

Nama : Alvio Damayanti

NIM : 222410103032

Kelas : Functional Programming A

Tugas Praktikum 1 dan 2

TUGAS 1

Kode 1

```
1  #Kode 1
2  def sequenceGenerator(start, stop):
3      x = [ ]
4      for i in range(start, stop+1):
5          x.append(i)
6      return x
7  print(sequenceGenerator(1,10))
```

Kode di atas diminta untuk diubah ke dalam 1 line, saya mengubahnya seperti di bawah

```
9  #Kode 1
10 sequenceGenerator_online = lambda start, stop: list(map(lambda i: i, range(start, stop + 1)))
11 print(sequenceGenerator_online(1, 10))
```

- **Menggunakan Lambda:** Di sini saya menggunakan ekspresi lambda “lambda i: i” untuk mendefinisikan fungsi anonim dengan mengambil argumen i dan mengembalikan nilai i (tidak ada operasi tambahan pada i).
- **Menggunakan Fungsi map:** map di sini untuk menerapkan fungsi lambda pada tiap elemennya dalam rentang angka yang dicetak atau dihasilkan oleh range(start, stop + 1), jadi tanpa perlu perulangan yang eksplisit seperti for pada kode soal. Variabel x juga dihilangkan dan diganti dengan map ini.

Output yang dihasilkan dari kedua kode

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Output kode asli

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Output dari kode 1 line

Kode 2

```
14  #Kode 2
15  def fizzBuzz(a,b):
16      x = []
17      for num in range(a,b):
18          if num % 3 == 0 and num % 5 == 0:
19              x.append('FizzBuzz')
20          elif num % 3 == 0:
21              x.append('Fizz')
22          elif num % 5 == 0:
23              x.append('Buzz')
24          else:
25              x.append(num)
26      return x
```

Kode di atas diminta untuk diubah ke dalam 1 line, saya mengubahnya seperti di bawah ini

```

28 #Kode 2
29 fizzBuzz_online = lambda a, b: print('\n'.join(map(lambda num: (num % 3 == 0) * "Fizz" + (num % 5 == 0) * "Buzz" or str(num), range(a, b))))
30 fizzBuzz_online(1, 16)

```

- **Menggunakan Lambda:** lambda di sini digunakan mendefinisikan fungsi anonim yang mengambil dua argumen yaitu a, b. Kemudian menggunakan ekspresi lambda ini untuk menggabungkan “Fizz” dan/atau “Buzz”.
 1. **lambda num: (num % 3 == 0) * “Fizz”**, nantinya akan menghasilkan “Fizz” jika num habis dibagi 3.
 2. **lambda num: (num % 5 == 0) * “Buzz”**, nantinya akan menghasilkan “Buzz” jika num habis dibagi 5.
 3. **(num % 3 == 0) * “Fizz” + (num % 5 == 0) * “Buzz”**, menggabungkan keduanya atau langkah 1 dan 2 dengan operator +, yang artinya num habis dibagi 3 dan juga 5 akan menghasilkan "FizzBuzz"
 4. **or str(num)**, ini digunakan jika nanti langkah 1 sampai 3 adalah string kosong maka akan mengembalikan num yang merupakan angka itu sendiri
- **Menggunakan Fungsi map:** map di sini untuk membuat daftar sesuai dengan rentang angka dari a hingga b. Hasil dari fungsi map ini berupa kata “Fizz”, “Buzz”, “FizzBuzz” atau angka itu sendiri.
- **‘\n’.join:** untuk menggabungkan daftar kata menjadi satu string dengan baris baru di setiap katanya.
- Variabel x yang ada pada soal dihilangkan kemudian diganti dan digabungkan menjadi satu line dengan fungsi lambda dan map.

Contoh Penggunaan dan Output yang dihasilkan

Di sini saya mengimplementasikan dari angka 1-16 (dan mengapa yang keluar hanya hingga 15, dikarenakan tidak menggunakan a, b+1)

Output kode asli:

```
[1, 2, 'Fizz', 4, 'Buzz', 'Fizz', 7, 8, 'Fizz', 'Buzz', 11, 'Fizz', 13, 14, 'FizzBuzz']
```

Output kode 1 line:

```

1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
FizzBuzz

```

Kode 3

```
35 #Kode 3
36 def twoNumber(l):
37     res = []
38     for i in l:
39         if l.index(i) == len(l) - 1:
40             break
41         z = i + l[i+1]
42         res.append(z)
43     return res
```

Kode di atas diminta untuk diubah ke dalam 1 line, saya mengubahnya seperti di bawah ini

```
46 twoNumber_online = lambda l: [l[index] + l[index + 1] for index, _ in enumerate(l[:-1])]
47 print(twoNumber_online([1, 2, 3, 4, 5]))
```

- **Menggunakan Lambda:** lambda di sini digunakan untuk menggantikan def twoNumber pada soal dengan menerima satu argument yaitu (l) yang nantinya menghasilkan daftar hasil penjumlahan.
- **Menggunakan Enumerate:** enumerate di sini untuk mengambil indeks di tiap elemennya dalam daftar l. Fungsi pada soal l.index (i) dihilangkan dan digantikan dengan ini.
- **Menggunakan Slicing:** menggunakan l[:-1] untuk menghindari perluasan dari dafranya sehingga memastikan perulangan hanya hingga elemen terakhir saja.
- Variabel res untuk menyimpan hasil penjumlahan pada soal dihilangkan karena langsung menghasilkan daftar dari hasil penjumlahan.

