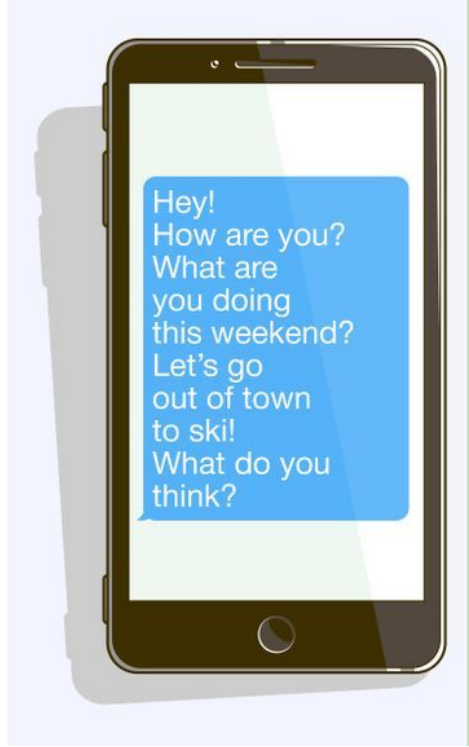
CS 301
Software Engineering
Module - 40

Eswaran Narasimhan

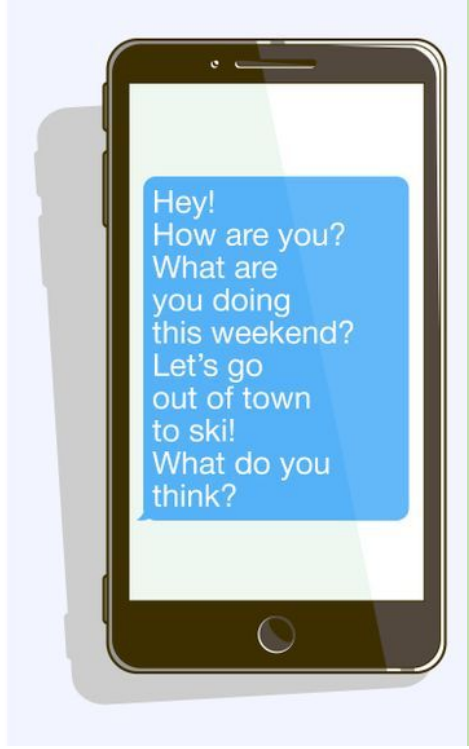
Photo Credits :https://www.facebookcom/bestparto/posts/the-perfect-tree-tunnel-france-photo-by-remo_daut/889001394872676/

Two Types of PEOPLE



Two Types of PEOPLE

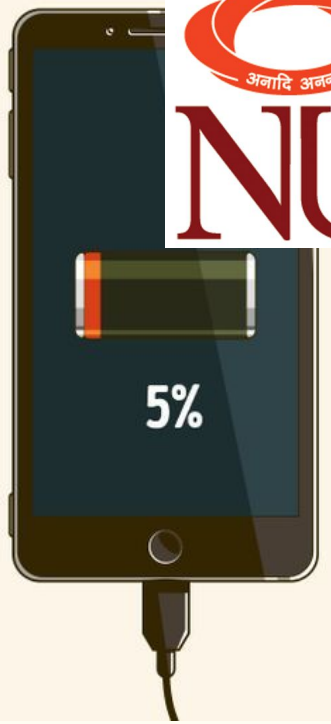
https://brightsideme/wonder-people/there-are-2-types-of-people-in-this-world-17-illustrations-that-prove-it-441610/

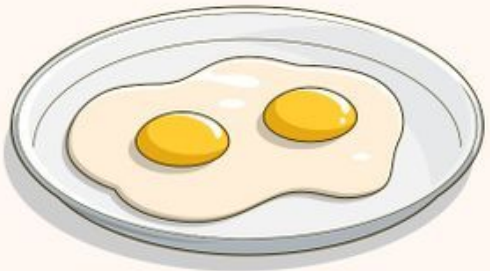


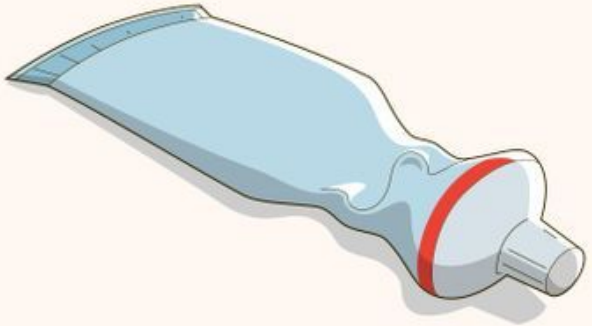
Two Types of PEOPLE

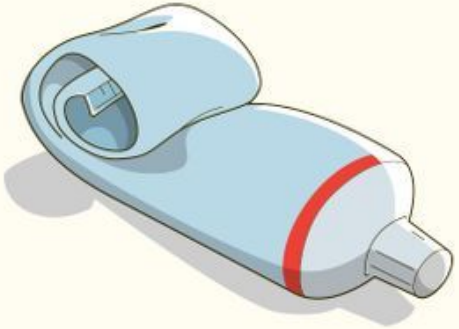
https://brightsideme/wonder-people/there-are-2-types-of-people-in-this-world-17-illustrations-that-prove-it-441610/

