How to create your own Telegram bot launcher
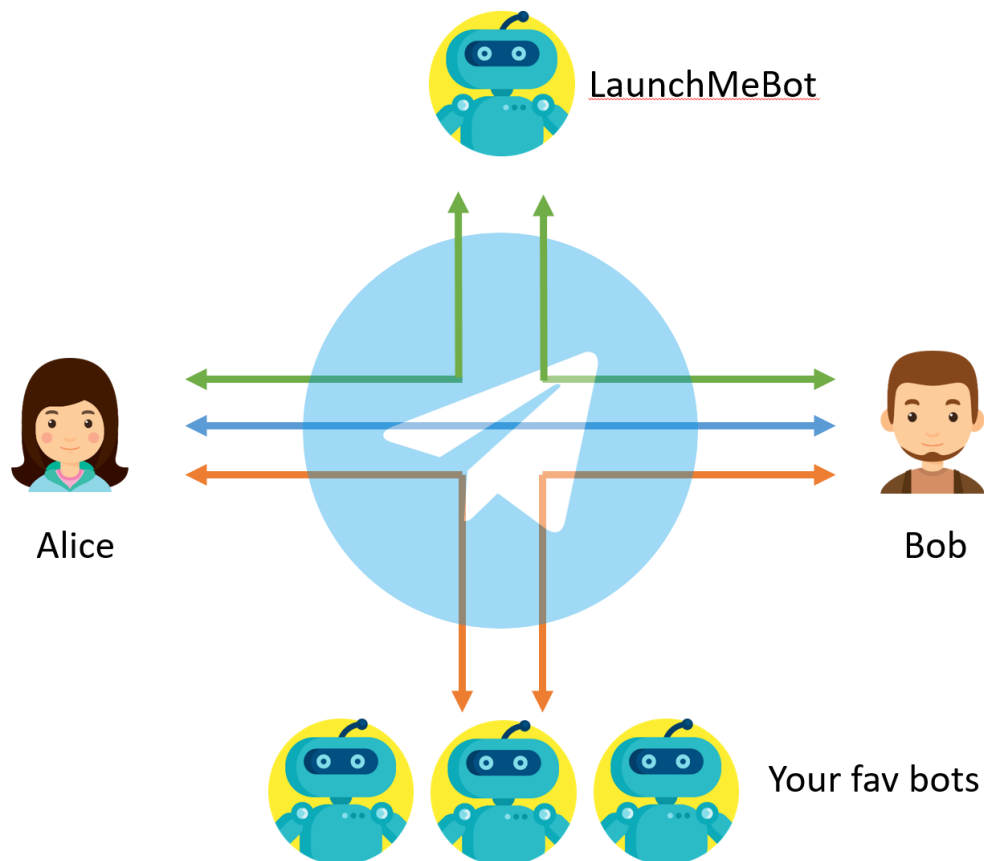
This is an introduction to creating your own Telegram bot and writing some code to make your bot do stuff for you. Let's create a bot launcher that will list all your favourite bots and launch them with just one click – no more searching or trying to remember their names.


Step 1: Set up Telegram client

Step 2: Create a Telegram bot

Step 3: Write and deploy code for your bot

Let's get started!

Step 1: Set up Telegram client

Use this link https://web.telegram.org/#/login to log in to your Telegram account. While it is possible to use Telegram on mobile client, using the web client is more convenient because we will need to copy the bot token and paste it into your code later. Once you have set up the client and logged in, you will be connected to the Telegram server and ready for the next step.
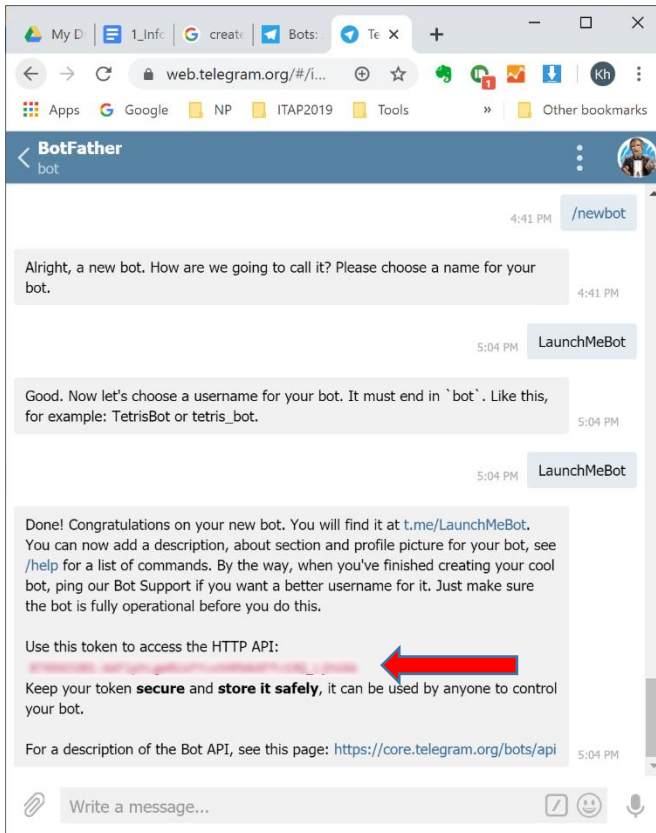

