

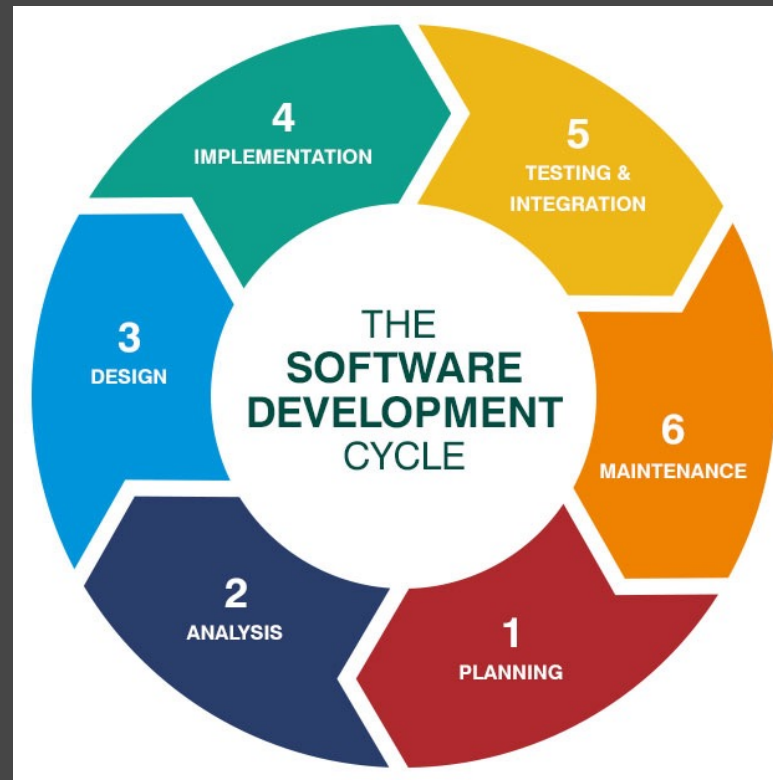


The Steps to Software Development

COMP16321 - Introduction to Programming

Dr. Gareth Henshall

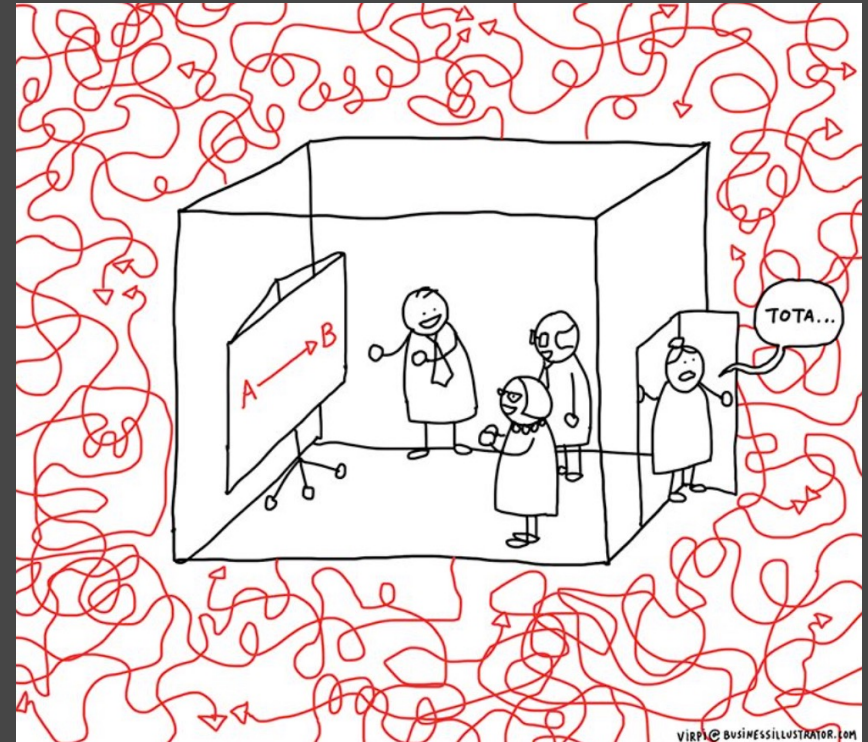
What is Software Development



Why is Software
Development Hard?

No Silver Bullet

"There is no single development, in either technology or management technique, which by itself promises even one order-of magnitude improvement within a decade in productivity, in reliability, in simplicity." - Fred Brooks, 1986



Common Problems Encountered by New Programmers

Understanding the
Problem

Managing Time

Producing Quality
Code

Testing &
Debugging

How do we Solve a
Problem?