Two Types of PEOPLE

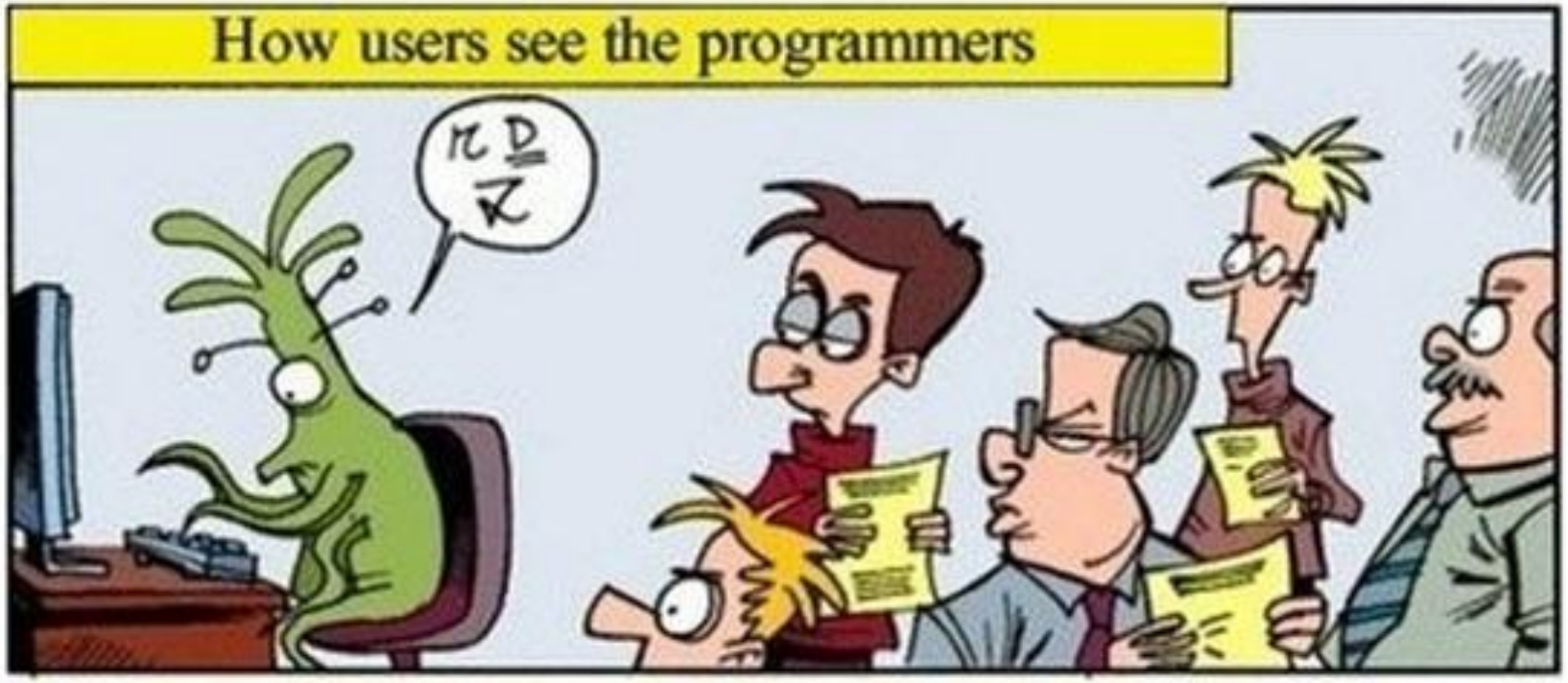


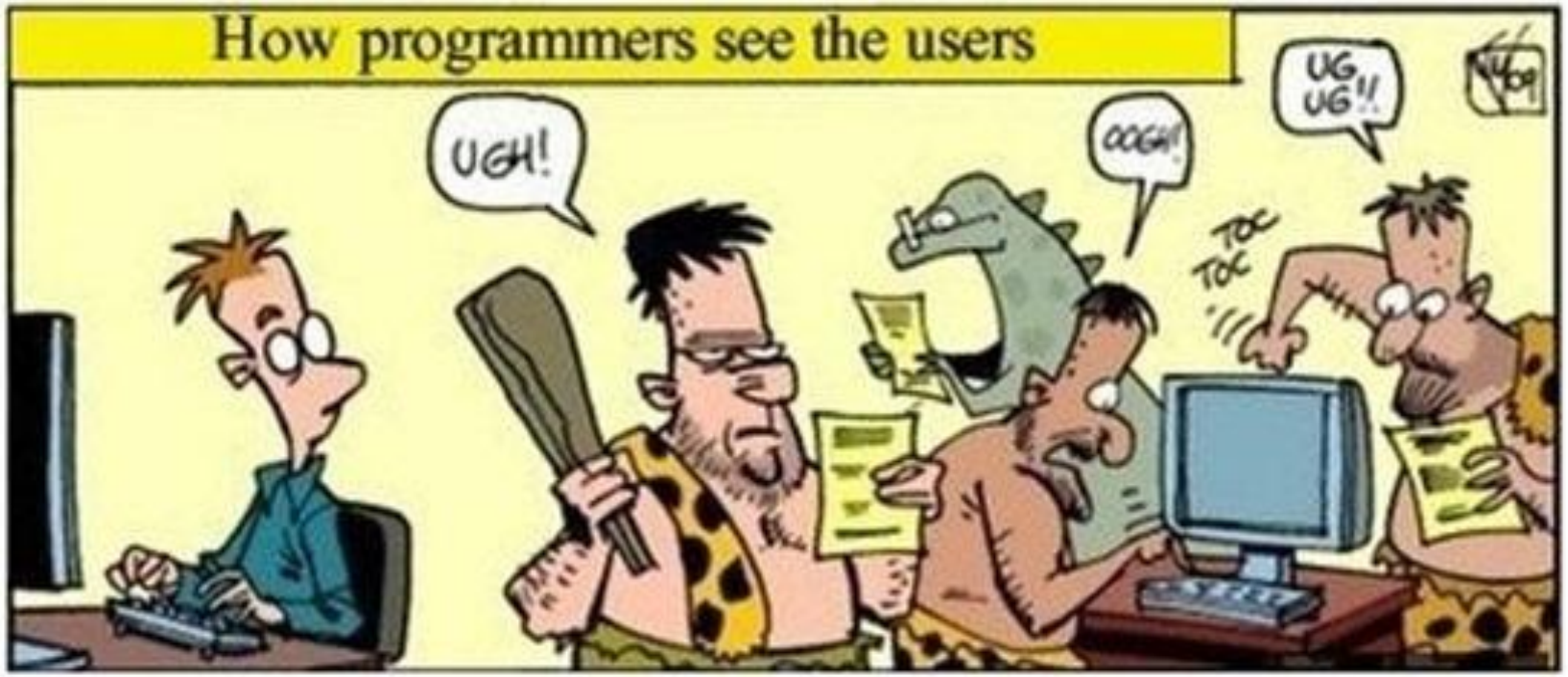




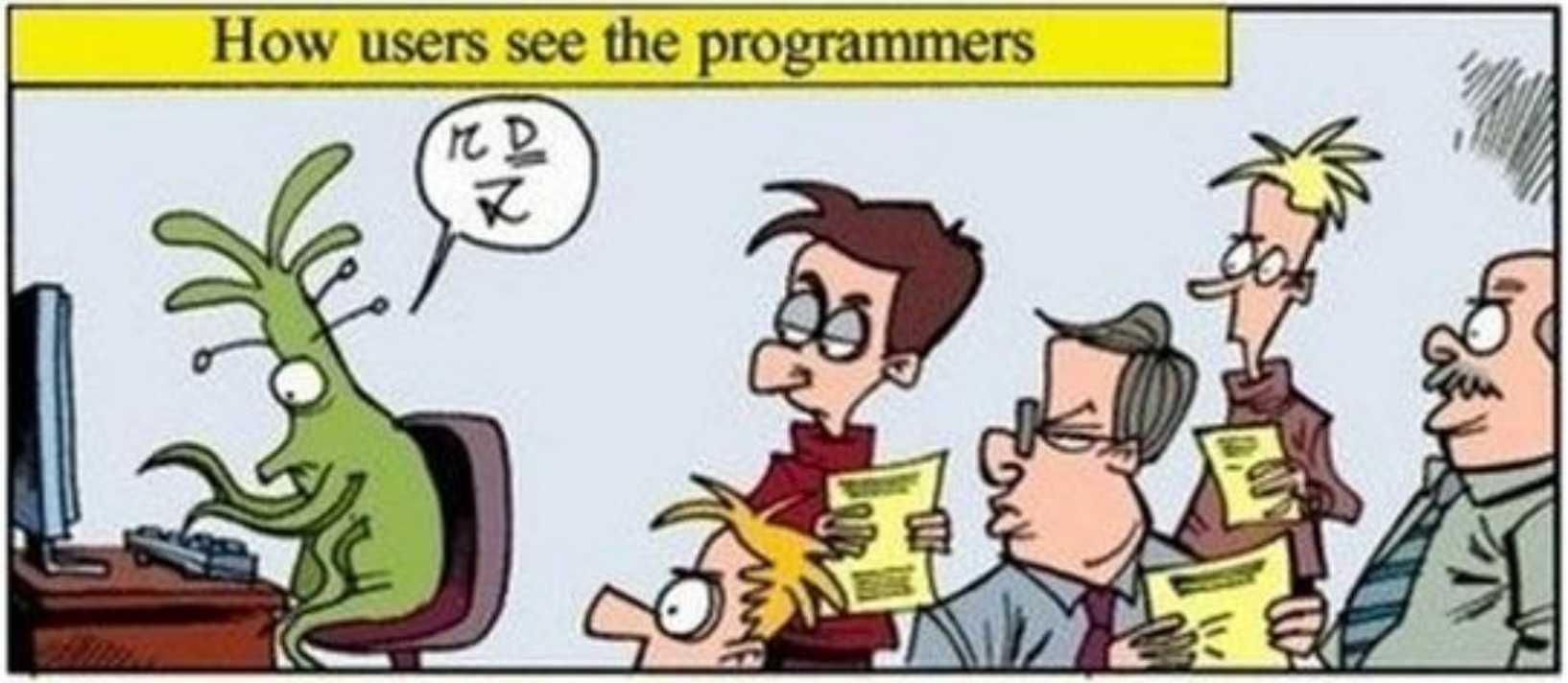


https://brightsideme/wonder-people/there-are-2-types-of-people-in-this-world-17-illustrations-that-prove-it-441610/