User   Telegram clients   Telegram server
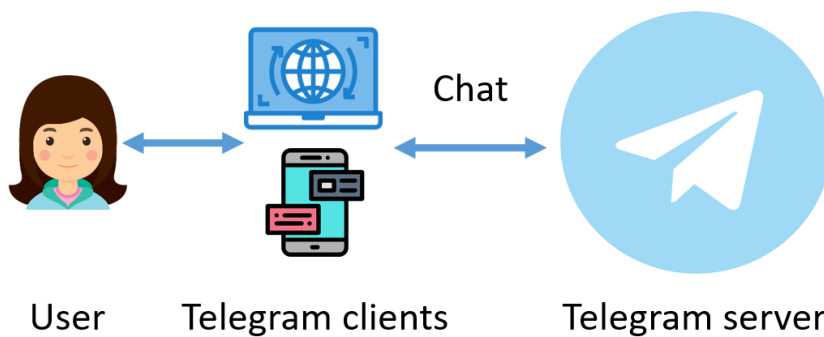
Step 2: Create a Telegram bot

Let's use a Telegram bot called "BotFather" to help us create our bot. On your Telegram web client, click Search and look for "BotFather". Click Start or send command /start to BotFather.
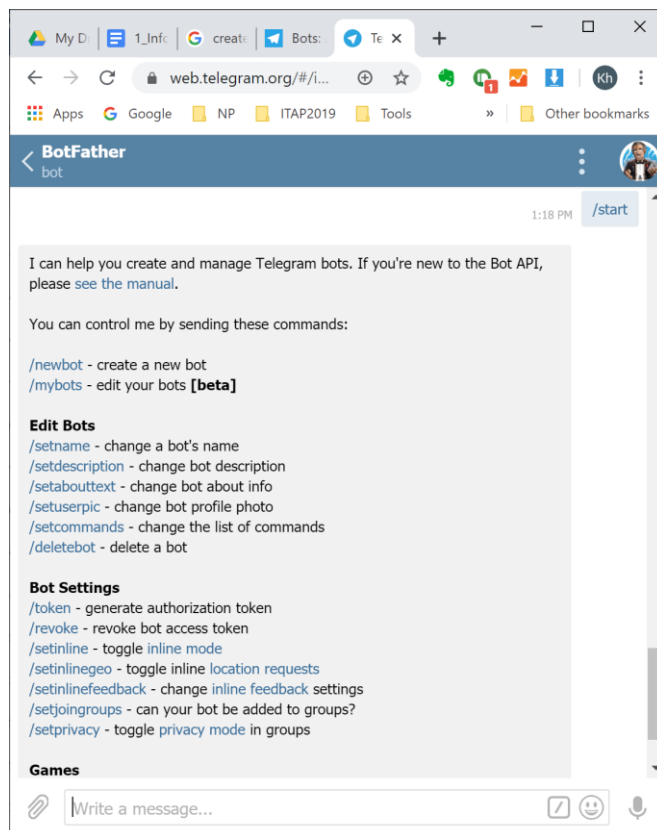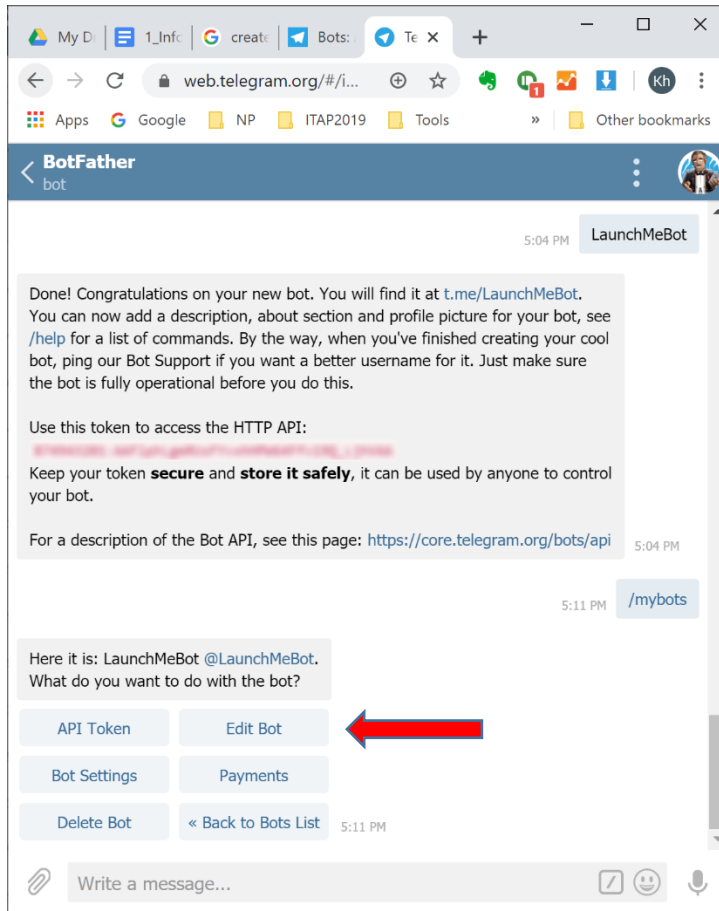
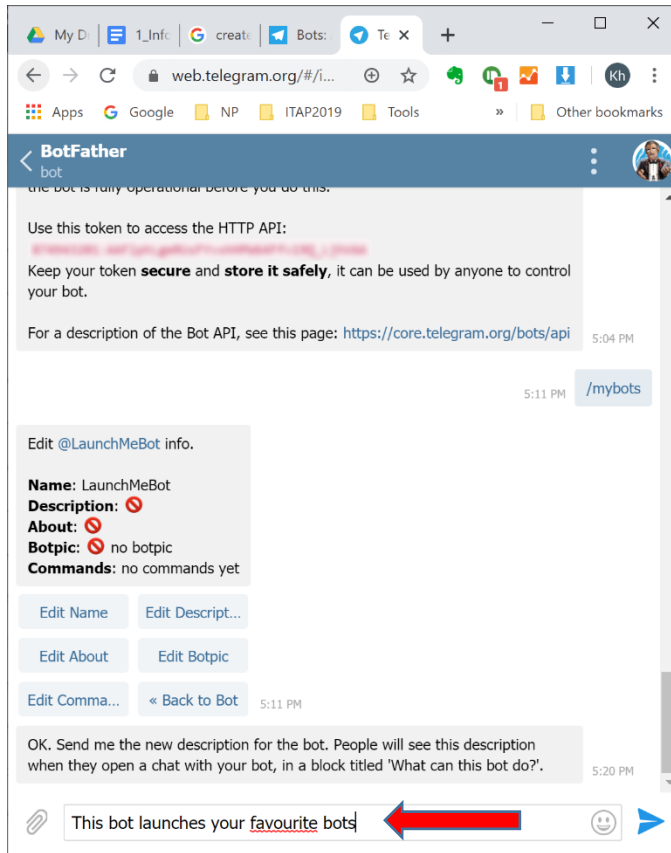Send /newbot to create our bot. Choose Name and Username for your bot

