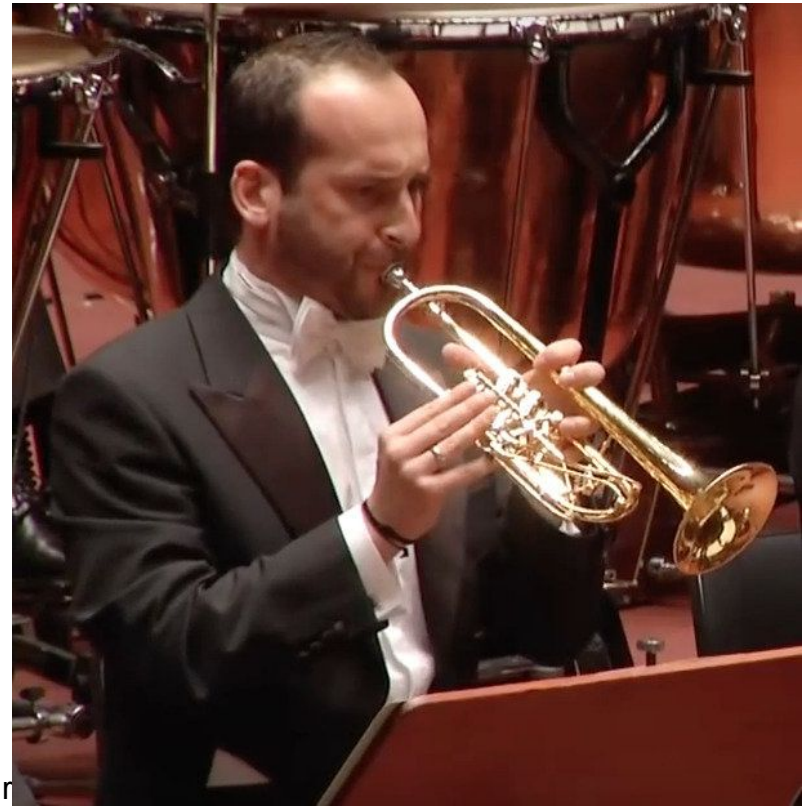
Notice The Trumpeter

Ran Regev @regev_ran regev.ran@gmail.com

The Trumpeter



MAINTENANCE

MAINTENANCE

The Force We Must Listen To

Ran Regev

Ran Regev

Ran Regev

SOFTWARE DEVELOPER
20+ YEARS

Ran Regev @regev_ran regev.ran@gmail.com

Ran Regev

SOFTWARE DEVELOPER 20+ YEARS

ISO WG21, C++, ISRAEL NATIONAL BODY

Ran Regev

SOFTWARE DEVELOPER 20+ YEARS

ISO WG21, C++, ISRAEL NATIONAL BODY

MAINTENANCE

MAINTENANCE

noun

1. the process of maintaining or preserving someone or something, or the state of being maintained.
   "crucial conditions for the maintenance of democratic government"

   Similar: preservation | conservation | continuation | continuance | continuity

2. the provision of financial support for a person's living expenses, or the support so provided

MAINTENANCE

noun

1. the process of maintaining or preserving someone or something, or the state of being maintained.
   "crucial conditions for the maintenance of democratic government"

   Similar: preservation | conservation | continuation | continuance | continuity

2. the provision of financial support for a person's living expenses, or the support so provided

Ran Regev @regev_ran regev.ran@gmail.com

MAINTENANCE

noun

1. the process of maintaining or preserving someone or something, or the state of being maintained.
   "crucial conditions for the maintenance of democratic government"

   Similar: preservation | conservation | continuation | continuance | continuity

2. the provision of financial support for a person's living expenses, or the support so provided

MAINTENANCE

noun

1. the process of maintaining or preserving someone or something, or the state of being maintained.
   "crucial conditions for the maintenance of democratic government"

   Similar: preservation | conservation | continuation | continuance | continuity

2. the provision of financial support for a person's living expenses, or the support so provided

MAINTENANCE

noun

1. the process of maintaining or preserving someone or something, or the state of being maintained.
   "crucial conditions for the maintenance of democratic government"

   Similar: preservation | conservation | continuation | continuance | continuity

2. the provision of financial support for a person's living expenses, or the support so provided

QUALITY

QUALITY

Readable

QUALITY

Readable

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

QUALITY

Readable

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable

QUALITY

Readable

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += i;
}
```

QUALITY

Readable

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable **Understandable**

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable **Understandable**

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable **Understandable**

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable **Confusing**

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

Ran Regev @regev_ran regev.ran@gmail.com

QUALITY

Readable **Understandable**

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable **Confusing**

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable **Understandable** **Bright**

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable **Confusing**

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

Ran Regev @regev_ran regev.ran@gmail.com

QUALITY

Readable **Understandable** **Bright**

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable **Confusing**

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable **Understandable** **Bright**

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable **Confusing** **Opaque**

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable **Understandable** **Bright**

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable **Confusing** **Opaque**

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable **Understandable** **Bright**

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable **Confusing** **Opaque**

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable Understandable Bright Exposes Layers

```
int sum = std::accumulate(v.begin(), v.end(), 7);
```

Unreadable Confusing Opaque

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable Understandable Bright Exposes Layers

```
int sum = std::accumulate(v.begin(), v.end(), f());
```

Unreadable Confusing Opaque

```
int sum = 7;
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

Ran Regev @regev_ran regev.ran@gmail.com

QUALITY

Readable Understandable Bright Exposes Layers

```
int sum = std::accumulate(v.begin(), v.end(), f());
```

Unreadable Confusing Opaque Surface

```
int sum = f();
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable Understandable Bright Exposes Layers

Snap!