Fizz Buzz

Print integers one-to-N, but print "Fizz" if an integer is divisible by three, "Buzz" if an integer is divisible by five, and "FizzBuzz" if an integer is divisible by both three and five.

Step 1: Understand the Problem

Print integers one-to-N, but print "Fizz" if an integer is divisible by three, "Buzz" if an integer is divisible by five, and "FizzBuzz" if an integer is divisible by both three and five.

Step 1: Understand the Problem

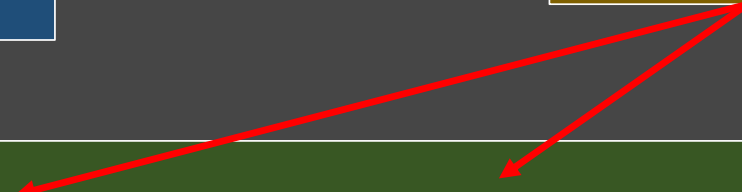
Outputs

Print integers one-to-N, but print "Fizz" if an integer is divisible by three, "Buzz" if an integer is divisible by five, and "FizzBuzz" if an integer is divisible by both three and five.

How do we want it printing?

Step 1: Understand the Problem

Inputs



Print *integers* one-to-*N*, but print "Fizz" if an integer is divisible by three, "Buzz" if an integer is divisible by five, and "FizzBuzz" if an integer is divisible by both three and five.

Step 1: Understand the Problem

Small Steps

Print integers one-to-N, but print "Fizz" if an integer is divisible by three, "Buzz" if an integer is divisible by five, and "FizzBuzz" if an integer is divisible by both three and five.

Step 1: Understand the Problem

Small Steps

- *Print integers one-to-N, but*
- *print "Fizz" if an integer is divisible by three,*
- *"Buzz" if an integer is divisible by five, and*
- *"FizzBuzz" if an integer is divisible by both three and five.*

Step 2: Work Through the Problem Manually



Step 3: Pseudocode

1. *Input n*
2. *Set variable count to 0*
3. *While count does not equal n*
 1. *If count /3 has no remainder, print "Fizz"*
 2. *If count /5 has no remainder, print "Buzz"*
 3. *If both, print "FizzBuzz"*
 4. *If none, print count*
 5. *Add 1 to count*

What is pseudocode?

Pseudocode is a *step-by-step written outline of your code* that you can gradually transcribe into the programming language.

wiki How to Write Pseudocode

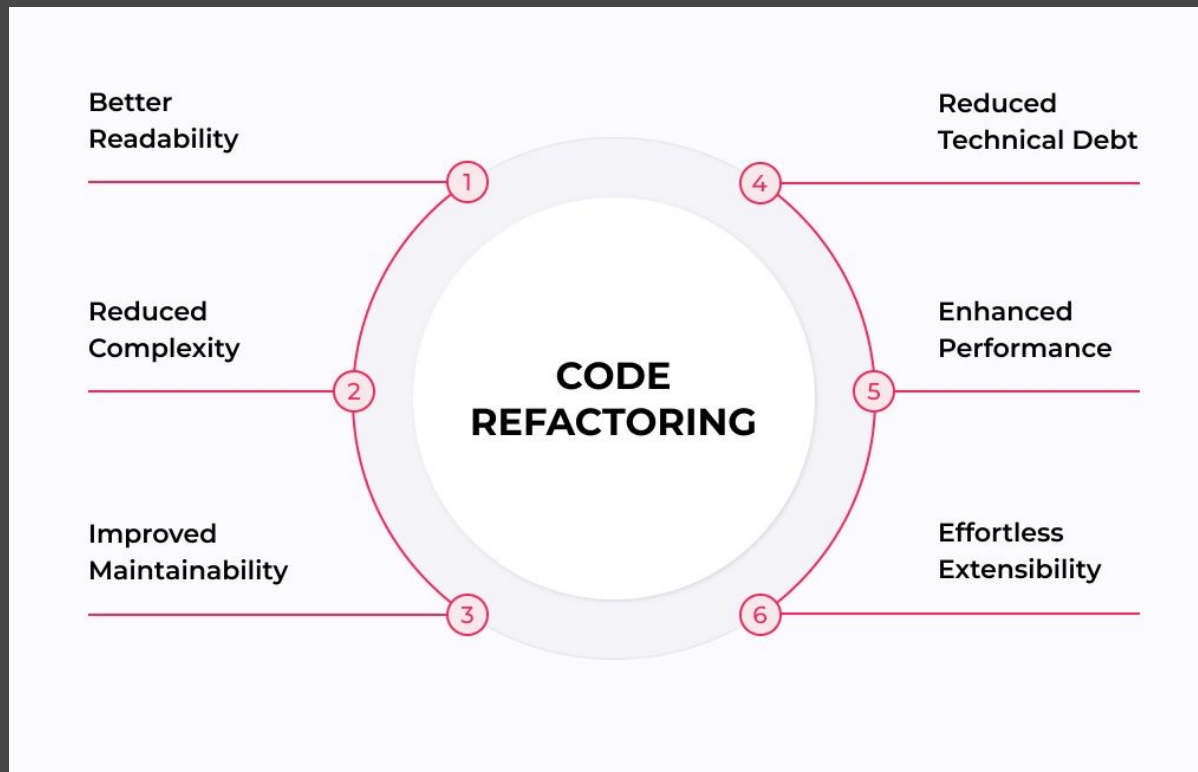
Step 4: Write the Code

```
10 n = int(input("Enter a number: "))
11 c = 0
12
13 while c < n:
14     if (c % 3 == 0):
15         print("Fizz")
16     if (c % 5 == 0):
17         print("Buzz")
18     if (c % 3 == 0 & c % 5 == 0):
19         print("FizzBuzz")
20     else:
21         print(c)
22
23     c = c + 1
24
```