Copy your token and save it in a txt file.

Send /mybots to BotFather. Click Edit Bot to edit the bot that you had just created.
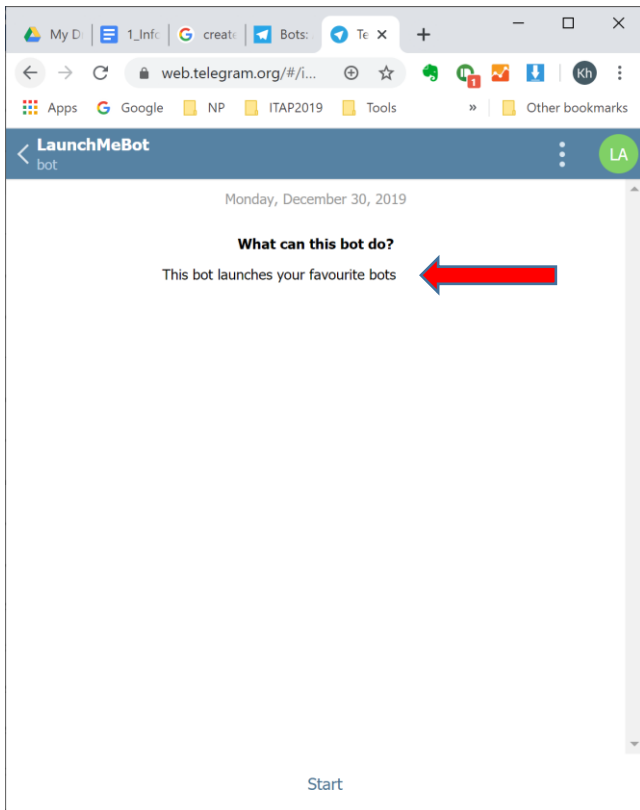
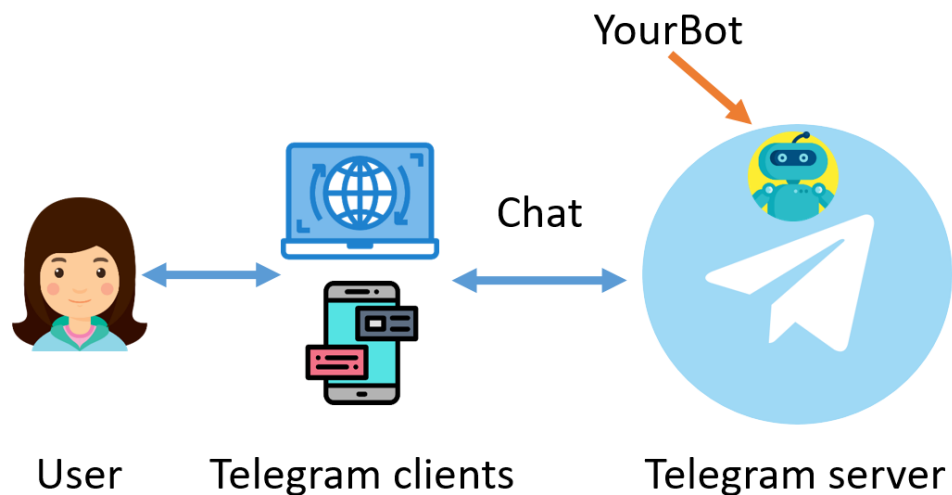Enter the description of your bot and send it to BotFather.