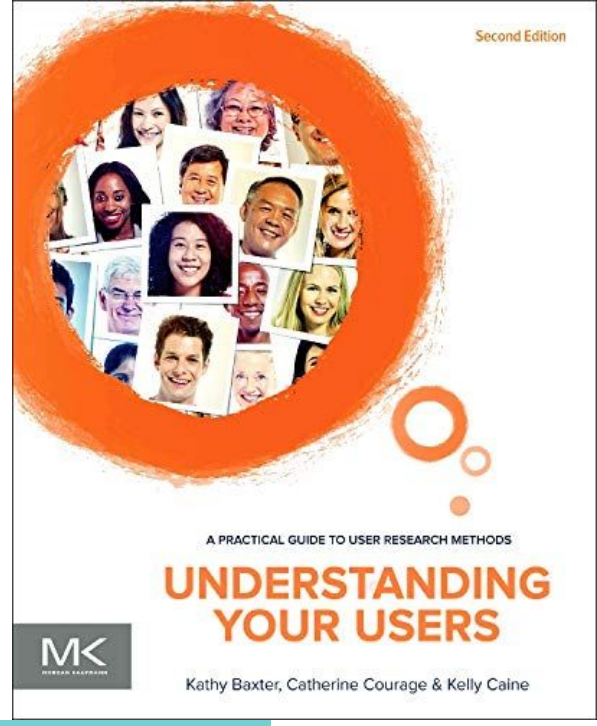




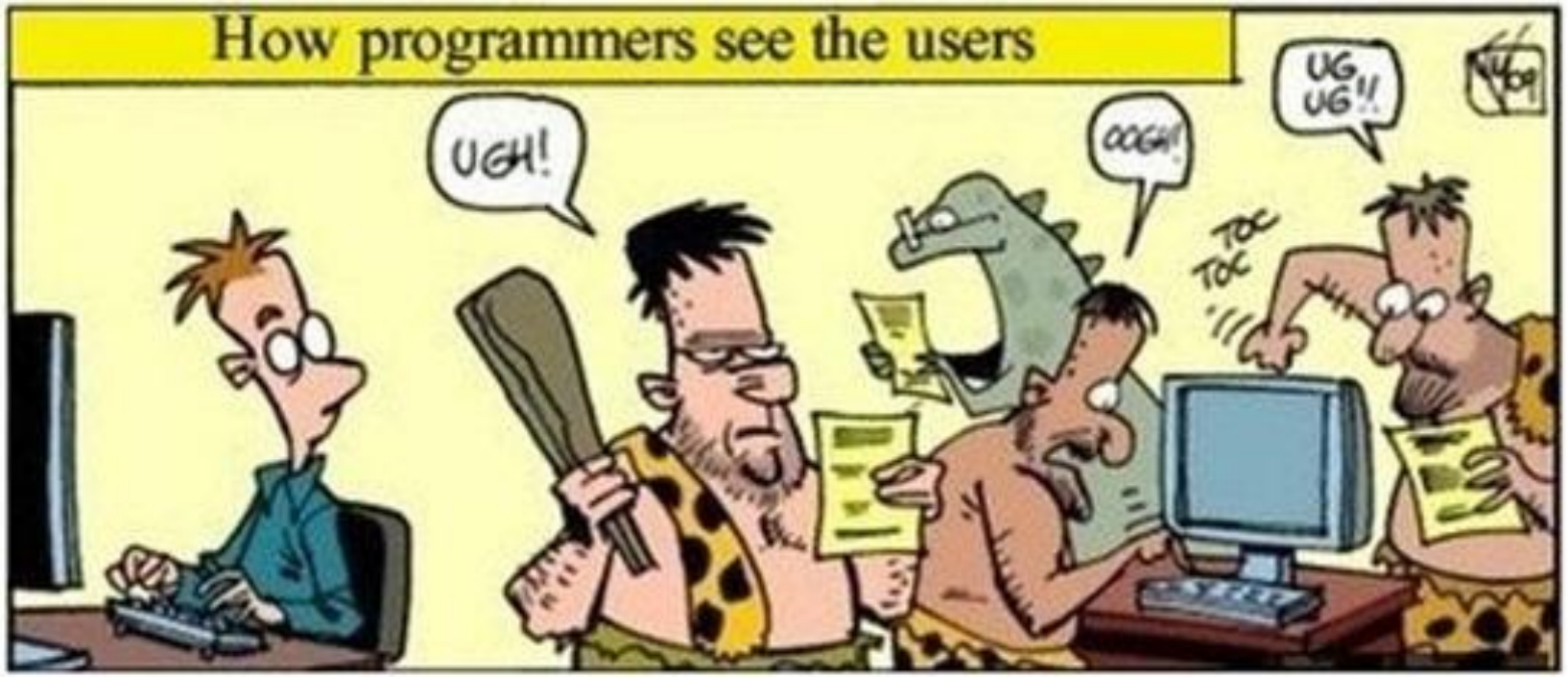
https://msiccnet/editorial-the-relationship-between-developers-and-users/



