

1. //バックエンドのphpをフォームから起動し非同期通信でフロントエンドに返す

2.

3. [index.html]-----

4. <!--テキストエリアと書き込みボタンフォームを作る-->

5. <form action="backend.php" method="POST">

6. <textarea name="comment"></textarea>

7. <input type="submit" value="書き込む">

8. </form>

9. <div class="box"></div>

10. <!--jQuery-->

11. <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script>

12. <script>

13. \$("form").on("submit", function(e) {

14. e.preventDefault();

15. let formData = new FormData(this);

16. \$.ajax({

17. url: "backend.php",

18. type: "POST",

19. data: formData,

20. contentType: false,

21. processData: false,

22. success: function(result) {

23. \$(".box").append("<p>" + result + "</p>");

24. }

25. })

26. })

27. </script>

28.

29. □function(e)のeはイベントオブジェクトといい、ユーザーイベントがあった時に自動的にeの中に色々な情報がセットされる

30. □e.preventDefault();は本来の挙動をキャンセルする（ここではフォーム送信でのブラウザ更新をキャンセル）

31. □let formData=new FormData(this);はフォームのコントロール（inputやtextareaのデータ）を格納する命令。

32. 難しく言うと、this（フォーム自体）のFormDataクラスのインスタンスとしてformDataを作ったという言い方をする。

33. □\$.ajaxは\$.getメソッドの進化系、()の中に

34. {

35. キー:"値",

36. キー:"値",

37. }

38. の形でプロパティを追加していく。

39. 主なキーは

40. type:"送信タイプ（POSTやGET）",

41. url:"送信先のファイルパス（URL）",

42. data:"送信するデータ", ...ここではformのinputやtextareから受け取ったデータである変数formData

43. success:function(data){

44. //やりたい処理・dataにはバックエンドでechoされたものが入っている

45. }

46. -----

47.

48.

49. [backend.php]-----

50. <?php

51. echo \$_POST["comment"];

52. ?>

53. -----

54. //\$.ajaxで非同期にバックエンドからの情報を引き出しました。

55. //これでブラウザが変遷することなくフロントエンド側にデータが送られます。

56. //次はコメントを保存する機能をバックエンドでテキストファイルを作成して呼び出すことで実装しようと思います。