من الالف للياء
FROM A Z
From A to
V13
BOT

المقدمة	من الصفحة* 3-4
الأساسيات	من الصفحة* 4-6
الرد التلقائي	من الصفحة* 6-13
الإمبيد	من الصفحة* 13-17
المتغيرات	من الصفحة* 17-19
علامات التحديد	من الصفحة* 19-20
السلاش كوماند هاندلر	من الصفحة* 20-34
الأزرار	من الصفحة* 34-41
السيليكت منيو	من الصفحة* 41-51
10 وظايف	من الصفحة* 51-57
كلام / اختيارت [عشوائية]	من الصفحة* 57-65
الأخطاء	من الصفحة* 65-71
كود مهم	من الصفحة* 71-76
كودات	من الصفحة* 76-111

المقدمه

الـ Discord بيقدم تجربة تواصل وترفيه مميزة، ومع وجود مجتمعات كتير فيه، يمكن تصميم بوتات خاصة لتلبية احتياجات المجتمعات دي. وبعد ما نزلت النسخة الـ 13 من مكتبة Discord.js، صار المطورين يقدروا يصمموا بوتات بشكل أسهل وأكثر فاعلية. ولذلك، يسرني أن أقدم لكم كتاب "من الألف للياء مع Discord.js"، والذي يشرح بالتفصيل كيفية بناء بوتات Discord باستخدام Discord.js v13، ويتناول جميع المواضيع اللازمة لتطوير بوتات متطورة بسهولة. سيتم تغطية جميع الجوانب الرئيسية لبرمجة البوتات، بما في ذلك تصميم وتفعيل الأوامر، والتفاعل مع قواعد البيانات، والتعامل مع الرسائل والأحداث والمزيد. دعونا نتعلم سويًا كيفية تصميم بوتات Discord لتحسين تجربة المستخدم وتعزيز مجتمعاتنا،هنتلعلم ازاي نعمل بوت ب javascript ب مكتبه discord.js النسخه 13 بمحرر الاكواد replit

تأليف

علي محمد علي

1-خش موقع replit.com

2-اعمل بوت من موقع discord developer وتحط اي اسم مثلا test وبعد كده تحط الصوره وخش عند bot واعمل add bot وفعل PRESENCE INTENT و SERVER MEMBER واعمل MESSAGE CONTENT وانسخ التوكن save

3-وارجع ل موقع replit.com

4-خش على shell واكتب npm install discord.js@13 الامر ده بيخليك تنزل v13 عشان تبرمج بيها وتستخدم اداواتها

5-روح لل index.js وحط الكود ده

```const { Client, Intents } = require('discord.js');

const client = new Client({ intents:
[Intents.FLAGS.GUILDS,
Intents.FLAGS.GUILD_MESSAGES] });```
```

وشرحه ببساطه هو كود سورس سطرين بتحطه عشان تظبط الدنيا شرحه---

1-const { Client, Intents } = require('discord.js');

ذا بيثبت انك استخدمت اداه او يعرف انك استخدمت اداه لو ما حطيتها واستخدمت اداه معينه ما بيشتغل بيطلعلك ايرور
فمثلا ما تقدر تسوي ايمبيد غير لما تكتب الامر
const { MessageEmbed } require('discord.js');
والمثال الموجود داخل الكود بيعرف ان الclient تبع مكتبdiscord.js مش اي مكتبه اخرى والIntents ذي الصلاحيات تبع البوت بشرحها دلوقت

2-const client = new Client({ intents: [Intents.FLAGS.GUILDS, Intents.FLAGS.GUILD_MESSAGES] });
السطر ده بقا هو الintents الي قولت عليه الintents الي بقا الغرض او الوظيفه بتاعت البوت الي بتدي البوت الصلاحيات الي تخليه يوصل للسيرفر مثلا انه يضيف ريكاشن على الرسالة يرد على الرسائل يقرا الرسائل الموجوده يرسل رسايل في الخاص ،الخ في السطر ده في حجتين بس الي هما الي الاوله الي هي Guilds ودي بتمكن البوت انه يوصل لمعلومات السيرفر. مثال: عدد الاعضاء ، القنوات، رتب ،الخ

ونلاحظ انها معموله GUILDS مش GUILD وده الي بيخلي البوت يوصل لاكتر من سيرفر

والثانية الي هي GUILD_MESSAGES ودي بتخلي اليوت يقرا الرسايل المبعوته على السيرفر

كده عرفنا اول سطرين اساسين في بدايه اي بروجكت نخش بقا في الجد هنبدا اول حاجه ازاي تعمل رد تلقائي

AUTO REPLAY | الرد التلقائي

VERSION 13(V13)

دلوقتي هنعرف ازاي نعمل رد تلقائي ب discord.js v13 اول حاجه ناخد كود مثال <<

```
const { Client, Intents } = require('discord.js');

const client = new Client({ intents:
[Intents.FLAGS.GUILDS,
Intents.FLAGS.GUILD_MESSAGES] });
```

```
client.on('messageCreate', message => {
```

```
if (message.content === 'السلام عليكم') {
  message.reply('وعليكم السلام ورحمة الله وبركاتة نورت💖');
}
```

```
});

client.login(process.env.token); >>>
```

ده كود رد تلقائي لما حد بيكتب السلام عليكم البوت بيرد ب وعليكم السلام ورحمة الله وبركاتة نورت💖 وده شرح الكود

```
1-const { Client, Intents } = require('discord.js');

const client = new Client({ intents:
[Intents.FLAGS.GUILDS,
Intents.FLAGS.GUILD_MESSAGES] });
```

دول انا لسه قيله والسطرين دول لازم اساسيين في اي كود يلا نخش على الجديد

```
client.on('messageCreate', message => { -2
```

السطر ده بقا مهم جدا قبل ما تعمل اي رد او اي تفاعل لازم تحطه بس هتغير فيه شويه لو تفاعل هيبقا interactionCreate لسه هنقولها بعدين بس خلينا في الرد التلقائي بيجي بقا السطر ده معانا ثابت لازم يتحط في اول اي رد تلقائي وشرحه بالتفصيل الي هو client يعني العميل او البوت فانت كده لما تكتب client بتضيف امر جديد وغالبا لما

تعمل امر لازم on بعد client اه في حاجات تانيه زي off او once او emit وغيرهم بس الاساس في discord.js هي on فكده شرح client.on الي هو بيعرف امر جديد للبوت زي مقولنا. بعدين فتحت قوس والقوس ده مبيتقفلش غير لما نخلص الامر الي هتكتبه يعني تقفله في الاخر خالص بعد كده فتحت علمتين تنصيص الي هما ' ' دول عشان نحدد النص عشان كده اسمهم علامات تنصيص وفي غيرهم مثلا " " او ` ` بس بيختلفوا في حاجات بسيطه هشرحها بعدين وبعد ما حطيت علامات التنصيص حطيت جواهم الوظيفة اي هي messageCreate دي وظيفه موجوده عشان تعمل رد تلقائي مش شرط يكون رد تلقائي بس مش هخش في التفاصيل الزيادة عشان محدش يتلغبط في وظايف تانيه غيرها كتير وبعد messageCreate عملنا, دي عشان تكمل الوظيفه وعشان كده حطينا بعدها message الي هي معنها رسالة وهي بتكمل messageCreate وبعديها حطينا => وده بيحدد استخدام الوظيفه يعني مثلا انا كتبت messageCreate هستخدمها في ايه بقا دي هي وظيفه ال=> بتحدد اهميه الوظيفه الي هتستخدمها وعشان كده احنا بنحددها في قوسين عشان كده فتحت القوس ده {

2- if (message.content === 'السلام عليكم') { }

وبعد كده طالما فتحت قوس هنزل السطر الي بعده واول حاجه هكتبها فيه هي if وif ديه لوحدها قصه بس عشان الخص if يعني لو او بتضيف شرط الي لو حدث معين حصل

فحدث تاني يحصل يعني مثلا لو حد كتب اهلا البوت يرد عليه باهلا حدث وعليه نتائجة او شرط وعليه نتائجة فدة شرح مبسط لif وبعديها هنفتح قوس طبعا عشان if فطبعا لازم نحط فيه وظيفه معينه وفي الرد التلقائي ممكن تتغير على حسب الرغبة فمثلا في كذا وظيفه اول حاجه هي message.content ودي عشان الرد التلقائي او الامر يشتغل لازم تتكتب الامر لوحده في رسالة يعني لو كتبت اهلا السلام عليكم ورحمة مش هتشتغل لازم تكتب السلام عليكم بس من غير حاجه قبلها ولا بعدها وفي بردوا message.content.includes دي لو الرسالة احتوت على الكلمه يعني لو واحد كاتب رسالة 100 سطر وفي فيها كلمه السلام عليكم البوت هيرد عليه ويقول وعليكم السلام ورحمة الله وبركاتة على حسب الرد والامر الي انت كتبه جوا الكود وفي اخر حاجه message.content.startsWith دي بقا شرطه انك تبدا بالامر يعني بس تكتب في اول الجمله السلام عليكم لو كتبت بعديها الف سطر بردوا هيشتغل كده قولنا على التلات وظايف ايوه في غيرهم زي انك تبدا ببريفكس بس خلينا في دول دلوقت انا عملت جدول شرح ليهم يلخص التلات وظايف دول والجدول موجود تحت الصفحه دي علطول

وضفت وظيفه جديده في الصورة الي هي message.mention.has(")) وشرحه موجود في الصورة

3-message.reply('وعليكم السلام ورحمة الله وبركاتة نورت 💖');