https://msiccnet/editorial-the-relationship-between-developers-and-users/

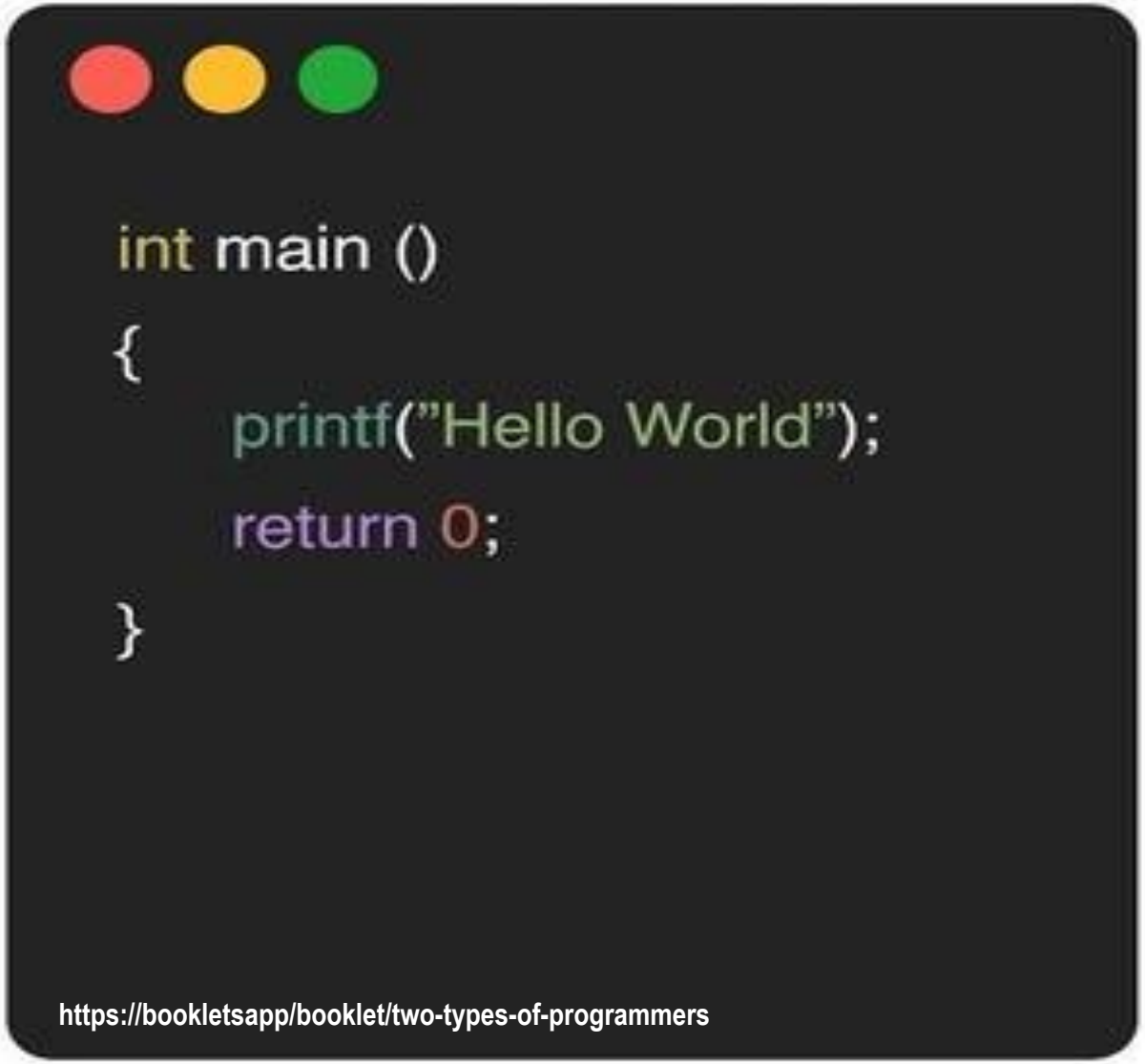


https://wwwamazoncom/Understanding-Your-Users-Interactive-Technologies/dp/0128002328

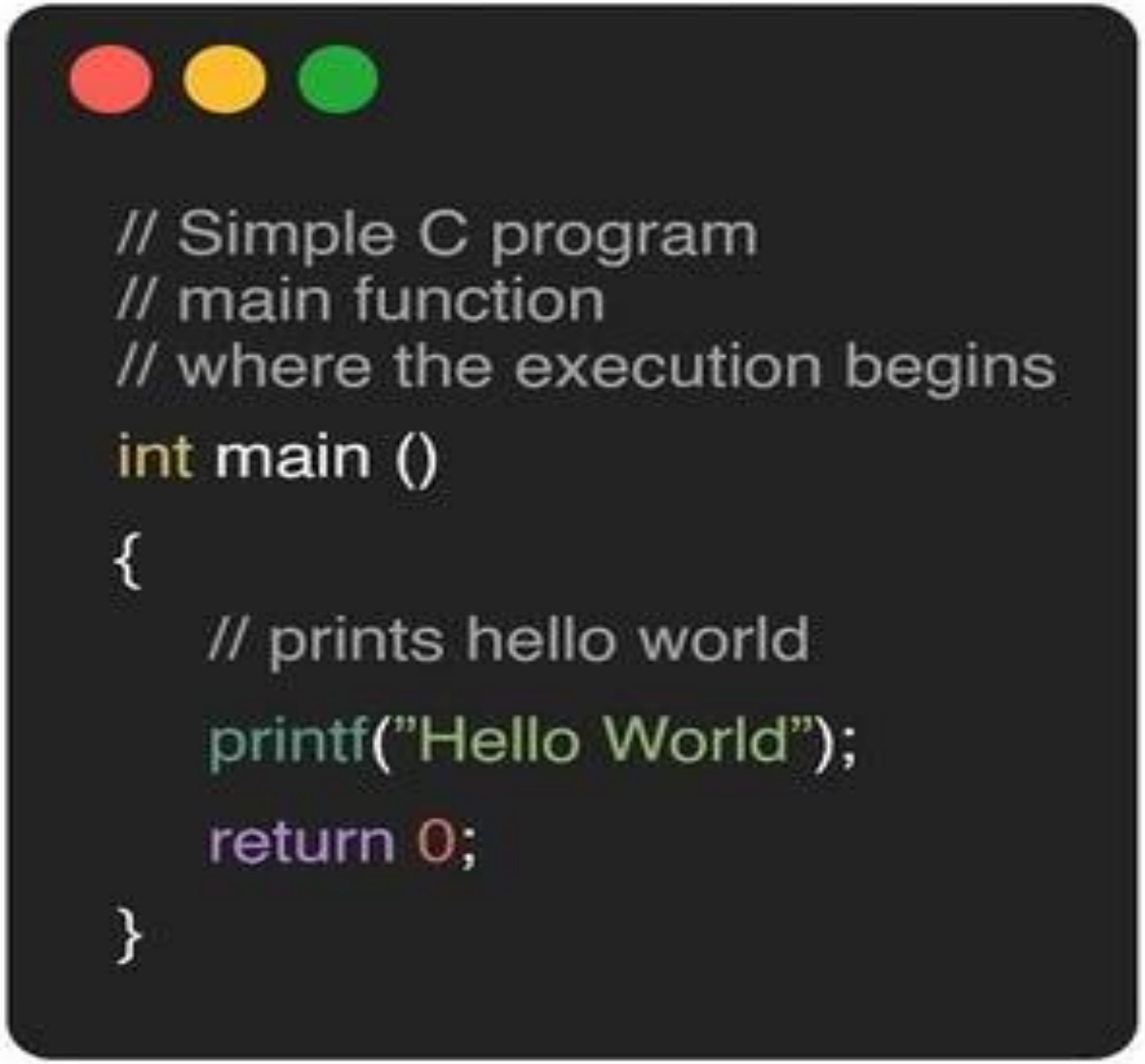


https://devadacom/resources/

TWO TYPES OF PROGRAMMERS



```c
int main ()
{
    printf("Hello World");
    return 0;
}
```