Contoh Penggunaan dan Output yang dihasilkan

[3, 5, 7, 9] Ouput kode 1 line

Di sini saya menggunakan list angka [1, 2, 3, 4, 5]

TUGAS 2

Ubah kode python dari ketiga soal menjadi kode Dart

Kode 1

```
1 // Kode1
2 List<int> sequenceGeneratorOnline(int start, int stop) => List<int>.generate(stop - start + 1, (i) => start + i);
3 void main() => print(sequenceGeneratorOnline(1, 10));
```

- List<int> sequencegeneratorOnline merupakan fungsi yang nantinya akan menerima dua parameter berupa 'start' dan stop'
- List<int>.generate(stop – start + 1, (i) => start + 1. List generate di sini digunakan untuk membuat daftar yang sesuai dengan jumlah elemen yang dihitung dari stop –

start + 1. Intinya untuk memasukkan bilangan dari start ke stop, termasuk start dan stop itu sendiri).

- a) stop – start + 1 = Jumlah elemen dalam daftar yang akan dihasilkan.
- b) (i) => start + i = Fungsi yang dijalankan untuk setiap elemennya dengan i merupakan indeks elemen saat ini.
- void main() => print(sequenceGeneratorOnline(1,10)). Fungsi main yang nantinya akan dijalankan sehingga akan mencetak hasil.

Output yang dihasilkan

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

Kode 2

```
5 // Kode2
6 void fizzBuzzOnline(int a, int b) => print(Iterable<int>.generate(b - a + 1, (i) => a + i).map((n) => (n % 3 == 0 ? 'Fizz' : '') +
7 (n % 5 == 0 ? 'Buzz' : '') + (n % 3 != 0 && n % 5 != 0 ? n.toString() : '')).take(b - a).join('\n'));
8 void main() => fizzBuzzOnline(1, 16);
```

- void fizzBuzzOnline(int a, int b) merupakan fungsi yang menerima dua parameter yaitu a dan b.
- Iterable<int>.generate(b - a + 1, (i) => a + i). Iterable.generate di sini digunakan untuk membuat iterasi dari bilangan bulat (mulai dari a hingga b). Fungsi (i) => a + i digunakan untuk menghasilkan urutan bilangan bulat dalam iterasi.
- .map((n) => (n % 3 == 0 ? 'Fizz' : '') + (n % 5 == 0 ? 'Buzz' : '') + (n % 3 != 0 && n % 5 != 0 ? n.toString() : '')). Ini nantinya yang akan memeriksa dari bilangan bulat yang ada dalam iterasi.
 - a) (n % 3 == 0 ? 'Fizz' : '') Jika bilangan n habis dibagi 3 akan diubah “Fizz”, jika tidak akan menghasilkan string kosong.
 - b) (n % 5 == 0 ? 'Buzz' : '') Jika bilangan n habis dibagi 5 akan diubah “Buzz”, jika tidak akan menghasilkan string kosong.
 - c) (n % 3 == 0 ? 'Fizz' : '') + (n % 5 == 0 ? 'Buzz' : '') Jika bilangan n habis dibagi 3 dan 5 akan diubah “FizzBuzz”.
 - d) (n % 3 != 0 && n % 5 != 0 ? n.toString() : '') Jika n tidak memenuhi dari ketiganya maka akan diubah n.toString().
- take(b-a) digunakan untuk memotong hasil menjadi panjang yang sesuai yaitu (b-a) agar tidak ada bilangan di luar rentang ini.

- `.join("\n")`, dari hasil yang diperoleh kemudian digabungkan dengan join dan dibuat baris baru.
- `void main() => fizzBuzzOnline(1, 16)`. Fungsi main yang nantinya akan dijalankan sehingga akan mencetak hasil.

Output yang dihasilkan

```
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
FizzBuzz
```

Kode 3

```
10 //Kode3
11 List<int> twoNumberOnline(List<int> l) => List<int>.from(Iterable<int>.generate(l.length - 1).map((i) => l[i] + l[i + 1]));
12 void main() => print(twoNumberOnline([1, 2, 3, 4, 5]));
```

- `List<int> twoNumberOnline(List<int> l)` merupakan fungsi yang menerima satu parameter, yaitu daftar bilangan bulat `l`.
- `List<int>.from(Iterable<int>.generate(l.length - 1).map((i) => l[i] + l[i + 1]))`
 - a) `Iterable<int>.generate(l.length - 1)`. `Iterable.generate` ini nantinya digunakan untuk membuat iterasi dari bilangan bulat yang akan digunakan sebagai index. `(l.length - 1)` untuk menghitung panjang dari daftar list agar tidak melampaui batas saat mengambil dua bilangan.
 - b) `.map((i) => l[i] + l[i + 1])`. Ini akan digunakan dengan mengambil dua bilangan berturut-turut dalam list `l` menggunakan indeks `i` dan `i + 1` kemudian menjumlahkannya.
- `void main () => print(twoNumberOnline([1, 2, 3, 4, 5]))`. Fungsi main yang nantinya akan dijalankan sehingga akan mencetak hasil. Hasil yang dicetak berupa penjumlahan dua bilangan berturut-turut.

Output yang dihasilkan

```
[3, 5, 7, 9]
```