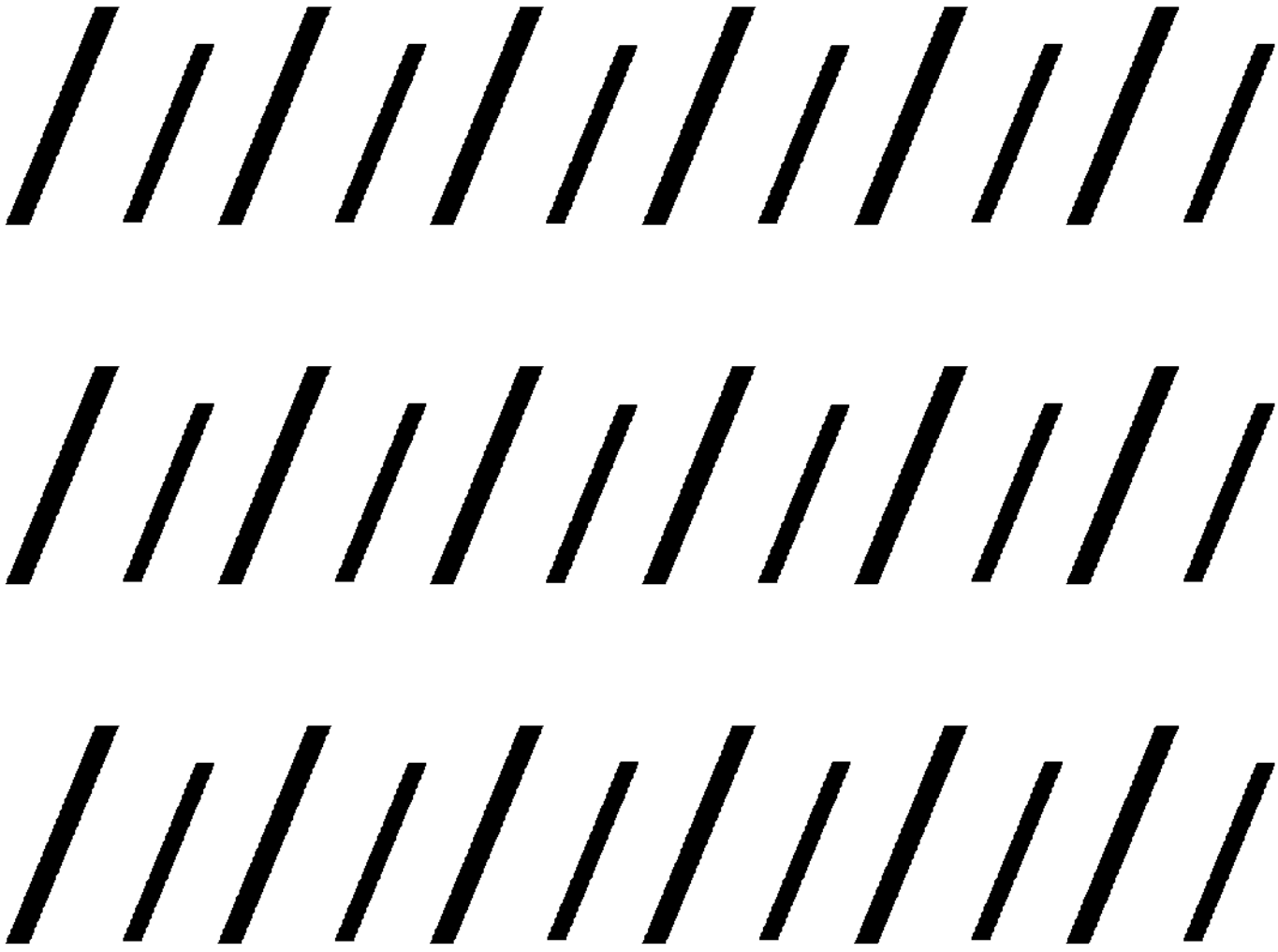
اول حاجه بديت بيها هي message.reply وديه وظيفه عشان ترد على الرسايل وممكن تبدلها ب message.channel.send الفرق بينهم ببساطة ان message.reply بترد على الرسالة يعني بتمنشن الرسالة او reply اما message.channel.send تبعت رسالة لوحدها من غير ما تذكر اي رسايل عموما بعدها فتحنا قوس عشان اي وظيفه لازم نفتح بعدها قوس وبعديها طالاما هنحط رسالة هنحط علامتين تنصيص وبعد كدا كتبت الرسالة وهنقفل القوس ونحط ; والعلامة دي بتستخدم عشان ننهي وظيفه او نحط فاصل بين الاوامر والوظايف وكدا

4-بعد كده هنقفل القوس الي هي بتاع وظيفه message.reply لما فتحنا قوس عشان نحط وظيفه جديده

5-بعد كده هنحط نهاية اي كود الي هي });(واول حاجه فيهم دي قفل قوس بداية انشاء الرسالة الجديده او الوظايف الجديده بعديها هنقفل قوس بدايه حدث messageCreate واخر حاجه هنحط ; وزي مقولت دي بتبقا في نهاية كود او ظيفتين متعلقين ببعض

6-وبعد ما خلصنا الكود هنحط تعريف البوت تعريف الclient

الي هنعرفه ب client.login ممكن كنت حطيت client.login('التوكن بتاع البوت هنا') بس دي مش امان فعشان كده هنحطه في السيكريتس وعشان نحط اي حاجه جوا السيكريتس بنحط (process.env.اسم السيكريت) ونخش على الsecerts ونحط في الname اسم السيكريت الي احنا كتبناه في الكود بتاعي ده هنكتب token وهنحط جوا الvalue التوكن بتاع البوت وكدا خلصنا ازاي نعمل رد تلقائي يلا نشوف الي بعدوا



EMBED|الايمبيد

هنحط مثال على ايمبيد دلوقتي وهيبقا مطول هيبقا ايمبيد طويل

```
client.on('messageCreate', message => {
  if (message.content === 'ايمبيد') {
    // Create a new embed object
    const embed = new MessageEmbed()
      .setTitle('ايمبيد جديد')
      .setDescription('هذا ايمبيد جديد')
      .setColor('#0099ff')
      .addField('حقل جديد', 'هذا حقل جديد')
      .setThumbnail('image.com')
      .setImage('https://example.com/image.png')
      .setFooter('Footer text', 'https://
```

```
example.com/footer.png');

    message.channel.send({ embeds: [embed] });
  }
});
```

الشرح

1-client.on('messageCreate', message => {
 if (message.content === 'ايمبيد') {

لسه شرحها فوق ازاي تعمل رد تلقائي

2-const embed = new MessageEmbed()
صنع ايمبيد جديد هنلاحظ انه عشان نعمل متغير او نعرف حاجه بنستخدم const وفي بردوا var و let هنشرح كل ده بعدين عموما هنا بنعرف الايمبيد ب const وعشان نستخدمه في سيرته الاصلية بنحطها بعد = ونكتب new يعني جديد وبعديها MessageEmbed واحنا المفرود نعرف الMessageEmbed الاول بالسطر ده

```
const { MessageEmbed } = require('discord.js');
```

وكدا عرفنا الmessageEmbed بdiscord.js وبعد كده فتحت قوسين () عشان المحرك يفهم ان انت سبهتا فاضيه وهتعرفه تحت

3- .setTitle('ايمبيد جديد')
ده بيحدد العنوان طبعا ممكن تغيره للانت عايزه

4- .setDescription('هذا ايمبيد جديد')
ده بيحدد الوصف الي هيتكتب جوا الايمبيد غيره براحتك
5- .setColor('#000000')
تحديد لون حواف الايمبيد وانا نزلت رموز الالوان الاساسية هنا

بنفسجي : #5865F2
اخضر : #57F287
اصفر : #FEE75C
وردي:#EB459E
احمر:#ED4245

FFFFFF#:
اسود:#000000
وفي غيرهم ياما على حسب الدرجه ممكن تلاقي اي لون انت عايزه بسهولة من جوجل او اي مصدر تاني

6-addField('حقل جديد', 'هذا حقل جديد')
السطر ده بيضيف حقل جديد بعنوان وقيمه او وصف ممكن تغيره بردوا

7-.setThumbnail('image.com')
هنا هتحط رابط الصورة المصغره بتاعتك الي هي صورة كدا

تتحط على الجنب وصغيرة عشان تحسن الشكل

8-.setImage('https://example.com/image.png')
وهنا هتحط رابط الصورة الاساسية الي انت عايزها والصورة دي هتظهر كبيرة وواضحه مش زي الThumbnail

9- .setFooter('Footer text', 'https://example.com/footer.png');
تحديد الfooter الي هيا الكتابة الي بتبقا تحت خالص في الامبيد وصغيرة خالص في السطر ده متقسمه لحجتين كتابة وصورة حددهم انت براحتك

10-message.channel.send({ embeds: [embed] });
وبعديها message.channel.send عشان يبعت رسالة في القناه و حددنا الايمبيد ب [embeds]

11- هنقفل الاقواس وكده خلصنا شرح ازاي تعمل ايمبيد الي بعده

---

variables | المتغيرات

المتغيرات بقا دي عشان تعمل قيمه حاجه او تحدد قيمة

حاجه مش معروفه ومتغيره وسميت كده عشان هي بتحدد قيمة متغيرة للحاجه من غير متغير في الحاجه نفسها القيمة بس الي بتتتغير وعشان نعرف المتغير في تلات كلمات اساسية وهما var,const,let وهشرح الفرق بينهم وامثلة عليهم اول حاجه هيا const

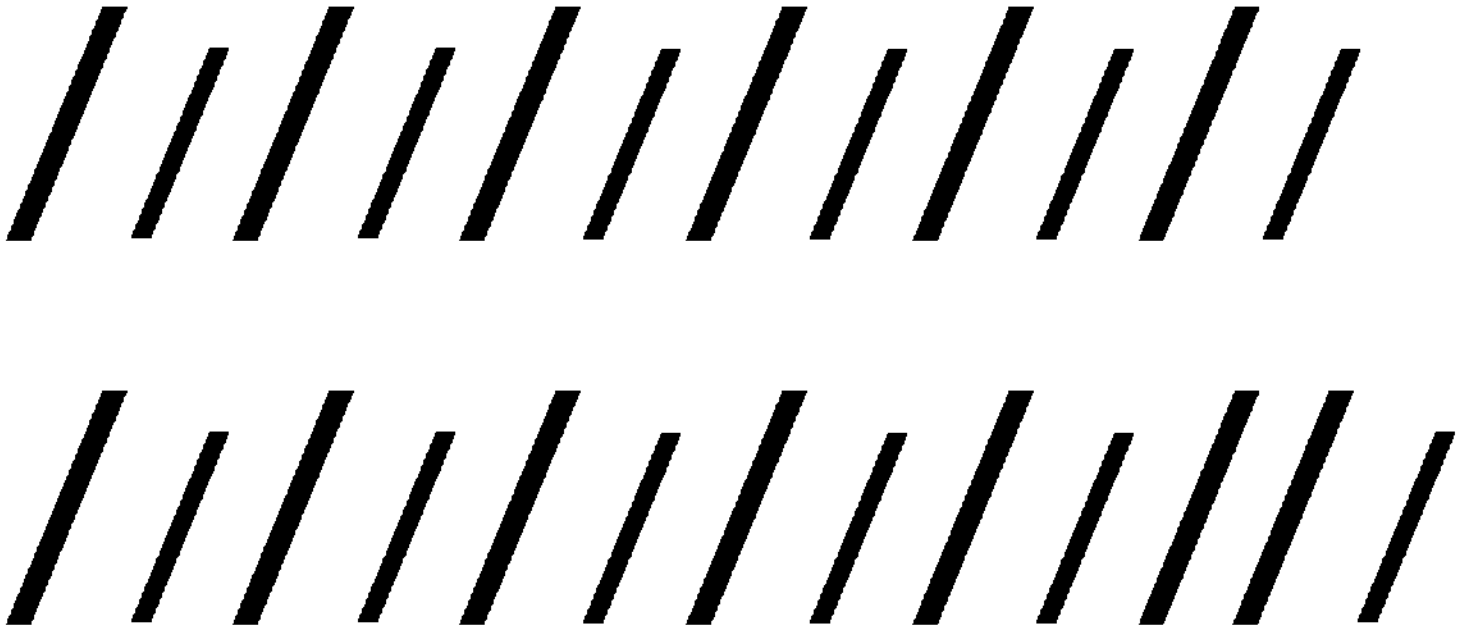
دي بتعرف متغير ثابت بمعنى حاجه هتكتبها جوا الكود ومش هتتغير خالص مثال ايام الاسبوع سبعه مش هتتغير شوف

const daysOfWeek = ["Sunday", الكود ده وهتفهم
"Monday", "Tuesday", "Wednesday", "Thursday",
;"Friday", "Saturday"]

client.on("messageCreate", message => {
  if (message.content === '!days') {
    message.reply(`دي ايام الاسبوع${daysOfWeek`});
  }
});
نلاحظ اننا لما نعوذ نخلي البوت يبعت رسالة فيها المتغير ده
بنحط قبله $ وهحطه بين القوسين دول {} نخش في ال let