Click Search and look for the name of your bot. Click Start or send /start to your bot.

Nothing happens at this moment because there is no code associated with this bot.

Your bot is now available on Telegram and we will write some code to make it do some work for us.

Step 3: Write and deploy code for your bot

Go to Google Drive and sign in using your Google account. Click New -> More -> Connect more Apps

Search for Google Apps Script and click Connect. Click New -> More -> Google Apps Script

Specify your project name (top left). Input the code below (using your own bot token)
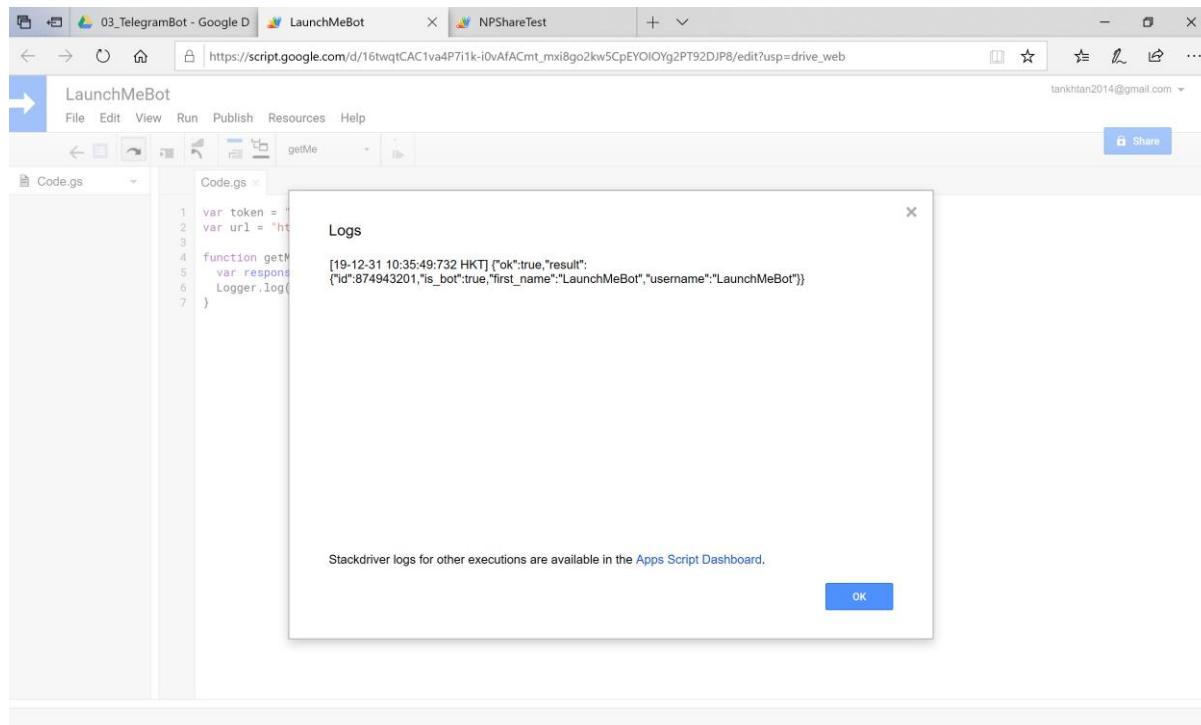
GitHub link to download code

var token = "your bot token";

var url = "https://api.telegram.org/bot" + token;

function getMe() {

  var response = UrlFetchApp.fetch(url + "/getMe");

