: PONG received
debug: No handler for tick
debug: No handler for tick
debug: No handler for tick
debug: No handler for tick
debug: PING sent
debug: No handler for tick
debug: PONG received
debug: PONG received
debug: No handler for tick
debug: No handler for tick
debug: No handler for tick
```

追加したbotのアイコンが  
光ってればOK!!

# Slackでbotを動かそう

「hello」か「hi」と打ってみて、返信があれば成功！





---

# BOT作成手順

---

1. Node（ノード）.jsのインストール
  2. Botkit（ボットキット）の導入
  3. Slackでbotを動かそう
  - 4. オリジナルの返答を追加**
-

---

# オリジナルの返答を追加

---

botkitフォルダ内の「slack\_bot.js」を開いて、  
以下のテンプレを追加する。（Slackで配布します）

```
↓  
controller.hears('コマンド', 'メッセージの種類', ↓  
  //コマンドに対する処理↓  
  function(bot, message) {↓  
    bot.reply(message, '返事');↓  
  }↓  
);↓  
↓
```

# オリジナルの返答を追加

～実装のテンプレート構成～

```
↓
controller.hears(第一引数, 第二引数, ↓
  //コマンドに対する処理↓
  function(bot, message) {↓
    bot.reply(message, '返事');↓
  }↓
);↓
↓
```

第三引数



---

# オリジナルの返答を追加

---

## 第一引数：コマンド

処理のトリガーとなるコマンドを、文字列か文字列配列で指定する。

例) `controller.hears( ['hello','hi'], ...`

上記の場合、「hello」もしくは「hi」に反応する。

---

# オリジナルの返答を追加

---

## 第二引数：メッセージの種類

どういうメッセージに対して反応するかを、カンマ区切りの文字列で指定する。

message_received	： 全てのメッセージに対して反応
direct_mention	： 直接メンションされた場合 例) @bot hello
direct_message	： 1 対 1 でメッセージが送信された場合
mention	： メンションされた場合 例) hello @bot

# オリジナルの返答を追加

## 第三引数：コマンドに対する処理

```
controller.hears( 'hello' , 'direct_message',  
    function(bot, message) {  
        bot.reply(message, 'Hello.');  
    }  
);
```

上記の場合、「hello」というコマンドに対して『Hello』という返事を返す処理が実装されています。



---

# オリジナルの返答を追加

---

というわけで、

**オリジナルの返事を  
追加してみましよう！！**

---

---

完

---