دي بقا بتعرف متغير ممكن يتغير كل شويه مثلا فلوس بوت
بنك كريديت كده وبردوا تنفع تبقا ثابته مثلا لو بدلنا const
بletفي الكود الفوق بردوا هيشتغل let بتشغل متغير ثابت
ومش ثابت عادي

اما اخر حاجه هي var وvar نفس let فروق بسيطه مش لازم
نعرفها ولا هتفيدنا في حاجه وكذلك var تنفع للثابت والمتغير



علامات التحديد

هنشرح علامات التحديد زي ' ' و " " و ` ` والفرق مبينهم دلوقت اول حاجه ' ' والي اسمهم علامات تنصيص

دي بتستخدم عشان تحدد النص او تعرف سلسلة نصيه تشمل ارقام وحروف ورموز

تاني حاجه هي العلامة المزدوجة " "

هي نفس علمتين التنصيص بس لو عايز تضيف رموز جوا

النص استخدمها مثلا message.channel.send("اهلا "صديقي"") ده الفرق بينهم

تالت حاجه هي العلامة العكسية ` ` دي بتستخدم زي علامات التنصيص والعلامة المزدوجه بس ممكن تضيف فيها متغير يعني مينفعش تحط متغير في ' ' او في " " لازم تبقا هنا بس ` `
وبتعرف المتغير جواها بالطريقة دي ${variable}$
وكده نكون خلصنا علامات التحديد يلا الي بعده

<> <> <> <> <> <> <> <> <> <> <> <> <> <> <> <>

<> <> <> <> <> <> <> <> <> <> <> <> <> <> <> <>

<> <> <> <> <> <> <> <> <> <> <> <> <> <> <> <>

السلاش كوماند

هاندلر|HANDLER

في الجزء ده هنشرح ازاي تعمل امر سلاش كوماند هاندلر بطريقة سهله جدا

هنبدا اول حاجه بمثال بامر hello والبوت يرد ب hello

1-اول خطوة انك هتحمل discord.jsv13 ب

npm i discord.js@13

وتحملdiscordjs/builders@ ب

npm install @discordjs/builders

2-هنعمل مجلد جديد واسمه commands ونضيف فيه ملف باسم الامر الي احنا عايزينه ونحط بعده .js مثلا في المثال بتاعنا الامر اسمه hello فهنسمي الملف hello.js

3-بعد منعمل الملف هنضيف فيه الكود ده

```
const { SlashCommandBuilder } = require('@discordjs/builders');

module.exports = {
  data: new SlashCommandBuilder()
```

```
    .setName('hello')
    .setDescription('Say hello to the bot!'),

  async execute(interaction) {
    await interaction.reply(`Hello`);
  },
};
```

هنشرحه بعد منخلص الخطوات

4-هنروح للملف index.js الي هو الملف الرئيسي
ونحط فيه الكود ده

```
const { Client, Intents, Collection } =
require('discord.js');
const { token } = require('./config.json');
const { readdirSync } = require('fs');

const client = new Client({ intents:
[Intents.FLAGS.GUILDS] });
client.commands = new Collection();
```

```
const commandFiles = readdirSync('./
commands').filter(file =>
file.endsWith('.js'));

for (const file of commandFiles) {
  const command = require(`./commands/$
{file}`);

client.commands.set(command.data.name
, command);
}

client.on('ready', () => {
  console.log(`Logged in as $
{client.user.tag}!`);
});

client.on('interactionCreate', async
interaction => {
  if (!interaction.isCommand()) return;
```

```
  const command =
client.commands.get(interaction.command
Name);

  if (!command) return;

  try {
    await command.execute(interaction);
  } catch (error) {
    console.error(error);
    await interaction.reply({ content: 'An
error occurred while executing the
command!', ephemeral: true });
  }
});
```

هنشرحه بعد منخلص بردوا

6-بعد كده هنعمل ملف جديد باسم config.json لازم
json مش js ونحط فيه الكود ده

```
{
  "token": "توكن بوتك هنا"
}
```
ونحط التوكن بتاع البوت

7-اعمل run للبروجكيت وهتلاقيه اشتغل بدون اي مشاكل

انا اختصرتلكم كل ده وعملتلكم بروجيكت جاهز بكل ملاحظه كودات الهاندلر بتتغير دايما وفقا لتحديثات نسخه v13

دلوقتي هنشرح الكودات الي حطيناها

```
1-const { SlashCommandBuilder } = require('@discordjs/builders');

module.exports = {
  data: new SlashCommandBuilder()
    .setName('hello')
    .setDescription('Say hello to the bot!'),
```

```
  async execute(interaction) {
    await interaction.reply(`Hello`);
  },
};
```

1-const { SlashCommandBuilder } = require('@discordjs/builders');

ده بيعرف عملية بناء السلاش كوماند ب discordjs/builders

2-module.exports = {

ديه لازم نبدا بيها اي امر سلاش هيا الي بتستخرج الامر وبعدها هنعمل = عشان نعرف الامر ونفتح قوس نعرفه جوا

3-data: new SlashCommandBuilder()

دي بردوا ثابتة معانا وبنحطه عشان نعرف ان في امر سلاش جديد هيتبني وهنبدا بقا نعرف الامر

4- .setName('hello')
    .setDescription('Say hello to the bot!'),

تعريف الامر ذا نفسه setName بتعرف اسم الامر
وsetDescription بتعرف شرح الامر ولازم نحط
بعدها , عشان وظيفةasync تشتغل

5-async execute(interaction) {
مقدمة اننا هنسخدم التفاعل

6-await interaction.reply(`Hello`);
  },
};
نفس لما نعمل رد تلقائي بس بدل message
interaction عشان تفاعل وهتقلي طب وawait
موجودة ليه هقلك عشان في وظيفة async ودي لازم
يجي بعدها await ولو مش عايزين await لازم نمسح
من الكود وكده خلصنا شرح اول كود asyncوawait

2-const { Client, Intents, Collection } =
require('discord.js');
const { token } = require('./config.json');
const { readdirSync } = require('fs');

```
const client = new Client({ intents:
[Intents.FLAGS.GUILDS] });
client.commands = new Collection();

const commandFiles = readdirSync('./
commands').filter(file =>
file.endsWith('.js'));

for (const file of commandFiles) {
  const command = require(`./commands/$
{file}`);

client.commands.set(command.data.name
, command);
}

client.on('ready', () => {
  console.log(`Logged in as $
{client.user.tag}!`);
});
```

```
client.on('interactionCreate', async
interaction => {
  if (!interaction.isCommand()) return;

  const command =
client.commands.get(interaction.command
Name);

  if (!command) return;

  try {
    await command.execute(interaction);
   } catch (error) {
    console.error(error);
    await interaction.reply({ content: 'An
error occurred while executing the
command!', ephemeral: true });
  }
});

1-const { Client, Intents, Collection } =
```

```
require('discord.js');
```

تعريف ال client وال intents و ال collection

2-
```
const { token } = require('./config.json');
```

تعريف ان ال token موجود في ملف ال config.json

3-
```
const { readdirSync } = require('fs');
```

تعريف readdirSync الي هي بتقرا الملفات

4-
```
const client = new Client({ intents:
[Intents.FLAGS.GUILDS] });
client.commands = new Collection()
```

تعريف ال intents ومفيش message_guild عشان البوت مش محتاج يقرا الرسايل عشان ده امر سلاش

5-
```
const commandFiles = readdirSync('./
commands').filter(file =>
file.endsWith('.js'));
```

تعريف ان الاوامر موجوده داخل مجلد commands

وبتنتهي ب js زي مثلا hello.js

6-for (const file of commandFiles) {
  const command = require(`./commands/${file}`);

بيكمل السطر الي فات في تعريف الاوامر

7-
client.commands.set(command.data.name, command);
}

ده بيعرف ان الاوامر هتتعمل او هتتحدد جوا الكود الي فيه data في ملف الcommands

8-client.on('ready', () => {
  console.log(`Logged in as ${client.user.tag}!`);
});

لو البروجكت جاهز يطبع في الكونسل ان loggen in) (اسم البوت as

9-client.on('interactionCreate', async

```
interaction => {
  if (!interaction.isCommand()) return;,
```

كانك بتعمل رد تلقائي بس هنا بتعمل تفاعل عشان كدا بدلت message بinteraction

10-`const command = client.commands.get(interaction.commandName);`

تعريف الcommand الي هو isCommand فكده بنعمل تفاعل مع الاوامر الي جوا الملفات

11- `if (!command) return;`

لو تم طلب الامر return يعني كذا يحصل لو الامر اطلب

12-
```
try {
   await command.execute(interaction);
```

اولا try تدل اننا بنجرب او بنخلي البوت يتحقق لو في اخطاء.ثانيا command.excute ده معناه اننا هنخلي البوت يحاول باستخدام دالة interaction وawait ممكن نشيلها لو شيلنا async الي قبلها

13- } catch (error) {
    console.error(error);
    await interaction.reply({ content: 'An error occurred while executing the command!', ephemeral: true });

الحته دي فايدتها انه لو في غلط تركيبة السلاش يطبع في الكونسل error ويكتب لما تطلب الامر من البوت يكتبلك An error occurred while executing the command!', وephemeral يعني غير مرئية