```
int sum = std::accumulate(v.begin(), v.end(), f());
```

Unreadable Confusing Opaque Surface

```
int sum = f();
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

QUALITY

Readable Understandable Bright Exposes Layers Snap!

```
int sum = std::accumulate(v.begin(), v.end(), f());
```

Unreadable Confusing Opaque Surface Heavy

```
int sum = f();
for ( int i = 0; i < v.size(); ++ i )
{
    sum += v[i];
}
```

DEVELOPMENT FORCES

Ran Regev @regev_ran regev.ran@gmail.com

DEVELOPMENT FORCES

Time To Market

DEVELOPMENT FORCES

Time To Market

Performance

DEVELOPMENT FORCES

Time To Market

Performance

```
if ( pa )
{
    if ( pa->f() )
    {
        // more code
    }
}
```

DEVELOPMENT FORCES

Time To Market

Performance

```
if ( pa && pa->f() )
{
    // more code
}
```

DEVELOPMENT FORCES

Time To Market

Performance

```
memset()
vs.
std::fill()
```

DEVELOPMENT FORCES

Time To Market

Performance

DEVELOPMENT FORCES

Time To Market

Performance

DEVELOPMENT FORCES

Time To Market

Performance

Ran Regev @regev_ran regev.ran@gmail.com

DEVELOPMENT FORCES

Time To Market

Performance

Ran Regev @regev_ran regev.ran@gmail.com

DEVELOPMENT FORCES

Time To Market

Performance

DEVELOPMENT FORCES

Time To Market

Performance

DEVELOPMENT FORCES

Time To Market

Performance

DEVELOPMENT FORCES

Time To Market

Performance

DEVELOPMENT FORCES

Time To Market

Performance

Correctness

DEVELOPMENT FORCES

Time To Market

Performance

Correctness

DEVELOPMENT FORCES

Time To Market

Performance

Correctness

DEVELOPMENT FORCES

Time To Market

Performance

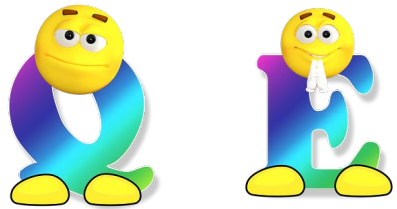
Correctness

DEVELOPMENT FORCES

Time To Market

Performance

Correctness



DEVELOPMENT FORCES

Time To Market

Performance

Correctness

DEVELOPMENT FORCES

Time To Market

Performance

Correctness

DEVELOPMENT FORCES

Time To Market

Performance

Correctness

Maintenance?

DEVELOPMENT FORCES

Time To Market

Performance

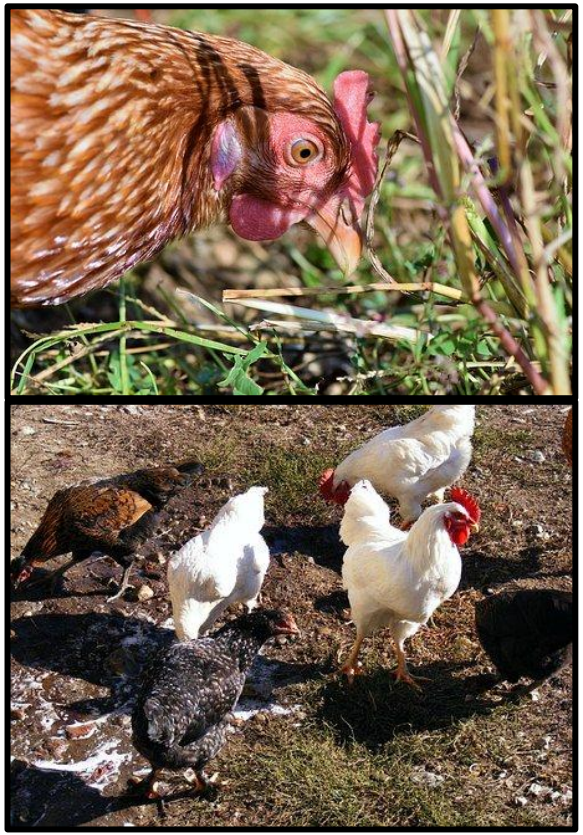
Correctness

Maintenance?