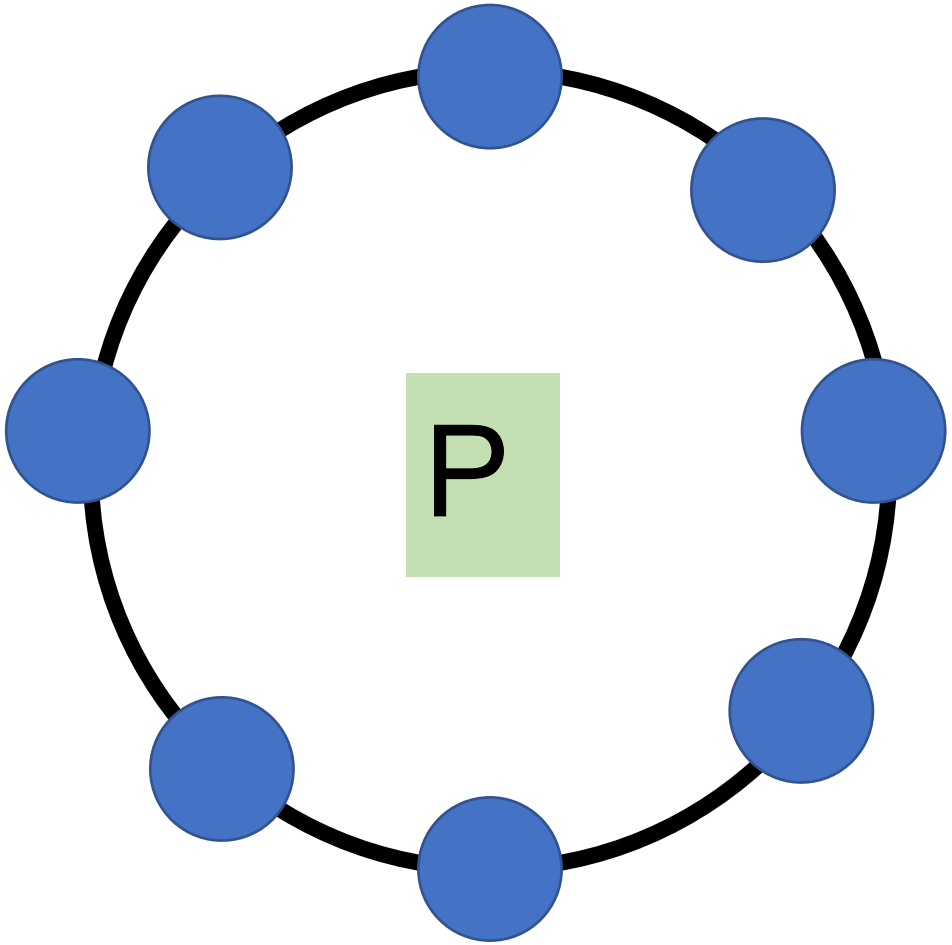


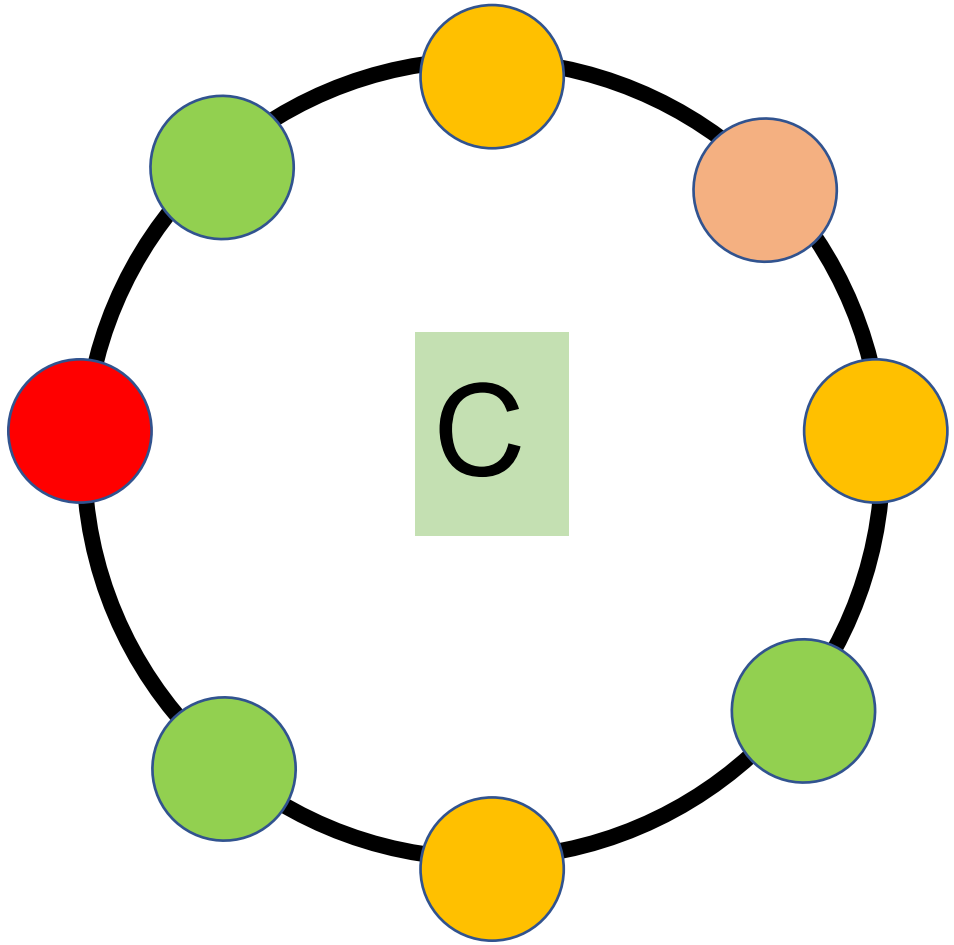
```c
// Simple C program
// main function
// where the execution begins
int main ()
{
    // prints hello world
    printf("Hello World");
    return 0;
}
```

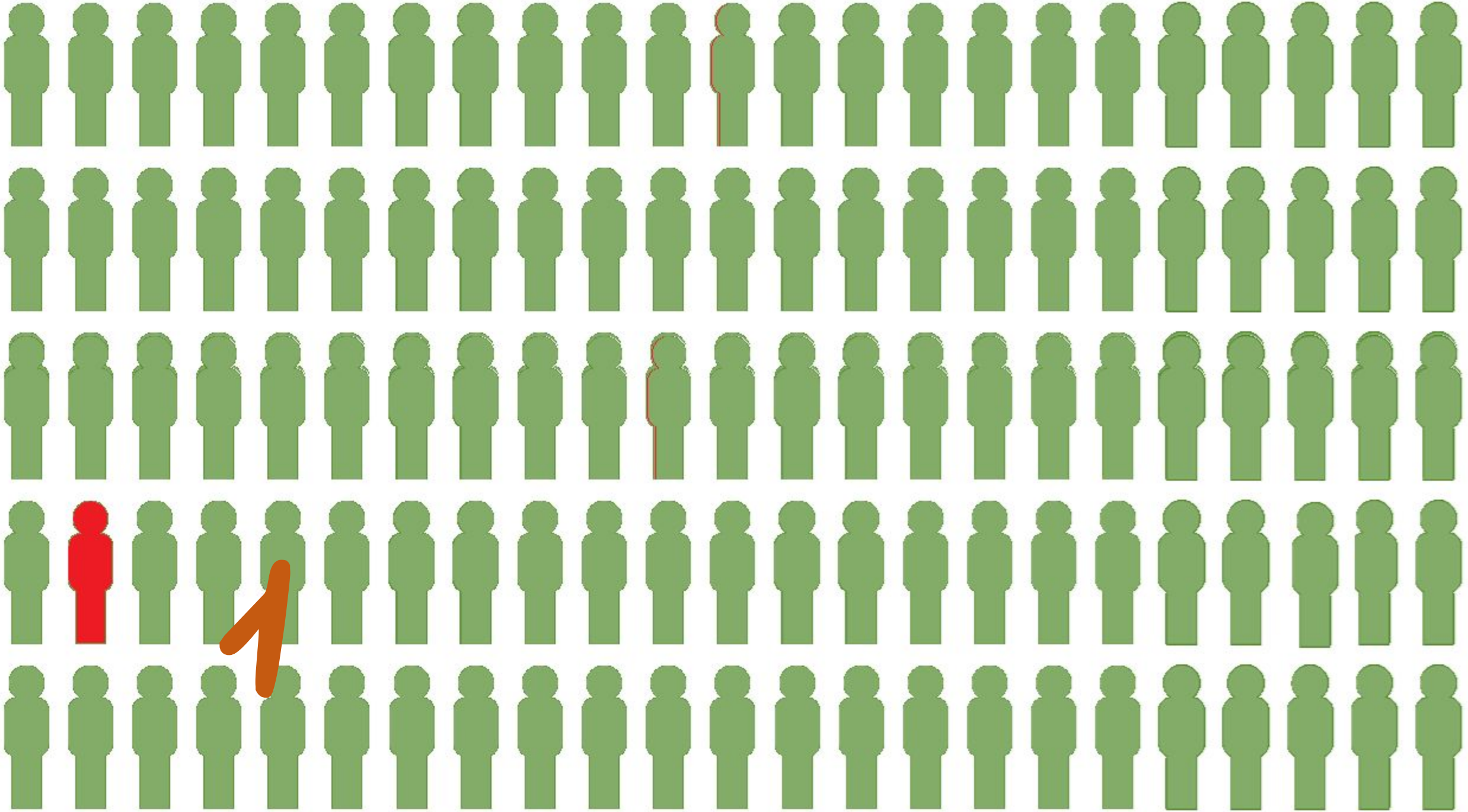
https://bookletsapp/booklet/two-types-of-programmers