  Logger.log(response.getContentText());

}
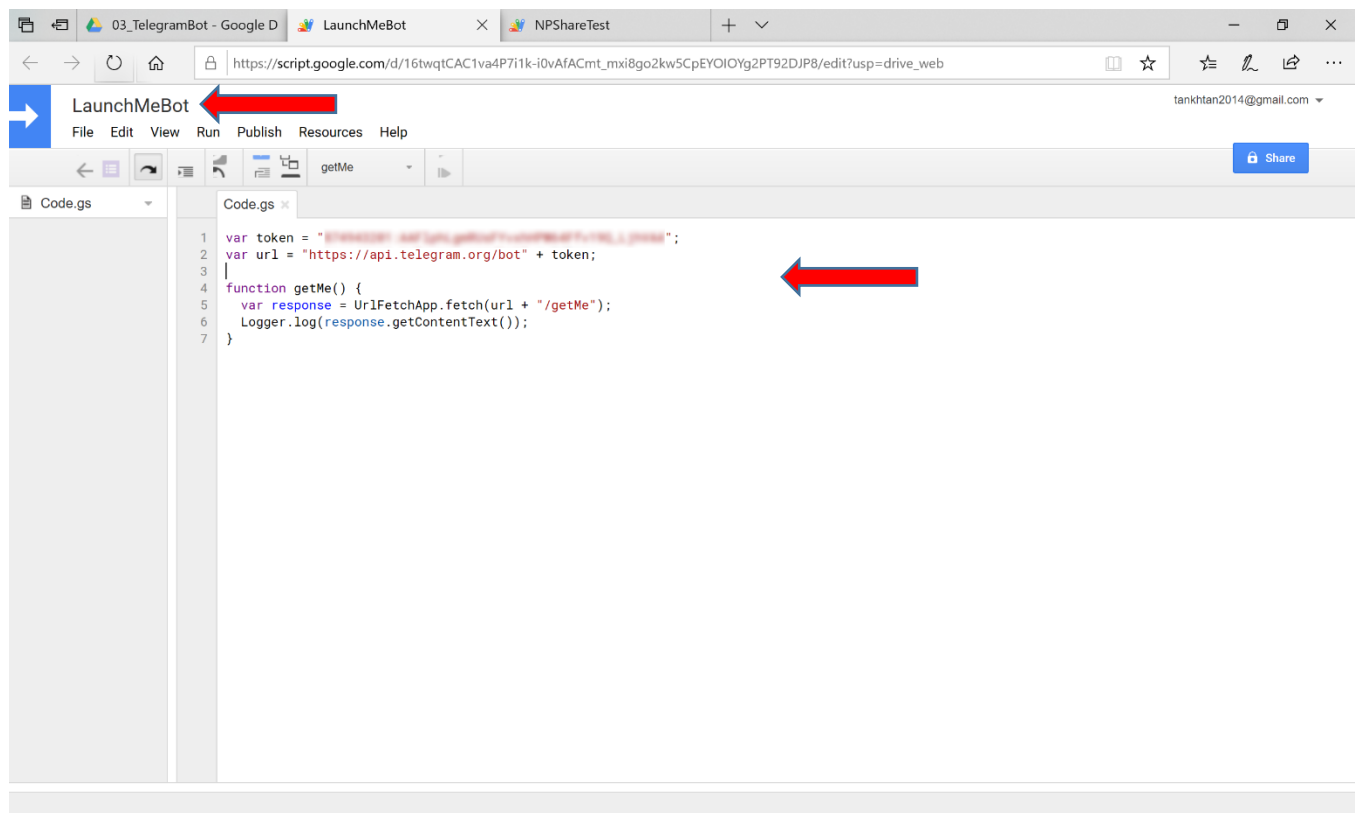
Click Run -> Run function and select getMe. Review permissions and allow access.

To check whether your bot is running - Click View -> Logs

If you see this in the log, that means the token and the url are set up correctly. Let's publish the code that you have written.

Publish your code - Apps Script

Click Publish -> Deploy as web app…

After setting the version and access rights, click Deploy

Your code is now deployed. Let's copy the web app URL into your txt file because we will need this URL to set up the webhook.