كده خلصنا شرح السلاش كوماند هاندلر يلا نخش في الي بعده

</> </> </> </> </>

</> </> </> </> </>

</> </> </> </> </>

Buttons | الازرار

في الجزء ده هنشرح ازاي تعمل رسالة تحتها زر او اكتر بسهولة وكالعادة هنبدا بمثال

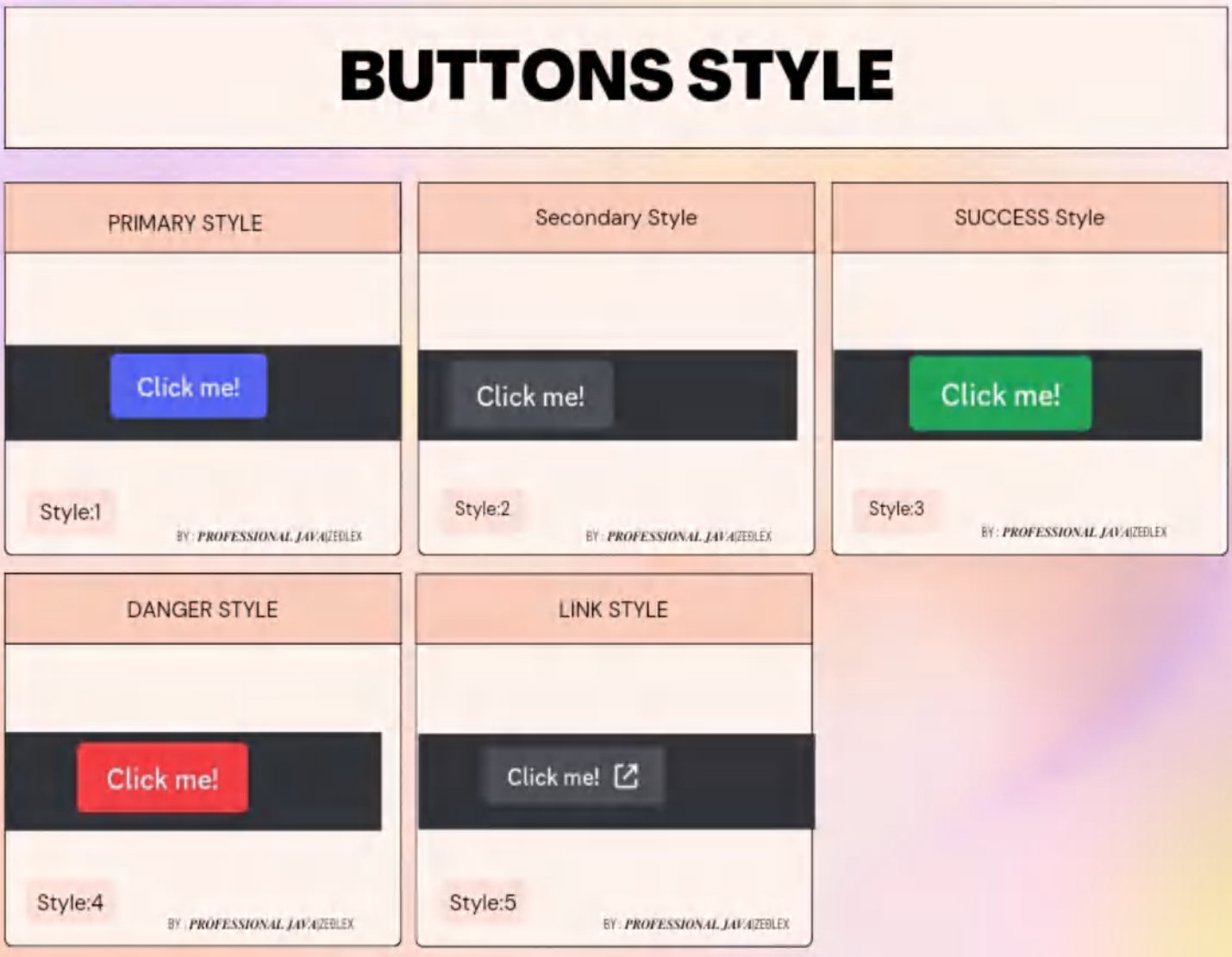
مثال 1-

```
const { MessageActionRow,
MessageButton } = require('discord.js');

client.on('messageCreate', (message) => {
  if (message.content === 'test-button') {
```

```
    const button = new MessageButton()
      .setCustomId('example-button')
      .setLabel('Click me!')
      .setStyle('PRIMARY');

    const row = new
MessageActionRow().addComponents(but
ton);

    const messageContent = 'Hello world!';

    message.channel.send({ content:
messageContent, components: [row] });
  }
});
```

-2 مثال

```
client.on('interactionCreate', async
(interaction) => {
  if (!interaction.isButton()) return;
  if (interaction.customId === 'example-
button') {
```




```
    await interaction.reply('You clicked the button!'); // رد على التفاعل
  }
});
```

جمع الكودين

```
const { MessageActionRow, MessageButton } = require('discord.js');
```

```
client.on('messageCreate', (message) => {
  if (message.content === 'test-button') {
    const button = new MessageButton()
      .setCustomId('example-button')
      .setLabel('Click me!')
      .setStyle('PRIMARY');

    const row = new
MessageActionRow().addComponents(but
ton);

    const messageContent = 'Hello world!';

    message.channel.send({ content:
messageContent, components: [row] });
  }
});
client.on('interactionCreate', async
(interaction) => {
  if (!interaction.isButton()) return;
  if (interaction.customId === 'example-
```

```
button') {
    await interaction.reply('You clicked the
button!');
  }
});
```

الشرح

1-const { MessageActionRow,
MessageButton } = require('discord.js');
تعريف وضع الازرار والتفاعل معاها

2-client.on('messageCreate', (message) =>
{
  if (message.content === 'test-button') {

طبعا كل ده شرحناه في الرد التلقائي

3-const button = new MessageButton()

```
    .setCustomId('example-button')
    .setLabel('Click me!')
    .setStyle('PRIMARY');
```

ده تعريف البوتون اول حاجة const عشان نعمل متغير ثابت وتاني حاجة button الي هو المتغير وبعد كده = عشان لازم نستخدمها لما نستخدم const وبعدها new MessageBtton() كل ده بنبدا في تعريف الزر بقا ذات نفسه وبعدها setCustomId دي بقا بتحدد اسم الزرار الي هتعمل تفاعل معاه وبعدين setLabel بنحدد فيها اسم الزر الي هيبان تحت الرسالة وبعدها set Style الي هو بقولك تختار style الزراز او نوع وانواع الbuttons وشكلها شرحته وهتلاقوا صورة الشرح في الصفحه الجاية

4-const row = new MessageActionRow().addComponents(button);

const row متغير ثابت جديد
= new MessageActionRow()

كده احنا بنعمل صف جديد عشان نعرف المتغير
addComponents(button);
معناه اننا بنضيف عنصر او الرسالة مع ال button

5-const messageContent = 'Hello world!';
هنا بنحط محتوى الرسالة

messageContent, components: [row] });
  }
});
هنا بنرسل كل الي احنا عملناه

7-client.on('interactionCreate', async
(interaction) => {
  if (!interaction.isButton()) return;
هنا بنعمل تفاعل جديد ونعرف ان التفاعل ده بتاع
buttonال

8- if (interaction.customId === 'example-
button') {

```
    await interaction.reply('You clicked the
button!');
  }
});
```

اول سطر بنحدد اي button الي هنعمل عليه التفاعل بالcustom id بتاعو وبعدها بقا بنحط الرد على التفاعل.وكده نكون خلصنا ال button يلا الي بعده

SELECT MENU|سيليكت منيو

في الجزء ده هنشرح ازاي تعمل رسالة فيها سيليكت منيو وكالعادة هنبدا بمثال

```
client.on('messageCreate', message => {
  if (message.author.bot || !
message.content.startsWith('!hello'))
return;

const { MessageActionRow,
```

```
MessageSelectMenu } =
require('discord.js');


const selectMenu = new
MessageSelectMenu()
    .setCustomId('menu-id')
    .setPlaceholder('اختر خيارًا')
    .addOptions([
        {
            label: 'خيار 1',
            description: 'وصف الخيار 1',
            value: 'option1'
        },
        {
            label: 'خيار 2',
            description: 'وصف الخيار 2',
            value: 'option2'
        },
        {
            label: 'خيار 3',
```

```
            description: 'وصف الخيار 3',
            value: 'option3'
        },
        {
            label: 'خيار 4',
            description: 'وصف الخيار 4',
            value: 'option4'
        }
    ]);


const actionRow = new
MessageActionRow()
    .addComponents(selectMenu);

message.channel.send({ content: 'اختر
خيارًا:', components: [actionRow] });


client.on('interactionCreate', interaction => {
    if (!interaction.isSelectMenu()) return;
```

```
    if (interaction.customId !== 'menu-id')
return;

    const selectedOption =
interaction.values[0];

    let replyMessage = '';
    switch (selectedOption) {
        case 'option1':
            replyMessage = 'لقد اخترت الخيار 1';
            break;
        case 'option2':
            replyMessage = 'لقد اخترت الخيار 2';
            break;
        case 'option3':
            replyMessage = 'لقد اخترت الخيار 3';
            break;
        case 'option4':
            replyMessage = 'لقد اخترت الخيار 4';
            break;
        default:
```

```
        replyMessage = ' لقد اخترت خيارًا غير
معروف';
        break;
    }

    interaction.reply(replyMessage);
});
```

يلا نشرحه

```
client.on('messageCreate', message => {-1
  if (message.author.bot || !
message.content.startsWith('!hello'))
return;
```

شرحناه قبل كده بس هنا ضفنا شرط
(message.author.bot) وده لو الي ماتب الامر مش
بوت بيفيد الشرط ده لو مثلا عايز تعمل خط تلقائي او
كدا وبعد كده بدل ما نحط كمان if عملنا || عشان
نضيف شرطين بسهولة

2-

```
const { MessageActionRow,
MessageSelectMenu } =
```

```
require('discord.js');
```
تعريف السليكت منيو والاكشن رو

3-
```
const selectMenu = new MessageSelectMenu()
```

هنا بقا هنبدا نضيف السليكت منيو جديد

4-
```
.setCustomId('menu-id')
```
هنا هنحدد ايدي المنيو الي هنتفاعل معاه هنحط اي حاجه المهم نحطها في التفاعل تاني

5-
```
.setPlaceholder('اختر خيارًا')
```
هنا بنحدد وصف الرسالة قبل متختار اي اختيار

6-
```
.addOptions([
    {
        label: 'خيار 1',
        description: 'وصف الخيار 1',
        value: 'option1'
    },
    {
```

```
            label: 'خيار 2',
            description: 'وصف الخيار 2',
            value: 'option2'
        },
        {
            label: 'خيار 3',
            description: 'وصف الخيار 3',
            value: 'option3'
        },
        {
            label: 'خيار 4',
            description: 'وصف الخيار 4',
            value: 'option4'
        }
    ]);
```

هنبدا ب addoptions ونفتح قوسين ونضيف كل الاختيارات الي احنا عايزينها ونحط وصفها وتحتها القيمة الي هنعمل معاها تفاعل زي الايدي كده

7-const actionRow = new MessageActionRow()

.addComponents(selectMenu);

هنبدا بقا نعرف الComponets الي هنحطها في الرسالة يعني من الاخر نضيف السيليكت منيو في الرسالة من خلال الaction row

8-message.channel.send({ content: 'اختر خيارًا:', components: [actionRow] });

هنخلي بقا البوت يرسل الرسالة بمحتواها بالسليكت منيو

9-client.on('interactionCreate', interaction => {
if (!interaction.isSelectMenu()) return;
if (interaction.customId !== 'menu-id') return;

هنعرف التفاعل وهنحط الcustom-id بتاع المنيو الي قولناه

10-const selectedOption =

interaction.values[0];
هنا بنعرف الخيارات الي هيحددها العضو عشان نعمل تفاعل معاها

11-let replyMessage = '';
هنعرف الرد على الرسايل وهنسيبها فضيه عشان البوت يفهم ان احنا هنعرفها تحت

12- switch (selectedOption) {
ده بقا بيقول لو حد اختار حاجه من الخيارات كذا يحصل وكذا الي هي السطور الجايه عشان كده فتحنا قوس