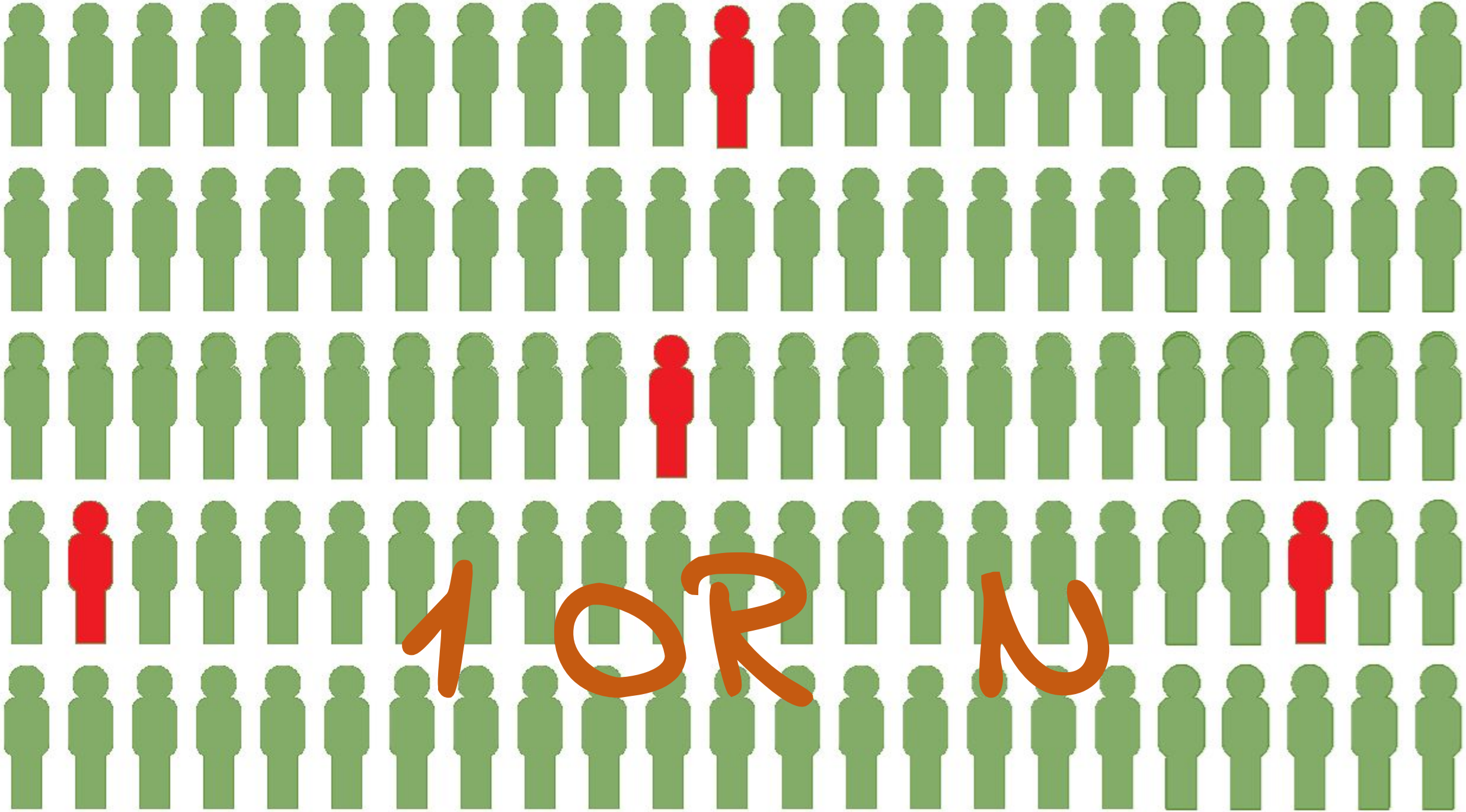
TWO TYPES OF ORGANIZATIONS

PRODUCT vs CUSTOMER






1


1 OR N

Across SDLC

- Requirements
- Design
- Development Stage
- Software Testing Stage
- Implementation & Integration
- Operations and Maintenance

P

C

- ~~Requirements~~
- Design
- Development Stage
- Software Testing Stage
- Implementation & Integration
- Operations and Maintenance

P

- Base Requirements – Requirements Prioritizing, Cost of Requirements
- Listening – Exercises
- Run twice as fast – to be in the same place

C

- Different Requirements
- Re-learn, Adapt
- Maintaining Abstractions and Uniqueness

- Requirements
- Design
- Development Stage
- Software Testing Stage
- Implementation & Integration
- Operations and Maintenance

P

- Flexible Design – Technology Driven
- Interoperability
- Standards and Protocols

C

- High level of Abstraction Variable Configuration
- Flexible Design Functionality Driven
- Compliance to Enterprise Architecture

- Requirements
- Design
- ~~Development Stage~~
- Software Testing Stage
- Implementation & Integration
- Operations and Maintenance

P

- Focus on DevOps – High
- Market Sensitive Releases
- Security Sensitivity - High

C

- Stable Design – Enterprise Roadmap
- Interface Adaptability
- Function vs Feature

- Requirements
- Design
- Development Stage
- ~~Software Testing Stage~~
- Implementation & Integration
- Operations and Maintenance

P

- Regression Testing – High
- Automated Testing

C

- Functional Testing Driven
- Sociability Testing

- Requirements
- Design
- Development Stage
- Software Testing Stage
- **Implementation & Integration**
- Operations and Maintenance

P

- User Choices
- Interoperability

C

- EA driven Solutions
- Longer Life

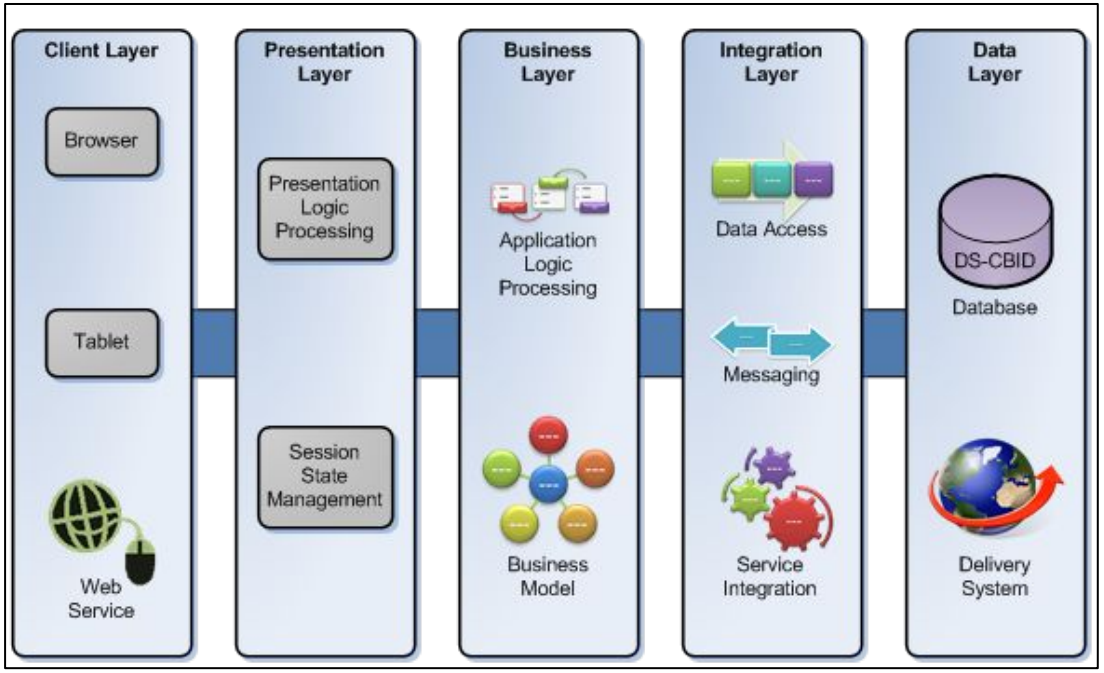
- Requirements
- Design
- Development Stage
- Software Testing Stage
- Implementation & Integration
- Operations and Maintenance