Step 5: Write some Tests

```
9
10 class TestFizzBuzz(unittest.TestCase):
11     def test_fizz(self):
12         for i in [3, 6, 9, 18]:
13             print('testing', i)
14             assert fizzbuzz(i) == 'Fizz'
```


Step 6: Refactor



<https://xbsoftware.com/blog/code-refactoring-techniques-in-software-engineering/>

Step 6: Refactor

v1

```
10 n = int(input("Enter a number: "))
11 c = 0
12
13 while c < n:
14     if (c % 3 == 0):
15         print("Fizz")
16     if (c % 5 == 0):
17         print("Buzz")
18     if (c % 3 == 0 & c % 5 == 0):
19         print("FizzBuzz")
20     else:
21         print(c)
22
23     c = c + 1
24
```

v2

```
10 n = int(input("Enter a number: "))
11 c = 1
12 o = ""
13
14 while c < n:
15     if (c % 3 == 0):
16         o = o + "Fizz"
17     if (c % 5 == 0):
18         o = o + "Buzz"
19     else:
20         o = str(c)
21     print(o)
22     o = ""
23     c = c + 1
```

v3

```
11 print('\n'.join(map(lambda i: 'Fizz' * (i % 3 == 0) + 'Buzz' * (i % 5 == 0) or str(i), range(1, 21))))
```

Step 7: Documentation

```
11 def fizzbuzz(i):
12     """
13     Function for processing what to print values is
14     based on input integer
15
16     Parameters: i(int): current n value
17     Returns: int/string: value of i
18     """
19
20     if i % 15 == 0:
21         return "FizzBuzz"
22     ...
```

That's **FizzBuzz**

But did you need to implement FizzBuzz?

No !

fizzbuzz 0.0.1

`pip install fizzbuzz`



Released: May 21, 2019

Python package which prints Fizz, Buzz, FizzBuzz divisible by 3 and 5 and both

Navigation

Project description

Release history

Download files

Project links

Homepage

Statistics

GitHub statistics:

Stars: 0

Forks: 0

Open issues: 1

Open PRs: 0

View statistics for this project via [Libraries.io](#), or by using [our public dataset on Google BigQuery](#)

Project description

FizzBuzz

Fizz buzz is a group word game for children to teach them about division. Players take turns to count incrementally, replacing any number divisible by three with the word "fizz", and any number divisible by five with the word "buzz"

Fizz buzz Implementation

Simple python program which iterates the integers from 1 to 50. For multiples of three print "Fizz" instead of the number and for the multiples of five print "Buzz". For numbers which are multiples of both three and five print "FizzBuzz"

Usage

Download it by clicking the green download button here on Github. You only need to parse argument of range you the divisible by 3, 5 and both.

```
>> from fizzbuzz.fizzbuzz import looprange
>> looprange('50')
```

Output

Step by Step

- 1. Understand the Problem*
- 2. Work through it Manually*
- 3. Pseudocode*
- 4. Write the Code*
- 5. Test the Code*
- 6. Refactor the Code*
- 7. Write Documentation*

Saving our Work

Version Control!