13-case 'option1':
        replyMessage = 'لقد اخترت الخيار 1';
        break;
    case 'option2':
        replyMessage = 'لقد اخترت الخيار 2';
        break;
    case 'option3':
        replyMessage = 'لقد اخترت الخيار 3';

```
        break;
    case 'option4':
        replyMessage = 'لقد اخترت الخيار 4';
```

هنحط التفاعلات بتاعت كل خيار case دي معنها في حالة 'option 1 مثلا الي هي القيمة الي حددنها في الvalue بتاعت كل خيار يعني السطر الاول معناته في حاله اختيار العضو 'الخيار 1' ونحط : عشان نبين اننا هنقول على الحدث الي هيحصل زي return كده وبعد كل واحده بقا replyMessage يعني البوت يرد بكذا ونفس الكلام في باقي الخيارات

14-

```
        break;
    default:
        replyMessage = 'لقد اخترت خيارًا غير معروف';
        break;
}
```

وهنا هنضيف السطر ده عشان لو مثلا انت نسيت تعرف اي خيار واخترته البوت هيقولك انك معرفتش الخيار ده ومش هيطبع حاجه في الكونسل ولا البوت هيقف

15-

```
interaction.reply(replyMessage);
```

وطبعا مش هينفع نعمل تفاعل غير ونضيف له الرد بتاعو الي هو replyMessage الي لسا معرفينها ونقفل الاقواس });

وبكده نكون خلصنا ازاي تعمل سليكت منيو جديد بالتفصيل يلا نشوف الي بعده

</> </> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </> </>

عشر وظايف جديدة

في الجزء ده هنعرف عشر وظايف جديده وفوائدهم واستخدامتهم وامثلة عليهم يلا نشوفهم

1-message.edit()

طبعا من اسمها هنعرف انها بتستخدم عشان نعدل الرسالة ودي بقا فويدها لو عايز تعدل رسالة بعد وقت معين وبتستخدم في حاجات كتير ومثال عليها

```
setTimeout(() => {
```

```
  message.edit('تم تعديل الرسالة!');
}, 5000);
```

طبعا طالاما قولنا بعد وقت معين ده هيدخلنا على وظيفه جديده ومهمه جدا الي هي setTimeout هنشرحها بس خلينا في المثال هنلاقي في المثال ده ان البوت هيعدل على الرسالة بعد 5 ثواني دي بس وظيفه message.edit يلا نخش فالوظيفه التانيه

2-setTimeout(()

الوظيفه دي مفيدة جدا وبتستخدم في حاجات كتير جدا والمثال عليها كان في message.edit مثلا وغالبا هتيجي بنفس الشكل الي هو اول سطر setTimeout(() => { وفي الاخر هنحط الوقت بالملي ثانيه وكده خلصنا وظيفه الsetTimeout يلا الي بعده

3-guildMemberAdd

الوظيفه دي طبعا من اهم واكتر الوظايف معرفه ومن اسمها هنعرف انها بتستخدم عشان نبعت رسالة لما عضو يدخل فوائدها نظام الويلكوم + رتب التلقائي للاعضاء الجدد مثال عليها

```
client.on('guildMemberAdd', member => {
```

```
  const channel =
member.guild.channels.cache.find(ch =>
ch.name === 'welcome');
  if (!channel) return;
  channel.send(`Welcome to the server, $
{member}`);
});
```

طبعا هنبدا باول سطر ونحط الوظيفه جوا client.on
عشان نحدد الحدث

} <= client.on('guildMemberAdd', member

وهنحدد القناه بوظيفه .guild.channels.cache.
وهنحدد اسم القناه بالظبط وحط تحت بالظبط مليون
خط وبعد كده هنعمل if (!channel) return; عشان
حددنا القناه بوظيفه .guild.channels.cache.
ونحط بقا channel.send مش هنحط message
عشان اصلا احنا مش هنحتاجها ولا حددناها احنا
حددنا القناه بس وبس كده

4-voiceStateUpdate

وظيفه دايما هنلقيها في معظم بوتات السيستم ديه
بتقولك اي حاجه متعلقه بالروم الصوتي مثلا لوج لما

حد يطلع من روم او يدخل في روم وحاجات تانيه كتير
مثال عليها

```
client.on('voiceStateUpdate', (oldState,
newState) => {
  const channel =
client.channels.cache.get('1234567890123
45678');
  if (!channel) return;
  const member = newState.member;
  if (newState.channelId &&
newState.channelId !==
oldState.channelId) {
    channel.send(`${member.displayName}
joined the voice channel $
{newState.channel.name}`);
  } else if (!newState.channelId) {
    channel.send(`${member.displayName}
left the voice channel $
{oldState.channel.name}`);
  }
```

});
في الكود دا عملت لوج لما عضو يدخل او يخرج هنحط
ايدي القناه بعد وظيفه channels.cache وبعدها بقا
هنحدد احنا عايزين ايه من البوتnewState دي لما
عضو يطلع اماoldstate لما عضو يطلع وبعد منحدد
العضو هنحط بقا الي هو لما وظيفه الoldState
تحصل البوت يبعت رسالة والعكس ومش هننسا
نستخدم else if عشان هنضيف اكتر من شرط وكدا
الكود خلص

5-client.on('ready', () => {
وظيفة سهله ومهمه استخدمها سطرين بس وهي
ببساطا بتستخدم عشان لو البوت شغال بدون مشكال
يطبعلك حاجه معينه في ال console وده المثال
```
client.on('ready', () => {
   console.log(`Logged in as ${client.user.tag}!`);
});
```
هنحط console.log عشان يطبع في الكونسل
client.user.tag ب as يعني تم الدخول logged in
الي هو اسم البوت طبعا ممكن نغير الكلام على حسب

الرغبة

6-setActivity

الوظيفه دي فايدتها ان ممكن من خلالها نغير حاله البوت مثلا playing وهكذا وببساطة ده مثال عن طريقة استخدمها

```
client.on('ready', () => {
   client.user.setActivity('ONLINE!', { type: 'PLAYING' });
});
```

هنستخدم client.on(ready

عشان زي مقلنا لو البوت شغال بدون مشاكل وممكن نعدل علكود نخليه بامر بدلclient.on(ready واظن اكيد عرفنا ازاي نعمل رسالة بامر وهنضيف args ودي هنشرحها بعدين وكده نكون خلصنا 6 وظايف

7-setUsername

زي الي قبليها بس دي بتغير اسم البوت وهي بتبقا في سطر واحد وانت بقا تحدد كودادتك براحتك والسطر ده هو

```
client.user.setUsername('New Bot Name');
```

وبس كده

8-setAvatar

اظن طبعا عرفناه زي الي قبله بيغير الافاتار كدا احنا عرفنا ازاي نغير اسم وصورة وحاله البوت وكذلك بيبقا نفس استعمال الSetusername كده

```
client.user.setAvatar('رابط الصورة');
```

9-createDM();

وظيفه في غاية الاهمية والسهولة بنستخدمها عشان نبعت رسالة في الخاص واستخدام العملية دي كالتالي

```
client.on('messageCreate', async message
=> {
  if (message.content === 'تيست!') {
    const member = message.member;
    const dmChannel = await
member.createDM();
    await dmChannel.send('تيست!');
  }
});
```

اول حاجه هنبدا ب client.on ونحط الكلام ده بعدها هنحط الامر الي عازينه ونحدد العضو انه هو الي طالب الامر ونحطط قناه الdm ونيجي بقا نستخدم وظيفه

وبعدها هنحدد الرسالة وبس كده createDm

10-message.delete();

تستخدم هذه الوظيفه عشان نسمح بيها الامر الي انبعت من العضو يعني مثلا هنكتب الامر وفي اخره هنكتب ;message.delete() وبكده نكون خلصنا ال10 وظايف نخش في الي بعده

كلام عشوائي | خيارات عشوائية

في الجزء ده هنشرح ازاي تخلي البوت يولدلك كلام عشوائي وازاي تخليه يختار خيار عشوائي الفرق بينهم ان في الكلام العشوائي انت بس هتحط للبوت الحروف وتختار طول الكلمة وهو هيولدلك كلمه عشوائيه اما الخيارات العشوائية هتحط للبوت كذا خيار وهو هيطلع واحد عشوائي زي عجله الحظ اول حاجه يلا نشوف الخيارات العشوائية وكالعادة

مثال:-

```
const options = [
 'خيار 1',
 'خيار 2',
'خيار 3'
];
client.on('messageCreate', message => {
  if (message.content === 'اختر!') {
    const randomIndex =
Math.floor(Math.random() *
options.length);
    const randomOption =
options[randomIndex];
    message.reply(randomOption);
  }
});
```

يلا نشرح

1-const options = [

'خيار 1',

'خيار 2',

'خيار 3'

];
بنحدد الخيارات الي عازين البوت يختار منها عشوائي

2-client.on('messageCreate', message => {
if (message.content === 'اخترا!') {
تحديد الامر

3-const randomIndex =
Math.floor(Math.random() *
options.length);
بنبدا نستخدم وظيفة math.random وبنحدد ان
هنعمل اختيارات ونامره انه يختار حاجه عشوائيه

4-const randomOption =
options[randomIndex];
بنحدد ان الrandomoption او ممكن نغير اسمه هو
ال randomIndex من الoptions الي هو الاختيار
العشوائي من الاختيارات وبنحدده عشان نرد بيه

5-message.reply(randomOption);
هنرد ب الاختيار العشوائي وبس كده خلصنا

الاختيارات العشوائية يلا نخش على الكلام العشوائي
مثال:-

```
const symbols =
'ABCDEFGHIJKLMNOPQRSTUVWXYZabcd
efghijklmnopqrstuvwxyz0123456789!@#$
%^&*()_+={}[]\\|;:\'",.<>?/~`';

function generateRandomString(length) {
  let result = '';
  for (let i = 0; i < length; i++) {
    result +=
symbols.charAt(Math.floor(Math.random()
* symbols.length));
  }
  return result;
}

client.on('messageCreate', async
(message) => {
  if (message.content === '!random') {
    const length = 10;
```

```
    const randomString =
generateRandomString(length);
    await message.reply(`\`${randomString}
\``);
  }
});
```

كالعادة يلا نشرح

1-const symbols =
'ABCDEFGHIJKLMNOPQRSTUVWXYZabcd
efghijklmnopqrstuvwxyz0123456789!@#$
%^&*()_+={}[]\\|;:\'",.<>?/~`';

هنحدد الحروف او الرموز او الارقام الي عايزين البوت يولد منها كلام عشوائي انا حطيت كل حاجه عايز تشيل او تزود براحتك

2-function generateRandomString(length) {

دي الوديفة الشائعة في جافا سكريبت عشان تولد كلام عشوائي والي هي generateRandomString اما function يعني وظيفة وlength الي هو طول الكلمة عايز البوت يطلعلك كام رمز وحرف ورقم عشوائي

3-let result = '';

هنحدد النتيجة وهنسبها فاضية عشان هي هتحدد

generateRandomString باستخدام وظيفه

وطالما سبنا حاجه فاضية وهنعرفها بوظيفة تانيه فغالبا

for هنستخدم

4-for (let i = 0; i < length; i++) {

result +=

symbols.charAt(Math.floor(Math.random()