P

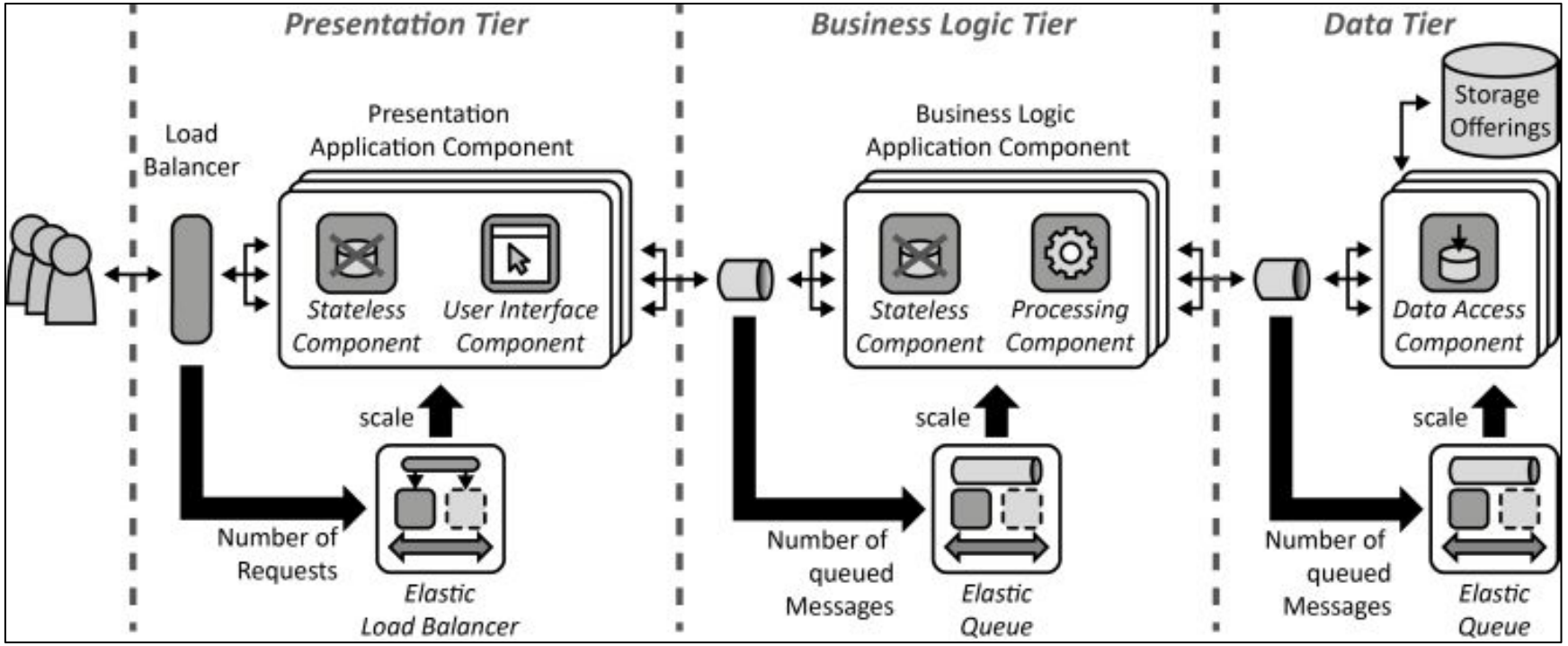
- Low Focus – Managed by New Releases

C

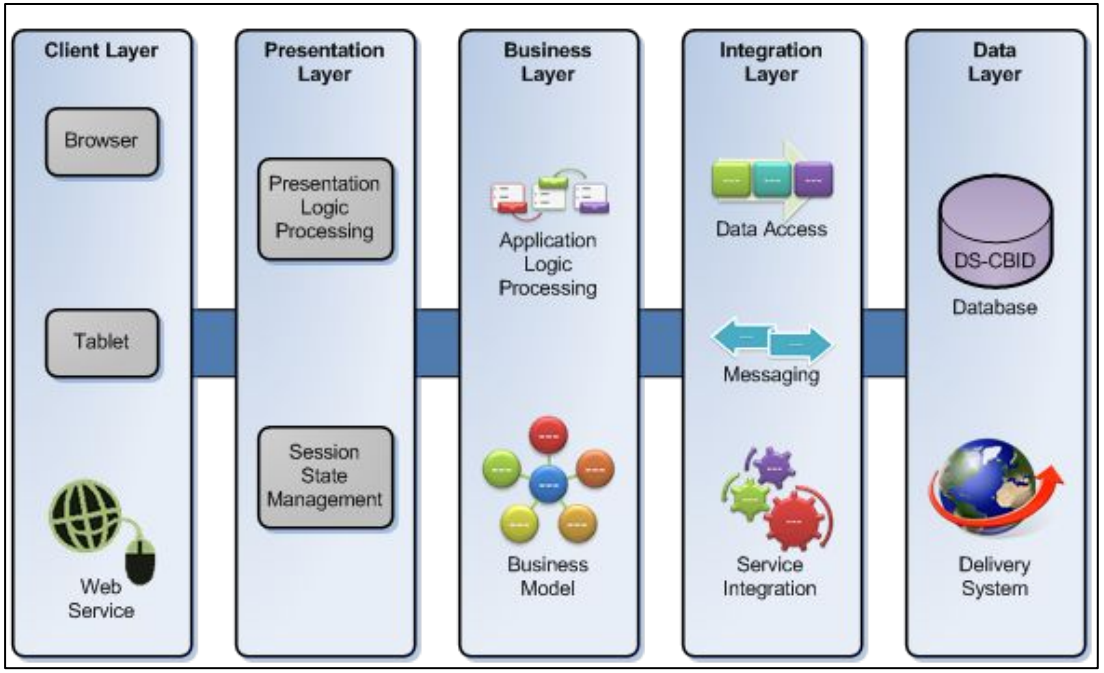
- Change Driven Releases
- Deferred Strategy for Functionality



https://www.ajboggs.com/our-experiences/health-information-technology/sisonline-supports-assessment/sisonline-system-architecture/



