

Deskripsi Soal

1. create_a_list_test

Deskripsi: Selesaikan fungsi `create_a_list_test` untuk mengembalikan list berisi 5 dictionary. Setiap dictionary harus memiliki kunci `'nama'`, `'domisili'`, `'suku'` dengan nilai yang unik.

Contoh Output yang Diharapkan:

```
[
    {"nama": "Ali", "domisili": "Jakarta", "suku": "Betawi"},
    {"nama": "Budi", "domisili": "Surabaya", "suku": "Jawa"},
    {"nama": "Citra", "domisili": "Medan", "suku": "Batak"},
    {"nama": "Dina", "domisili": "Bandung", "suku": "Sunda"},
    {"nama": "Eko", "domisili": "Makassar", "suku": "Bugis"}
]
```

Nilai Uji:

- Pastikan list berukuran 5.
- Pastikan setiap elemen adalah dictionary dengan kunci `'nama'`, `'domisili'`, `'suku'`.
- Pastikan setiap nilai dalam kunci tersebut unik.

2. sort_test

Deskripsi: Selesaikan fungsi `sort_test` untuk mengurutkan list dictionary berdasarkan kunci `'nama'` secara alfabetis. Fungsi tidak mengembalikan nilai (sorting dilakukan in-place).

Contoh Input:

```
data_acak = [
    {"nama": "Dina", "domisili": "Bandung", "suku": "Sunda"},
    {"nama": "Ali", "domisili": "Jakarta", "suku": "Betawi"},
    {"nama": "Citra", "domisili": "Medan", "suku": "Batak"},
]
sort_test(data_acak)
```

Contoh Setelah Sorting:

```
[
    {"nama": "Ali", "domisili": "Jakarta", "suku": "Betawi"},
    {"nama": "Citra", "domisili": "Medan", "suku": "Batak"},
    {"nama": "Dina", "domisili": "Bandung", "suku": "Sunda"},
]
```

Nilai Uji:

- Pastikan elemen dalam list diurutkan berdasarkan kunci 'nama' secara alfabetis.
- Fungsi tidak mengembalikan apa pun.

3. stack_test

Deskripsi: Selesaikan fungsi `stack_test` untuk menerima string dan mengembalikannya dalam kondisi terbalik menggunakan konsep stack.

Contoh Input:

```
data = "stack"
```

Contoh Output:

```
"kcats"
```

Nilai Uji:

- Pastikan string terbalik dengan benar.
- Gunakan stack untuk implementasi.

4. queue_test

Deskripsi: Selesaikan fungsi `queue_test` yang menerima list angka merepresentasikan waktu layanan pelanggan di dua kasir. Fungsi harus mengembalikan tuple waktu total layanan kedua kasir.

Contoh Input:

```
customer = [5, 3, 4, 2, 8, 6]
```

Proses:

- Pelanggan pertama (5 detik) -> Kasir 1.
- Pelanggan kedua (3 detik) -> Kasir 2.
- Pelanggan ketiga (4 detik) -> Kasir 2 (karena Kasir 2 selesai lebih cepat).
- Pelanggan keempat (2 detik) -> Kasir 1, dst.

Contoh Output:

```
(15, 17)
```

Nilai Uji:

- Pastikan waktu total layanan di kedua kasir dihitung dengan benar.
- Kasir yang selesai lebih dulu melayani pelanggan berikutnya.