* symbols.length));

}

```
TypeError: Cannot read properties of undefined (reading 'FLAGS')
    at Object.<anonymous> (/home/runner/Nodejs-29/index.js:2:47)
    at Module._compile (node:internal/modules/cjs/loader:1159:14)
Hint: hit control+c anytime to enter REPL.
```

return result;

}

ده هو الجزء الاساسي من وظيفة

generateRandomString

هو ده الي بنسخدمه عشان نولد الكلمة بطول محدد

وبحروف عشوائية في كل حرف هنسنخدم

math.random عشان نعمل حرف واحد او عشان

تدينا خيار عشوائي واحد وعلى حسب الطول الوظيفة

هتفضل تعيد نفسها وهنجعها مع بعضها لحد ميبقوا

```
Hint: hit control+c anytime to enter REPL.
/home/runner/Nodejs-29/node_modules/discord.js/src/client/websock
et/WebSocketManager.js:129
    const invalidToken = new Error(WSCodes[4004]);
                         ^

Error [TOKEN_INVALID]: An invalid token was provided.
    at WebSocketManager.connect (/home/runner/Nodejs-29/node_modu
les/discord.js/src/client/websocket/WebSocketManager.js:129:26)
    at Client.login (/home/runner/Nodejs-29/node_modules/discord.
js/src/client/Client.js:254:21)
    at Object.<anonymous> (/home/runner/Nodejs-29/index.js:10:8)
    at Module._compile (node:internal/modules/cjs/loader:1159:14)
 {
  [Symbol(code)]: 'TOKEN_INVALID'
}
```

على قد الطول وكل حاجه تمام وتعريف ال result كده بقا جاهز

5-client.on('messageCreate', async (message) => {

```
SyntaxError: missing ) after argument list
Hint: hit control+c anytime to enter REPL.
>
```

if (message.content === '!random') {

بناء الأمر

6-const length = 10;

```
/home/runner/Bestsystemlessevergreater-1/index.
333
client.login(process.env.token);

SyntaxError: Unexpected end of input
Hint: hit control+c anytime to enter REPL.
```

بنحدد طول الكلمة طبعا على حسب الرغبة نزود او نقلل

7-const randomString = generateRandomString(length);
    await message.reply(`\`${randomString}\``);

```
SyntaxError: Unexpected token ')'
Hint: hit control+c anytime to enter REPL.
> 
```

  }
});

بنحدد ان ال randomString هو ال generatedRandomString الي قعدين بنعرف فيه من بدري وبعد كده الرد على الرسالة بالنص التلقائي وبس كده

```
SyntaxError: await is only valid in async functions
 and the top level bodies of modules
Hint: hit control+c anytime to enter REPL.
```

</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>

</> </> </> </> </> </> </> </> </> </> </>

```
ReferenceError: message is not defined
    at Object.<anonymous> (/home/runner/Bestsystemlessevergreater-1/index.js:1327:3)
    at Module._compile (node:internal/modules/cjs/loader:1255:14)
```

</> </> </> </> </> </> </> </> </> </> </>

</> </> </> </> </> </> </> </> </> </> </>

</> </> </> </> </> </> </> </> </> </> </>

```
TypeError:  . . .  is not a function
    at Object.<anonymous> (/home/runner/Bestsystemlessevergreater-1/index.js:1326:8)
    at Module._compile (node:internal/modules/cjs/loader:1255:14)
```

errors | اخطاء

في الجزء ده هنجيب 10 ايرورات مختلفة وازاي نتعامل معاهم ونحلهم بكل سهولة يلا نبدا

1-

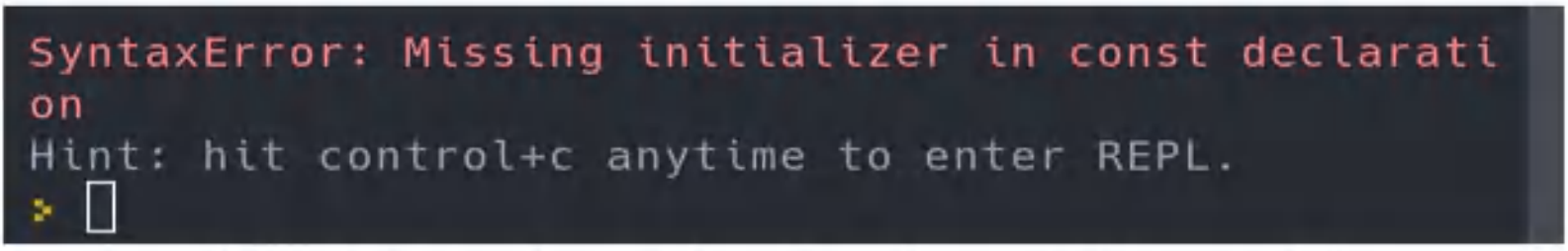
حل المشكله دي ببساطة انك هتروح الshell وتكتب
npm install discord.js@13

2-
حله سهل معناه ان توكن بوتك غلط لازم تتاكد منه انه صح ولو مش عارف تجيبه من
discord.developor.com

3-
ايرور سهل معناه انك ناسي تحط قوس (فبالتالي الحل هو انك هتدور على المكان الصح وتحط فيه قوس زيادة

4-

الايرور ده معناه ان اخر سطر في الكود الاقواس الي وراه ناقصه وحله انك هتروح قبل اخر سطر وتحط ;({



انا عندي اخر سطر client.login هوا في الايرور هيبينلك اخر سطر عندك

5-

الايرور ده بيظهرلك لما تكون حاطط قوس (بطريقة غلط يعني ياما القوس متزود على الفاضي مش مفيد ياما محطوط غلط والحل انك هتجرب تشيله لو منفعش معاك هتحط قبله { يعني هو كان عايز القوس ده قبله عشان يشتغل فلازم تجرب الاتنين عشان تحله

6-

الايرور ده بيقول ان وظيفة await محتاجه async والحل هو حاجه من اتنين يا انك تمسح كلمة await خالص ياما انك تحط كلمة async والاسهل انك تمسح await عشان ممكن تحط وظيفة async في مكان غلط

7-

الايرور ده بيقول ان ال message مش متحدده يعني
انت محددتهاش والايرور ده هيجي وانت بتعمل الامر
وناسي تعرف الرسالة والحل انك هتحط السطر ده
client.on('messageCreate', message => {
وبكده هتكون عرفت الmessage الي هي الرسالة
8-
الايرور ده بيقولك ان كذا مش وظيفة وانا مسحت الي
قبل is عشان تعرفوا ان الايرور مش محدد على حاجه
واحده فأي كلمه يجي بعدها الايرور ده اعرف ان حله
انك ياما كاتب اسم الوظيفة غلط مثلا بدل client.on
تكتب منغير قصد client.ok او اي حاجه غير كده
فلازم تتاكد من اسم الوظيفة وياما الوظيفة مش
متاحه على نسخه discord.js الي عندك مثلا تكون
مش متاحه غير علىv14 وانت بستخدم v13
9-
من اكتر الايرورات الي بتقابل الناس وبردوا مسحت
الي بعد token عشان مش مقتصر على حاجه واحده
الايرور ده بيقولك ان الكلمه الي انت كاتبها غلط يعني
محطوطة بشكل غلط وحله بيبقا على حسب الكلمه
بس في كذا حل بيبقا على الكل يعني انت جرب اول
حاجه لو انت بتحدد حاجه او بتحدد وظيفة حط قبليها

قوس { وابقا اقفله وسعات بيبقا حلها انك تحط نقطة .
في اخر السطر الي قبل الكلمه الي معمول عليها الايرور
وسعات بيبقا لازم تحط ; وسعات , على حسب
الاستخدام وسعات بيبقا معناها انك حاطط الكلمه في
مكان غلط وبردوا ممكن ان الكلمه تكون غلط وملهاش
لازمه ولازم تنحذف وفي حاجات تانيه كتير فالافضل
عشان تحل الوظيفة دي انك تطلب السبورت في
سيرفرنا للمساعده والمختصين في البرمجه هيسعدوك

رابط السيرفر

---

https://discord.gg/Vf9Nrr5att

10-
الايرور ده بيقلك انك ناسي علامة تعريف المتغير
وعلامة التعريف في javascript هي = فلازم تحط
قبل متعرف المتغير يعني لو انت كاتب const hello
اكتب hi const hello = hi وبكده نكون خلصنا
</> </> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </> </>

</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>
</> </> </> </> </> </> </> </> </> </> </>

كود مهم مع الشرح

في الجزء ده هنجيب كود مهم جدا ونشرحه كلمة

كلمة يلا نبدأ

1-

```
client.on('messageCreate', async message
=> {
  if (message.content === 'create-room') {
    const channel = await
message.guild.channels.create('New
Room', {
      type: 'GUILD_VOICE',
      parent: message.channel.parentId,
      permissionOverwrites: [
        {
          id: message.guild.roles.everyone.id,
          deny: ['VIEW_CHANNEL']
        },
        {
          id: message.author.id,
          allow: ['VIEW_CHANNEL']
        }
```

```
    ]
   });

    await message.reply(`Created new voice
channel: ${channel}`);
  }
});
```

ده كود بيعمل قناه صوتيه جديده في السيرفر يلا نشرحه

1
```
client.on('messageCreate', async
message => {
  if (message.content === 'create-room') {
```

صنع الامر

2-
```
const channel = await
message.guild.channels.create('New Room
```

بنحدد القناه وبنخلي البوت يعمل قناه جديده باستخدام وظيفه guild.channels.create وبنحدد اسم القناه انا هنا عملته New room

3-, {
type: 'GUILD_VOICE',
بنحدد نوع القناه انا حددته هنا voice الي هي صوتية
لو عايز تخليه كتابية هتبدل VOICE ب TEXT

4-parent: message.channel.parentId,
وفي id الي هي الكاتوجري وبنحدد الكاتوجري parent
السطر ده متحدد انه ايدي الكاتوجري هو ايدي
الكاتوجري بتاع القناه الي انبعت فيها الرسالة

5-permissionOverwrites: [
بنبدا نستخدم وظيفه permissionOverwrites
عشان نعدل صلاحيات القناه