https://www.cloudcomputingpatterns.org/three_tier_cloud_application/

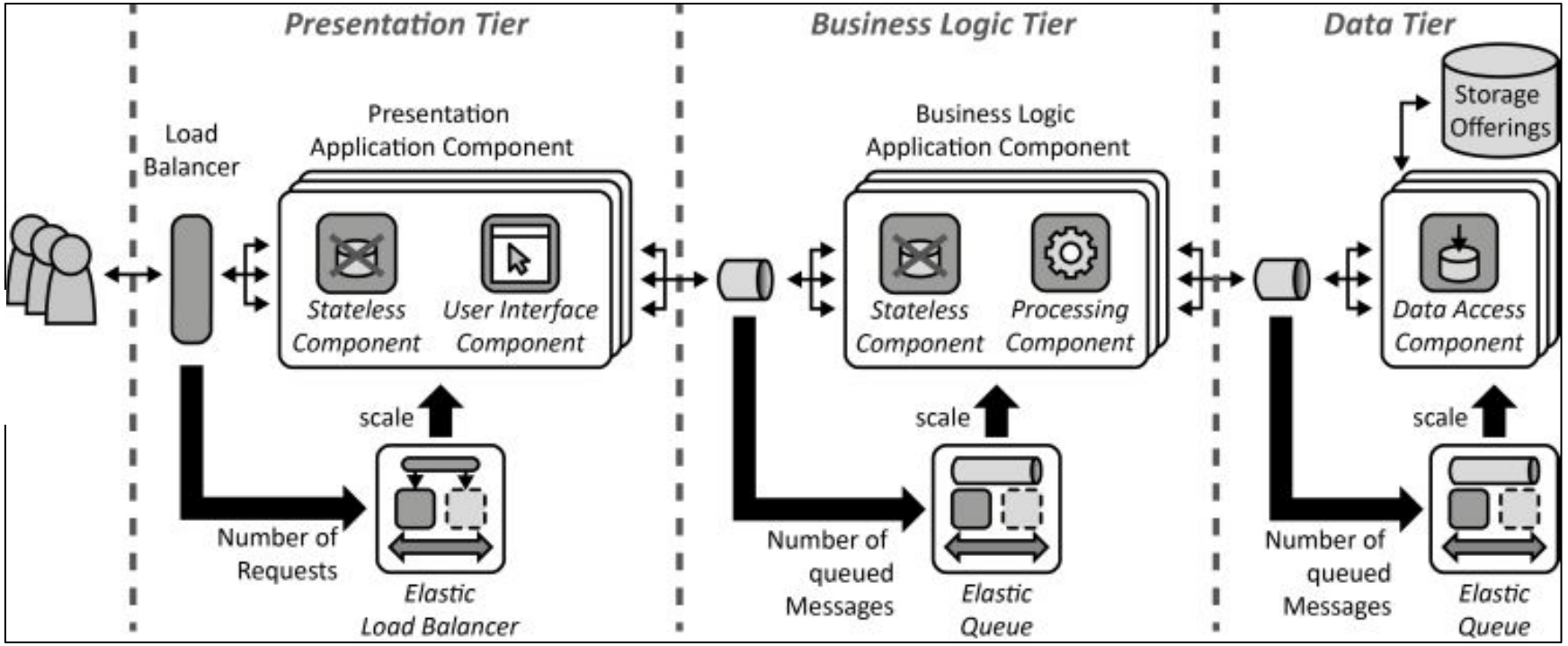


https://www.ajboggs.com/our-experiences/health-information-technology/sisonline-supports-assessment/sisonline-system-architecture/

n-Tier

3-Tier

https://www.cloudcomputingpatterns.org/three_tier_cloud_application/



NU

Requirements
Customer
Understanding

PROCESS

SDLC

Big Data
Engineering

Interfaces / Feeds

DATA

Data Science

SECURITY

Cybersecurity

Observe

Observe

Lear
n

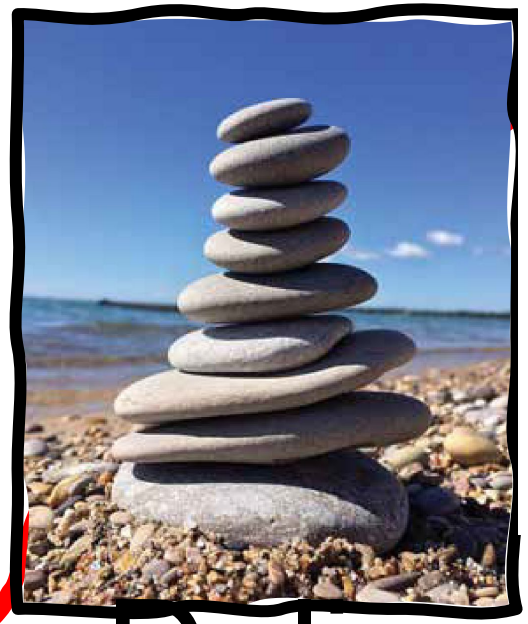
Observe

Lear
n

Articulate

Observe

Lear
n



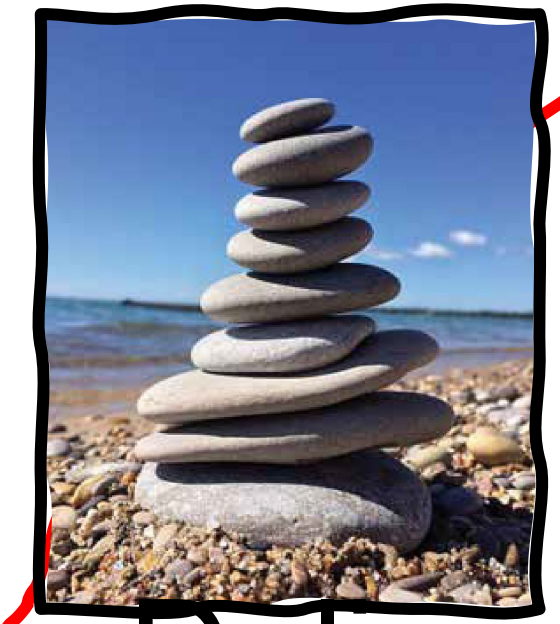
Delive
r

Articulate

Observe

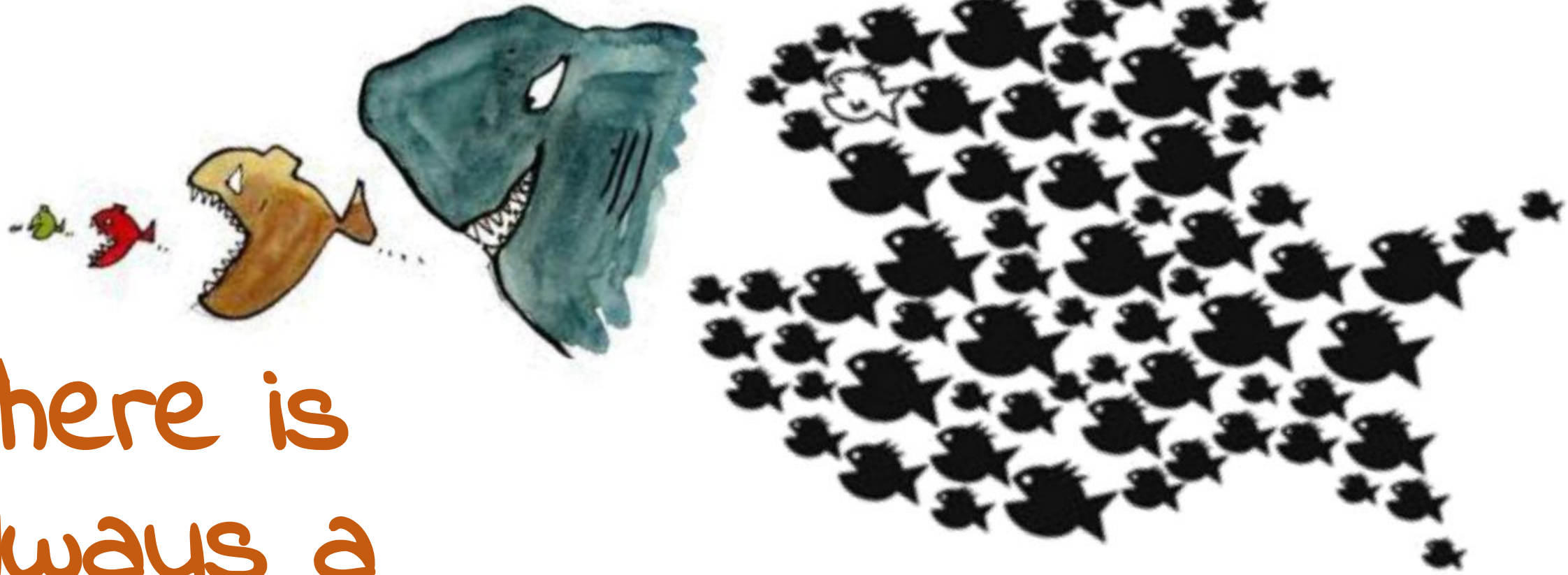
Lear
n

Articulate



Delive
r

Celebrat
e



There is always a bigger fish...