

# MFB 機器人(中文版)

*\*Discord 是現時流行的聊天程式*

MFB Discord 機器人(以下會簡稱為 MFB)是一個使用 JAVA 語言和 JDA 的環境製造的功能機器人。MFB 是為了讓 Discord 用戶可以增加管理效率為目標開發而成。開發初期考慮到使用 Discord 的用戶可能來自不同國家，所以 MFB 裏使用了三種語言,包括中文(繁體,簡體)、英文和日文.如圖所視



每一個 IETF 語言標籤都代碼一個語言，而 ServerID.json 會保存每一個 Discord 伺服器的主要使用語言。而且當用戶更換 Discord 伺服器的主要使用語言,ServerID.json 也會按照需求更改。

除此之外,MFB 在開發途中加入了金錢系統,增加 MFB 的趣味性。而且每個群組有一個獨立的金錢系統，互不相通。而且如果你想獲得金錢,你只能輸入指令進行工作,但不一定可以成功工作。除了工作外,MFB 有小遊戲可以賺金錢。

MFB 在初期為了安全和穩定使用了由 JET BRAINS 開發的 IDEA，這能保障用戶的資料不會遺失。

上面提及到，語言和金錢都使用了 Json 格式並且儲存到 Json 檔案中，為了方便起見使用了，使用了現時流行的 Google Gson 也稱為 Gson。使用 Gson 時為了合理將代碼減少，我把單獨所需的功能合併。如下圖所視

```
public static Boolean WriteJsonFile(String FilePort, String Key, String Text) {
    try {
        FileReader fileReader = new FileReader(FilePort);
        BufferedReader bufferedReader = new BufferedReader(fileReader);
        String jsonString = "";
        String line;

        while ((line = bufferedReader.readLine()) != null) {
            jsonString += line;
        }
        bufferedReader.close();
        JSONObject jsonObject = new JSONObject(jsonString);
        jsonObject.put(Key, Text);
        String up = jsonObject.toString();
        FileWriter fileWriter = new FileWriter(FilePort);
        BufferedWriter bufferedWriter = new BufferedWriter(fileWriter);
        bufferedWriter.write(up);
        bufferedWriter.close();
        return true;
    } catch (JSONException | IOException e) {
        System.out.println("上傳失敗，錯誤： 0x00000017");
        return false;
    }
}
```

```
public static String readJsonFile_All(String FilePort) {
    try {
        StringBuilder stringBuilder = new StringBuilder();
        BufferedReader bufferedReader = new BufferedReader(new FileReader(new File(FilePort)));
        String line;
        while ((line = bufferedReader.readLine()) != null) {
            stringBuilder.append(line);
        }
        bufferedReader.close();
        return stringBuilder.toString();
    } catch (FileNotFoundException e) {
        System.out.println("系統找不到指定的路徑。 錯誤： 0x00000017");
    } catch (IOException e) {
        System.out.println("系統找不到指定的路徑。 錯誤： 0x00000017");
    }
    return null;
}
```

我把多個功能合併後，可以有效減少程式的運行時間。而且對於不同語言的使用，只需提供語言的關鍵字即可。如下面代碼所視:

```
LangStructure.Detection(event.getGuild().getIdLong(), "Playing")
```

[檔案名稱].[區塊]{群組所用語言代碼, "代碼關鍵字"}

解決了多個檔案使用重複的代碼。

# ！ 注意 ！

禁止進行任何的抄襲動作！

嚴禁盜用本儲存庫代碼另行架設機器人或使用本儲存庫代碼進行二創！

## Attentive

Any plagiarism is prohibited!

It is strictly prohibited to steal the code of this repository to set up another robot or use the code of this repository for secondary creation!