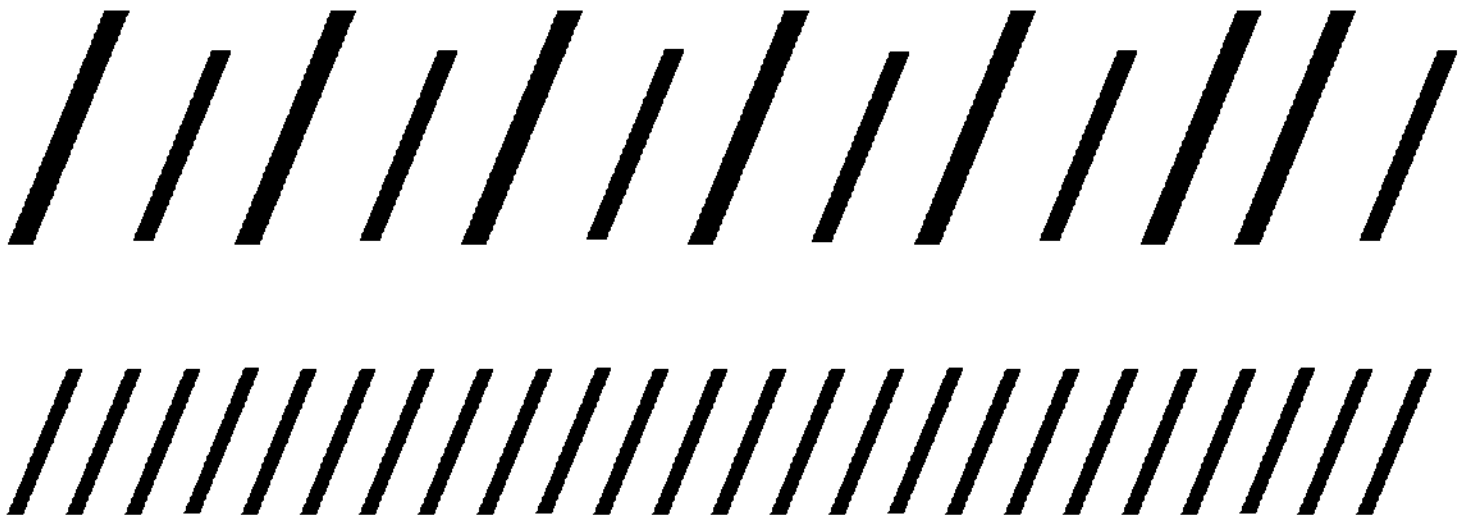
{
6- id: message.guild.roles.everyone.id
بنقفل القوس الي هو كان بتاع تعريف اسم ونوع
والكاتوجري بتاع القناه عشان نبدا نغير في
الpermissions وهنبدا نحدد الرتبه الي احنا عازينها
انا هنا محدد ايدي رتبه ال everyone من خلال

message.guild.roles.everyone.id وظيفه

7-deny: ['VIEW_CHANNEL']
},
بعد محددنا الرتبه لازم نحدد احنا عايزين نعمل فيها ايه
انا هنا كنت عايز امنع عنها انها تشوف القناه
فاستخدمت deny وحطيت صلاحية view_channel
اما لو عايز تعمل العكس فعكس deny هو allow كده
خلصنا التعديل الاول في الصلاحيات

8-id: message.author.id,
allow: ['VIEW_CHANNEL']
}
]
});
بعد منقفل قوس التعديل الاول ونفتح قوس للتعديل
الجديد بنحدد الايدي الجديد او ممكن نحدد
everyone تاني بس هنا انا حددت صاحب الامر
وخليته هو بس الي يقدر يشوف القناه واستخدم
وظيفه allow وحددت الايدي بتاعه من خلال وظيفه
message.author.id وهنقفل الاقواس كلها وبكده

نكون خلصنا كده الكود



كودات|CODES

وصلنا الى نهاية الكتاب وهنا سنعرض عليكم 35 كود بدون شرح ومن المفترض ان تكون هذه الجزئية مفيدة لبوتاتكم واختبار لكم اذا اسطتعتم شرحهم بدون مسعادة ويلا نبدا

1-كود انشاء رتبة جديده

```
client.on('messageCreate', async message
=> {
  if (message.content === '!newrole') {
    const guild = message.guild;
    const role = await guild.roles.create({
      name: 'new-role',
      color: 'BLUE',
      permissions: ['KICK_MEMBERS',
'BAN_MEMBERS']
    });
    message.reply(`Role created: ${role}`);
  }
});
```

2-كود اعطاء رتبه لعضو

```
client.on('messageCreate', async message
=> {
  if (message.content === '!addrole') {
    const member = message.member;
```

```
    const role =
message.guild.roles.cache.find(role =>
role.name === 'new-role');
    if (role) {
      await member.roles.add(role);
      message.reply(`Role added: ${role}`);
    } else {
      message.reply('Role not found');
    }
  }
});
```

3-كود بيبعت رسالة بعد فترة معينه

```
const { Client, Intents } =
require('discord.js');
const client = new Client({ intents:
[Intents.FLAGS.GUILDS,
Intents.FLAGS.GUILD_MESSAGES] });

client.on('messageCreate', async message
=> {
```

```
  if (message.content === '!
delayedmessage') {
    const channel = message.channel;
    const delay = 5000; // 5 seconds
    const content = 'This is a delayed
message!';
    setTimeout(() => {
      channel.send(content);
    }, delay);
    message.reply(`Message will be sent in $
{delay / 1000} seconds`);
  }
});
```

4-كود بيبعت رسالة لو العضو اتكى على ايموجي

```
client.on('messageCreate', async message
=> {
  if (message.content === '!reactionsend') {
    const channel = message.channel;
    const content = 'This is a reaction-based
```

```
message!';
    const filter = (reaction, user) =>
reaction.emoji.name === '👍' && user.id
=== message.author.id;
    const collector =
channel.createMessageComponentCollect
or({ filter, time: 15000 });
    collector.on('collect', () => {
      channel.send(content);
      collector.stop();
    });
    message.reply('React with 👍 to send
the message');
  }
});
```

5-كود بيبعت رسالة ويحدد الوقت

```
client.on('messageCreate', message => {
  if (message.content === '!time') {
    const now = new Date();
```

```
    const hours = now.getHours();
    const minutes = now.getMinutes();
    const seconds = now.getSeconds();
    message.channel.send(`The time is ${hours}:${minutes}:${seconds}`);
  }
});
```

6-كود يوقف تشغيل البوت

```
client.on('messageCreate', message => {
  if (message.content === '!shutdown' && message.author.id === 'ايدي صاحب البوت') {
    message.channel.send('Shutting down...').then(() => {
      client.destroy();
    });
  }
});
```

7-كود رسالة ويلكوم

```
client.on('guildMemberAdd', member => {
  const channel =
```

```
member.guild.channels.cache.find(channel
 => channel.name === 'welcome');
  if (channel) {
    const embed = new MessageEmbed()
      .setTitle(`Welcome to the server, $
{member.displayName}`)
.setDescription(`We are glad to have you
with us!`)
      .setColor('#0099ff');
    channel.send({ embeds: [embed] });
  }
});
```

8-كود إزاله رتبه من العضو

```
client.on('messageCreate', message => {
  if (message.content.startsWith('!remove-
role')) {
    const member =
message.mentions.members.first();
    const role =
message.guild.roles.cache.find(role =>
```

```
role.name === 'Role Name');
    if (member && role) {
      member.roles.remove(role);
      message.channel.send(`Removed role ${
role.name} from ${member}`);
    }
  }
});
```

9-كود لو البوت استلم اكتر من رسالة يرد على كل واحده بالرد بتعها

```
const responses = {
'هلا': 'اهلا',
'ازيك': 'الحمدلله',
'سلام': 'باييي',
};
client.on('messageCreate', message => {
  const response =
responses[message.content.toLowerCase(
)];
```

```
  if (response) {
    message.channel.send(response);
  }
});
```

10-كود عشان يخلي البوت يشغل ملف صوتي جوا القناه الصوتيه

```
const { joinVoiceChannel,
createAudioResource, createAudioPlayer,
AudioPlayerStatus } = require('@discordjs/
voice');
const ytdl = require('ytdl-core');

client.on('messageCreate', async message
=> {
  if (message.content === '!play') {
    const connection = joinVoiceChannel({
      channelId:
message.member.voice.channelId,
      guildId: message.guild.id,
      adapterCreator:
```

```
message.guild.voiceAdapterCreator,
    });

    const stream = ytdl('https://
www.youtube.com/watch?
v=dQw4w9WgXcQ', { filter: 'audioonly' });
    const resource =
createAudioResource(stream);
    const player = createAudioPlayer();

    player.play(resource);
    connection.subscribe(player);

    player.on(AudioPlayerStatus.Idle, () => {
      connection.destroy();
    });
  }
});
```

11-إرسال رسالة لما حد يطلع من السيرفر

```
client.on('guildMemberRemove', member
=> {
  const goodbyeChannel =
member.guild.channels.cache.find(channel
 => channel.name === 'اسم القناه');
  if (!goodbyeChannel) return;
  goodbyeChannel.send(`Goodbye, $
{member}!`);
});
```

12-أمر ping

```
client.on('messageCreate', async message
=> {
  if (message.content === '!ping') {
    const startTime = Date.now();
    const response = await
message.channel.send('Pinging...');
    const endTime = Date.now();
```

```
    const ping = endTime - startTime;
    const embed = new MessageEmbed()
      .setTitle('Pong!')
      .setDescription(`Latency: ${ping}ms`)
      .setColor('#00ff00');
    await response.edit({ embeds: [embed]
});
  }
});
```

13-کود بان

```
client.on('messageCreate', async message
=> {
  if (message.content.startsWith('!ban')) {
    const user =
message.mentions.users.first();
    if (!user) return message.reply('You must
mention a user to ban.');
    const member =
message.guild.members.cache.get(user.id)
;
```

```
    if (!member) return message.reply('That
user is not in this server.');
    const reason = message.content.slice(5
+ user.id.length).trim() || 'No reason
provided.';
    try {
      await member.ban({ reason });
      message.channel.send(`${user.tag}
has been banned.`);
    } catch (error) {
      console.error(`Error banning user $
{user.tag}: ${error}`);
      message.channel.send(`An error
occurred while trying to ban ${user.tag}.`);
    }
  }
});
```

14-الغاء البان

```
client.on('messageCreate', async message
```

```
=> {
  if (message.content.startsWith('!unban')) {
    const userId =
message.content.slice(7).trim();
    if (!userId) return message.reply('You
must provide a user ID to unban.');
    try {
      await
message.guild.members.unban(userId);
      message.channel.send(`User with ID $
{userId} has been unbanned.`);
    } catch (error) {
      console.error(`Error unbanning user
with ID ${userId}: ${error}`);
      message.channel.send(`An error
occurred while trying to unban user with ID
${userId}.`);
    }
  }
});
```

15-كود طرد عضو من السيرفر

```
client.on('messageCreate', async message
=> {
  if (message.content.startsWith('!kick')) {
    const user =
message.mentions.users.first();
    if (!user) return message.reply('You must
mention a user to kick.');
    const member =
message.guild.members.cache.get(user.id)
;
    if (!member) return message.reply('That
user is not in this server.');
    const reason = message.content.slice(5
+ user.id.length).trim() || 'No reason
provided.';
    try {
      await member.kick(reason);
      message.channel.send(`${user.tag}
has been kicked.`);
```

```
    } catch (error) {
      console.error(`Error kicking user ${user.tag}: ${error}`);
      message.channel.send(`An error occurred while trying to kick ${user.tag}.`);
    }
  }
});
```