Modify the code – add webhook

Setting a webhook means you are supplying Telegram with a location in the form of a URL. This URL is used by Telegram to send updates to your bot. Let's modify your code by adding the highlighted text below.
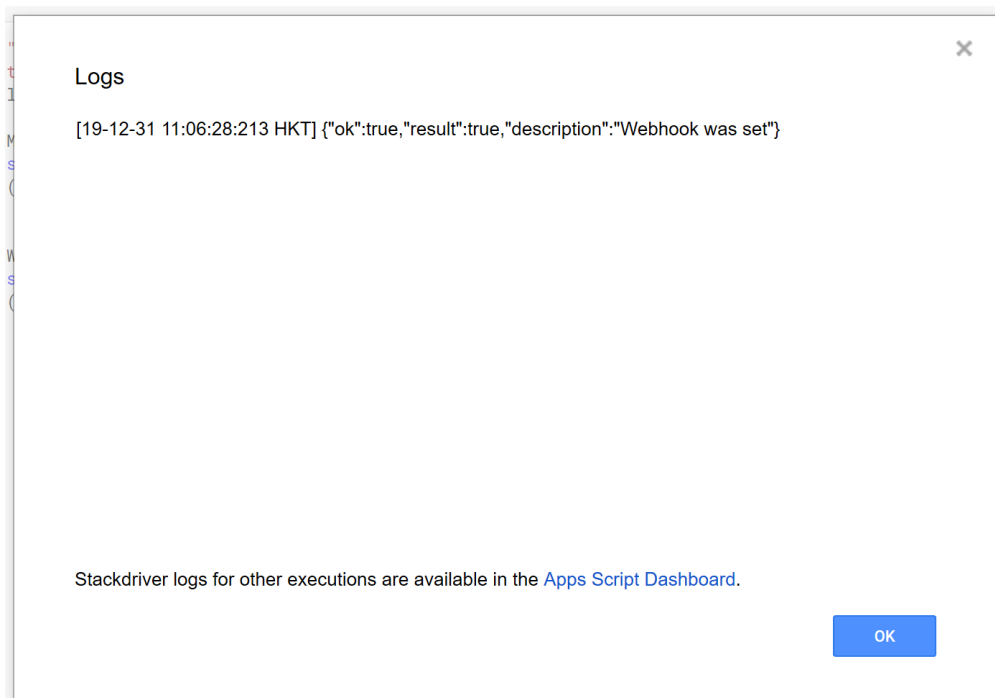
GitHub link to download code

```
var token = "your bot token";

var url = "https://api.telegram.org/bot" + token;

var webAppUrl = "your webAppUrl";


function getMe() {

  var response = UrlFetchApp.fetch(url + "/getMe");

  Logger.log(response.getContentText());

}


function setWebhook() {

  var response = UrlFetchApp.fetch(url + "/setWebhook?url=" + webAppUrl);

  Logger.log(response.getContentText());

}
```

Click Run -> Run function and select setWebhook

To check whether webhook is set - Click View -> Logs

Logs

[19-12-31 11:06:28:213 HKT] {"ok":true,"result":true,"description":"Webhook was set"}

Stackdriver logs for other executions are available in the Apps Script Dashboard.

OK

If you see this in the log, it means that your webhook was set up properly.

Modify the code – implement keyboard

Modify your code by adding the highlighted text below. [GitHub link to download code](#)

var token = "your bot token";

var url = "https://api.telegram.org/bot" + token;

var webAppUrl = "your webAppUrl";


```
var mainMenu_kb = {
 "inline_keyboard": [
   [{
    "text": "NPShareBot",
    "callback_data": "NPShareBot",
    "url": "https://t.me/NPShareBot"
   }],
   [{
    "text": "BotFather",
    "callback_data": "BotFather",
    "url": "https://t.me/BotFather"
   }],
   [{
    "text": "SG Food Deals",
    "callback_data": "SGfooddeals",
    "url": "https://t.me/SGfooddeals"
   }]
 ]
};


function doPost(e) {
 try {
   var contents = JSON.parse(e.postData.contents);
   var message = contents.message;
```

```javascript
    var resp = sendTextWithButtons(message.from.id, "Select bot to launch", mainMenu_kb);
  }
  catch (err) {
    Logger.log(err);
  }
}


function sendTextWithButtons(chatId, text, keyboard) {
  var data = {
    method: "post",
    payload: {
      method: "sendMessage",
      chat_id: String(chatId),
      text: text,
      parse_mode: "HTML",
      reply_markup: JSON.stringify(keyboard)
    }
  };
  var response = UrlFetchApp.fetch('https://api.telegram.org/bot' + token + '/', data);
  return response.getContentText();
}


function getMe() {
  var response = UrlFetchApp.fetch(url + "/getMe");
  Logger.log(response.getContentText());
}


function setWebhook() {
  var response = UrlFetchApp.fetch(url + "/setWebhook?url=" + webAppUrl);
  Logger.log(response.getContentText());
```
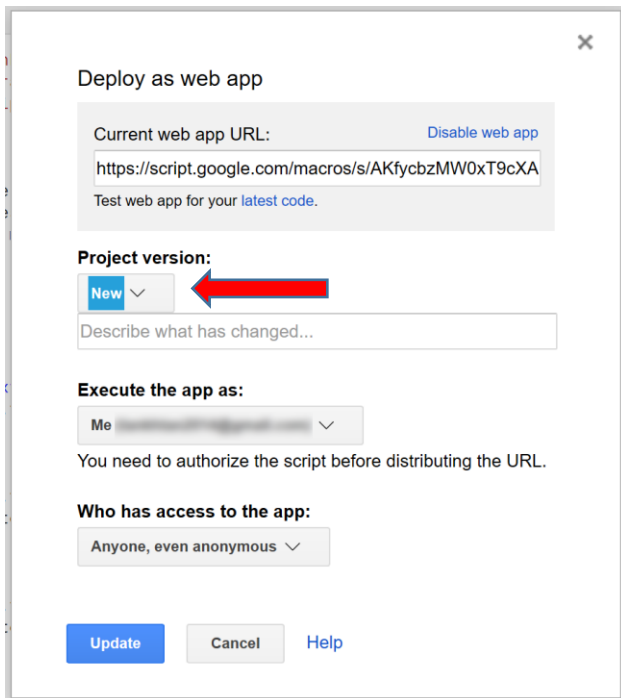
}

