1. Di Indonesia, ada pecahan mata uang rupiah, yaitu :

\* 100.000,-

\* 50.000,-

\* 20.000,-

\* 10.000,-

\* 5.000,-

\* 2.000,-

\* 1.000,-

\* 500,-

\* 200,-

\* 100,-

Buatlah sebuah fungsi untuk menghitung berapa lembar pecahan yang harus dikeluarkan dari input harga (dengan pembulatan ke atas jita punya harga pecahan antara 1 sampai 99)

Input : 145.000

Output:

{

“Rp. 100.000”:1,

“Rp. 20.000”:2,

“Rp. 5.000”:1,

}

Input: 2050

Ouput:

{

“Rp. 2.000”:1,

“Rp. 100”:1,

}

1. func InToNominal(num int) map[string]int {

2. var result = make(map[string]int)

3.

4. for num > 0 {

5. switch {

6. case num >= 100000:

7. num -= 100000

8. result["Rp. 100.000"] += 1

9. case num >= 50000:

10. num -= 50000

11. result["Rp. 50.000"] += 1

12. case num >= 20000:

13. num -= 20000

14. result["Rp. 20.000"] += 1

15. case num >= 10000:

16. num -= 10000

17. result["Rp. 10.000"] += 1

18. case num >= 5000:

19. num -= 5000

20. result["Rp. 5.000"] += 1

21. case num >= 2000:

22. num -= 2000

23. result["Rp. 2.000"] += 1

24. case num >= 1000:

25. num -= 1000

26. result["Rp. 1.000"] += 1

27. case num >= 500:

28. num -= 500

29. result["Rp. 500"] += 1

30. case num >= 200:

31. num -= 200

32. result["Rp. 200"] += 1

33. case num >= 100:

34. num -= 100

35. result["Rp. 100"] += 1

36. }

37. }

38.

39. return result

40. }

41.

Source code:

1. Anda diminta untuk membuat sebuah function dimana function tersebut berfungsi untuk menentukan apakah dari dua data string yang diberikan membutuhkan sekali proses edit atau lebih. Jika lebih dari sekali proses edit berarti function tersebut akan mengembalikan response False, sedangkan jika hanya sekali proses edit maka function tersebut akan mengembalikan response True. Proses edit di sini dapat berarti melakukan insert sebuah character, remove sebuah character, atau replace sebuah character.

Contoh

1. GIVEN INPUT 1 → telkom

2. GIVEN INPUT 2 → telecom

3. RESULT → False

5. GIVEN INPUT 1 → telkom

6. GIVEN INPUT 2 → tlkom

7. RESULT → True

func Matcher(textone string, texttwo string) bool {

var i = 0

var diff = 0

if math.Abs(float64(len([]rune(textone))-len([]rune(texttwo)))) > 1 {

return false

} else if len([]rune(textone)) > len([]rune(texttwo)) {

texttwo = fmt.Sprintf("%s%s", texttwo, textone[(len(textone)-1):])

} else if len([]rune(textone)) > len([]rune(texttwo)) {

texttwo = strings.TrimSuffix(texttwo, texttwo[(len(texttwo)-1):])

}

if textone == texttwo {

return true

}

for len(textone) > i {

if string([]rune(textone)[i]) != string([]rune(texttwo)[i]) {

diff += 1

}

i++

}

if diff > 1 {

return false

}

return true

}

Source code:

1. Apakah ada kesalahan dari script di bawah ini? Jika ada tolong jelaskan dimana letak kesalahannya dan bagaimana anda memperbaikinya. Jika tidak ada, tolong jelaskan untuk apa script di bawah ini.

1. FROM golang

2. ADD . /go/src/github.com/telkomdev/indihome/backend

3. WORKDIR /go/src/github.com/telkomdev/indihome

4. RUN go get github.com/tools/godep

5. RUN godep restore

6. RUN go install github.com/telkomdev/indihome

7. ENTRYPOINT /go/bin/indihome

8. LISTEN 80

Kesalahan:

* Belum ada command atau perintah yang menyatakan build dari aplikasi
* Penggunaan ```LISTEN 80``` harus menggunakan ```EXPOSE 80``` untuk dibuka port yang digunakan.

Seharusnya:

FROM golang

ADD . /go/src/github.com/telkomdev/indihome/backend

WORKDIR /go/src/github.com/telkomdev/indihome

RUN go get github.com/tools/godep

RUN godep restore

RUN go install github.com/telkomdev/indihome

RUN go build -i -v -o . /go/bin/indihome

ENTRYPOINT ./go/bin/indihome

EXPOSE 80

1. Menurut anda apakah tujuan penggunaan microservices?

Tujuan Penggunaan microservise untuk system atau architecture yang scalable, kemudian dengan microservice memberikan kebebasan developer untuk mengembangkan dan mengintegrasikan service secara mandiri. Microservices juga dapat dijalan kan tanpa harus menjalankan seluruh service.

1. Bagaimana cara index bekerja pada sebuah database?
2. Buat sebuah sebuah backend service Shopping Cart yang harus memilik api:
3. Api tambahProduk

* Attribute kodeProduk, namaProduk, kuantitas
* Menambahkan produk dengan kuantitas yang ditentukan.
* Apabila produk sudah ada di dalam Cart, tambahkan kuantitasnya.

1. Api hapusProduk(string kodeProduk)

* Menghapus produk dari Cart.

1. Api tampilkanCart

* Menampilkan isi Cart dengan format   
  {kodeProduk}**-** {namaProduk} - ({kuantitas })
* Memiliki filter namaProduk dan kuantitas

Buatlah backend service cart berikut feature code dan unit testnya.