16-كود ميوت

```
client.on('messageCreate', async message => {
  if (message.content.startsWith('!mute')) {
    const user = message.mentions.users.first();
    if (!user) return message.reply('You must mention a user to mute.');
    const member = message.guild.members.cache.get(user.id)
```

```
;
    if (!member) return message.reply('That
user is not in this server.');
    const reason = message.content.slice(5
+ user.id.length).trim() || 'No reason
provided.';
    const mutedRole =
message.guild.roles.cache.find(role =>
role.name === 'Muted');
    if (!mutedRole) {
      try {
        mutedRole = await
message.guild.roles.create({
          name: 'Muted',
          color: '#000000',
          permissions: []
        });

message.guild.channels.cache.forEach(as
ync (channel, id) => {
          await
```

```
channel.permissionOverwrites.create(mute
dRole, {
          SEND_MESSAGES: false,
          ADD_REACTIONS: false
        });
      });
    } catch (error) {
      console.error(`Error creating or setting
permissions for Muted role: ${error}`);
      message.channel.send('An error
occurred while creating or setting
permissions for Muted role.');
      return;
    }
  }
  try {
    await member.roles.add(mutedRole);
    message.channel.send(`${user.tag}
has been muted.`);
  } catch (error) {
    console.error(`Error muting user $
```

```
{user.tag}: ${error}`);
      message.channel.send(`An error
occurred while trying to mute ${user.tag}.`);
    }
  }
});
```

17-كود إلغاء الميوت

```
client.on('messageCreate', async message
=> {
  if (message.content.startsWith('!
unmute')) {
    const user =
message.mentions.users.first();
    if (!user) return message.reply('You must
mention a user to unmute.');
    const member =
message.guild.members.cache.get(user.id)
;
    if (!member) return message.reply('That
```

```
user is not in this server.');
    const mutedRole =
message.guild.roles.cache.find(role =>
role.name === 'Muted');
    if (!mutedRole) return
message.reply('There is no "Muted" role in
this server.');
    try {
      await
member.roles.remove(mutedRole);
      message.channel.send(`${user.tag}
has been unmuted.`);
    } catch (error) {
     console.error(`Error unmuting user $
{user.tag}: ${error}`);
      message.channel.send(`An error
occurred while trying to unmute $
{user.tag}.`);
    }
  }
});
```

18-کود تایم اوت

```
client.on('messageCreate', async message
=> {
  if (message.content.startsWith('!
timeout')) {
    const user =
message.mentions.users.first();
    if (!user) return message.reply('You must
mention a user to timeout.');
    const member =
message.guild.members.cache.get(user.id)
;
    if (!member) return message.reply('That
user is not in this server.');
    const duration =
parseInt(message.content.replace(/^!
timeout\s+(\d+)\s+.+$/, '$1'), 10) || 0;
    if (duration <= 0 || duration > 1440)
return message.reply('You must provide a
duration between 1 and 1440 minutes.');
```

```
    const reason =
message.content.slice(10 +
duration.toString().length).trim() || 'No
reason provided.';
    const mutedRole =
message.guild.roles.cache.find(role =>
role.name === 'Muted');
    if (!mutedRole) {
      try {
        mutedRole = await
message.guild.roles.create({
          name: 'Muted',
          color: '#000000',
          permissions: []
        });

message.guild.channels.cache.forEach(as
ync (channel, id) => {
          await
channel.permissionOverwrites.create(mute
dRole, {
```

```
            SEND_MESSAGES: false,
            ADD_REACTIONS: false
          });
        });
      } catch (error) {
        console.error(`Error creating or setting
permissions for Muted role: ${error}`);
        message.channel.send('An error
occurred while creating or setting
permissions for Muted role.');
        return;
      }
    }
    try {
      await member.roles.add(mutedRole);
      message.channel.send(`${user.tag}
has been timed out for ${duration}
minutes.`);
      setTimeout(async () => {
        await
member.roles.remove(mutedRole);
```

```
        message.channel.send(`${user.tag} has been unmuted after a timeout.`);
      }, duration * 60 * 1000);
    } catch (error) {
      console.error(`Error timing out user ${user.tag}: ${error}`);
      message.channel.send(`An error occurred while trying to timeout ${user.tag}.`);
    }
  }
});
```

19-كود إلغاء التايم اوت

|||لو عايز تلغي التايم اوت هتستخدم نفس كود الغاء الميوت عادي||

20-لوج لما اي رسالة تنحذف

```
client.on('messageDelete', message => {
```

```
  message.channel.send(`A message from
${message.author.username} was deleted:
${message.content}`);
});
```

21-أضفاه ايموجي على رسالة

```
const message = await
channel.send('Click the reaction to get a
role!');

message.react('👍');
```

22-لوج لما حد ياخد رتبه

```
client.on('guildMemberUpdate', async
(oldMember, newMember) => {
  const addedRoles =
newMember.roles.cache.filter(role => !
oldMember.roles.cache.has(role.id));
  const addedRoleNames =
addedRoles.map(role => role.name).join(',
```

```
');

  if (addedRoleNames.length > 0) {
    const logChannel =
newMember.guild.channels.cache.get('log
ChannelID');
    logChannel.send(`$
{newMember.user.tag} was given the
role(s): ${addedRoleNames}`);
  }
});
```

23-لوج لما حد ينشال منه رتبه

```
client.on('guildMemberUpdate', async
(oldMember, newMember) => {
  const removedRoles =
oldMember.roles.cache.filter(role => !
newMember.roles.cache.has(role.id));
  const removedRoleNames =
removedRoles.map(role =>
```

```
role.name).join(', ');

  if (removedRoleNames.length > 0) {
    const logChannel =
newMember.guild.channels.cache.get('log
ChannelID');
    logChannel.send(`$
{newMember.user.tag} had the role(s) $
{removedRoleNames} removed.`);
  }
});
```

24-لوج لما حد يعدل على رسالة

```
client.on('messageUpdate', async
(oldMessage, newMessage) => {
  const logChannel =
newMessage.guild.channels.cache.get('log
ChannelID');
  logChannel.send(`A message by $
{newMessage.author.tag} was edited in $
```

```
{newMessage.channel.name}:\nOld
message: ${oldMessage.content}\nNew
message: ${newMessage.content}`);
});
```

25-لوج لما حد يحط رياكشن

```
client.on('messageReactionAdd', async
(reaction, user) => {
  const logChannel =
reaction.message.guild.channels.cache.get
('logChannelID');
  logChannel.send(`Reaction $
{reaction._emoji.name} added by $
{user.tag} to message: $
{reaction.message.content}`);
});
```

26-لوج لما حد يشيل رياكشن

```
client.on('messageReactionRemove',
```

```
async (reaction, user) => {
  const logChannel =
reaction.message.guild.channels.cache.get
('logChannelID');
  logChannel.send(`Reaction $
{reaction._emoji.name} removed by $
{user.tag} from message: $
{reaction.message.content}`);
});
```

27-لوج لما حد يعمل انفايت

```
client.on('inviteCreate', async invite => {
  const logChannel =
invite.guild.channels.cache.get('logChannel
ID');
  logChannel.send(`Invite created: $
{invite.url}`);
});
```

28-لوج لو في قناه جديده اتعملت

```
client.on('channelCreate', async channel =>
{
  const logChannel =
channel.guild.channels.cache.get('logChan
nelID');
  logChannel.send(`Channel created: $
{channel.name}`);
});
```

29-كود لو في صلاحيات رتبة اتغيرت

```
client.on('roleUpdate', async (oldRole,
newRole) => {
  const logChannel =
newRole.guild.channels.cache.get('logCha
nnelID');

  if (oldRole.permissions !==
newRole.permissions) {
    logChannel.send(`Permissions for role $
```

```
{newRole.name} have been updated`);
  }

  if (oldRole.name !== newRole.name) {
    logChannel.send(`Role ${oldRole.name}
has been renamed to ${newRole.name}`);
  }
});
```

30-لوج لو حد خد بان

```
client.on('guildBanAdd', async (guild, user)
=> {
  const logChannel =
guild.channels.cache.get('logChannelID');
  logChannel.send(`${user.tag} has been
banned from the server`);
});
```

31-لوج لو حد انشال منه البان

```
client.on('guildBanRemove', async (guild,
user) => {
  const logChannel =
guild.channels.cache.get('logChannelID');
  logChannel.send(`${user.tag} has been
unbanned from the server`);
});
```

32-لوج لو في رتبة انعملت

```
client.on('roleCreate', async role => {
  const logChannel =
role.guild.channels.cache.get('logChannelI
D');
  logChannel.send(`Role ${role.name}
created`);
});
```

33-لوج لو في رتبه اتمسحت

```
client.on('roleDelete', async role => {
```

```
  const logChannel =
role.guild.channels.cache.get('logChannelI
D');
  logChannel.send(`Role ${role.name}
deleted`);
});
```

34-لوج لو في قناه اتمسحت

```
client.on('channelDelete', async channel =>
{
    const logChannel =
channel.guild.channels.cache.get('logChan
nelID');
    logChannel.send(` channel deleted: $
{channel.name}`);
});
```

35-كود اوتو تاكس

```
client.on('messageCreate', async message
```

```
=> {
  if (message.author.bot ||
message.channel.id !== 'ايدي القناه') return;

  const match =
message.content.trim().match(/^(\d+)
([kmb])?$/i);
  if (!match) return;

  let number = parseInt(match[1]);
  if (isNaN(number)) return;

  const unit = match[2]?.toLowerCase();
  if (unit === 'k') {
    number *= 1000;
  } else if (unit === 'm') {
    number *= 1000000;
  } else if (unit === 'b') {
    number *= 1000000000;
  }
   else if (unit === 't') {
```

```
    number *= 10000000000000000;
  }
  const response = await fetch(`https://
lightgreenorderlyusers.arzex27.repl.co/tax/
${number}`);
  const tax = await response.text();

  const embed = new MessageEmbed()
    .setTitle('💵 **ضريبة البروبوت** 💵')
    .setDescription(`الرقم: **$
{number.toLocaleString()}**\nالضريبة كاملة: $
{tax}`)
    .setColor('#030202')
    .setFooter('bot by ZEĐLEX');

  message.reply({ embeds: [embed] });
});
</> </> </> </> </> </>
</> </> </> </> </> </>
```

</> </> </> </> </> </>

خاتمة:

بعد قراءة هذا الكتاب، يجب أن يمتلك المطورون المبتدئون والمتقدمون المهارات والمعرفة اللازمة لتصميم وتطوير بوتات Discord باستخدام Discord.js v13. ومن خلال قرائه الكتاب وفهمه جيدا سيتمكن المبرمجين ببرمجه بوتات بسهوله انشاءالله

يشمل هذا الكتاب جميع المواضيع الرئيسية لبرمجة البوتات، بما في ذلك تصميم وتفعيل الأوامر، والتفاعل مع قواعد البيانات، والتعامل مع الرسائل والأحداث والمزيد، وتم شرح كل جزء منها بالتفصيل.

وفي النهاية، نود أن نشكركم على قراءة هذا الكتاب، ونتمنى أن يكون قد أضاف قيمة لمعرفتك ومهاراتك في تصميم بوتات Discord. نحن نقدر جهودكم في تطوير مجتمعات Discord، ونتمنى لكم النجاح

والتوفيق فيما تقومون به. شكراً لكم والسلام عليكم ورحمه الله وبراكته

تمت الكتابه بواسطه

الاسم | علي محمد علي

مع افضل التمنيات