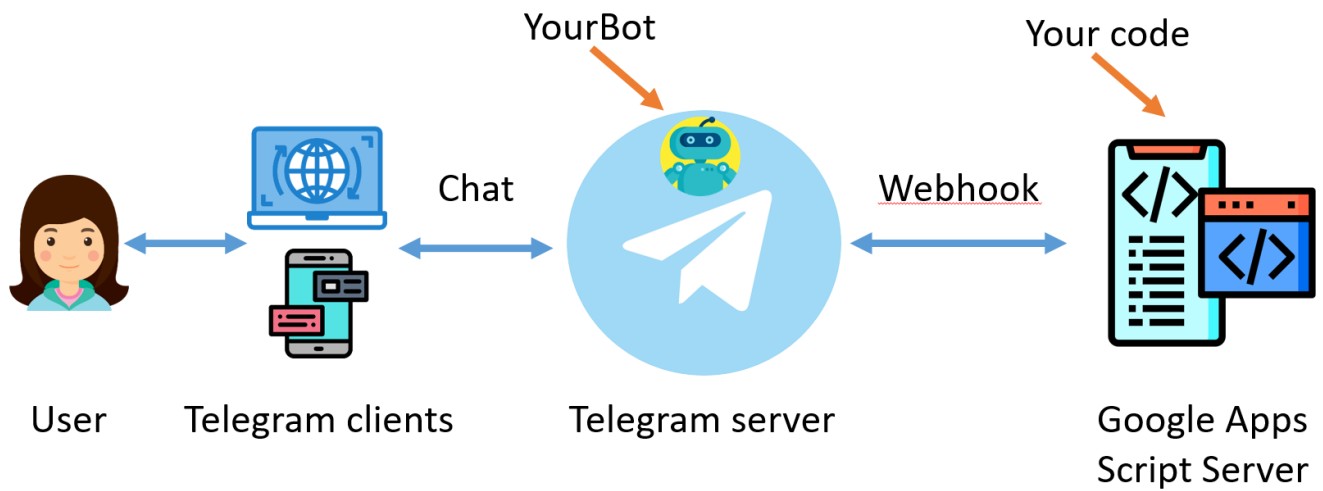Deploy the modified code. Ensure that the Project version is New. Click Update.

When you send /start to your bot, the doPost() function will be executed. This function tells your bot to reply to your command with 3 buttons, each will launch a different bot when clicked.
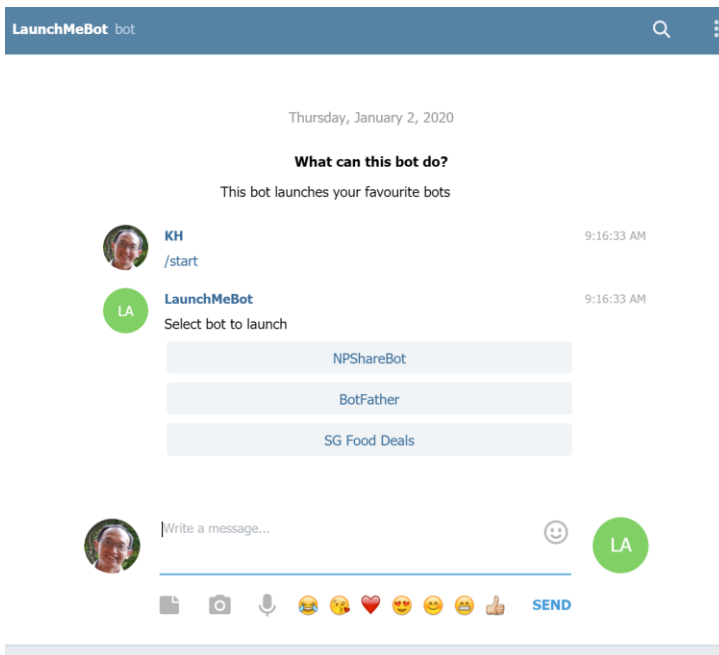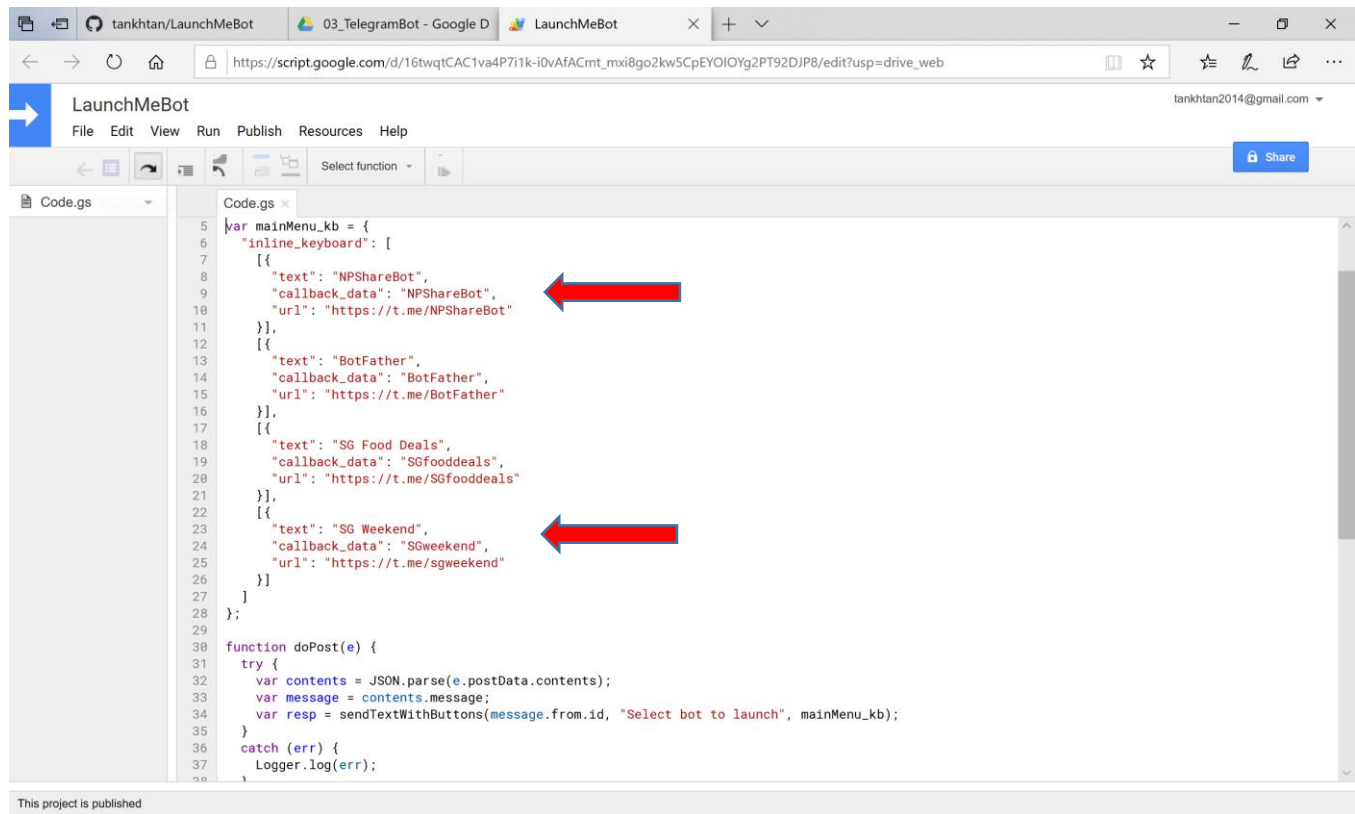
Congratulations! You have completed all 3 steps and this is what you have implemented.

Here's a summary of what you have done:

- Logged in to Telegram
- Created your bot using BotFather
- Saved your bot token
- Modified your bot's properties e.g. description
- Created code for your bot using Google Apps Script
- Set up webhook
- Deployed your code
- Tested your bot

You can add more buttons or select other bots or other channels by modifying the text, callback_data and URL fields of mainMenu_kb

To explore more features of bots, please refer to bot API.

Hope you have found this tutorial useful. If you have any questions, let me know.

Tan Kok Hui

The Sandbox / Makers' Academy

https://t.me/tankhtan