UNMAINTAINED OUTCOME

UNMAINTAINED OUTCOME

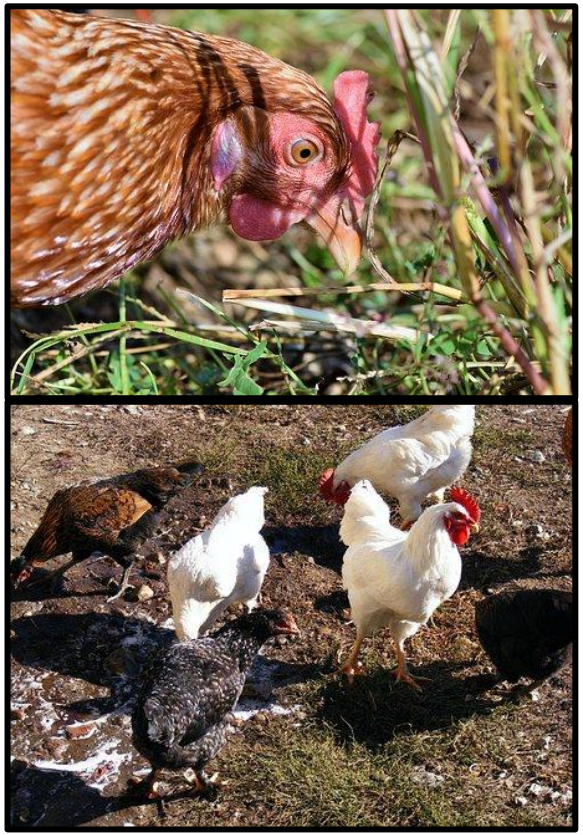
UNMAINTAINED OUTCOME



Ran Regev @regev_ran regev.ran@gmail.com

UNMAINTAINED OUTCOME

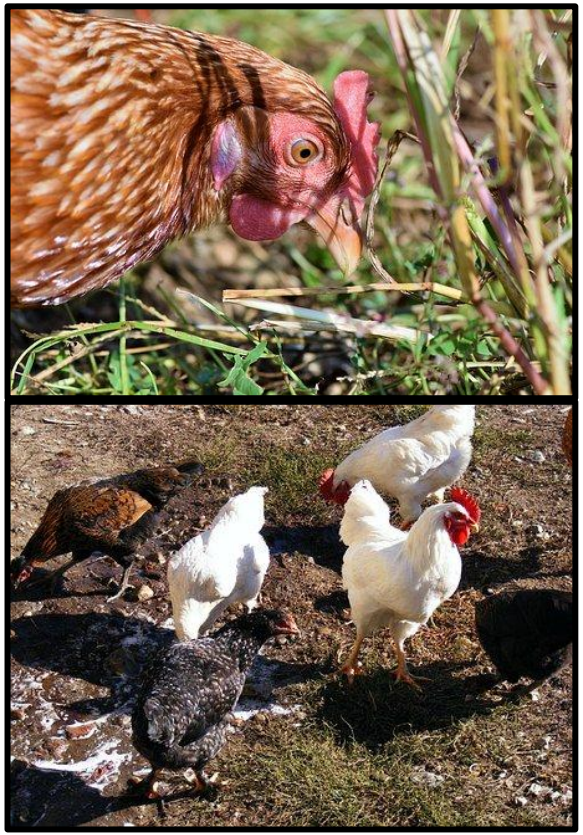
Long Development Times



UNMAINTAINED OUTCOME

Long Development Times

Huge Learning Curves

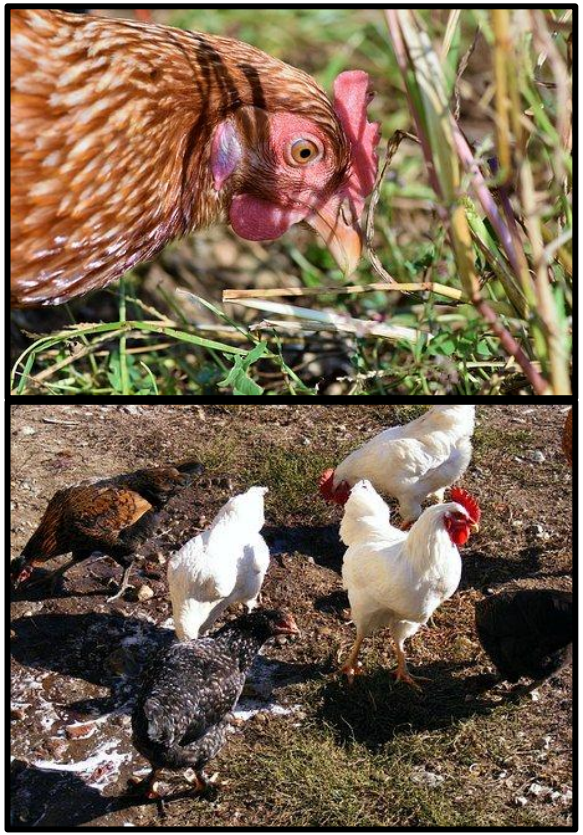


UNMAINTAINED OUTCOME

Long Development Times

Huge Learning Curves

Untouchable Areas



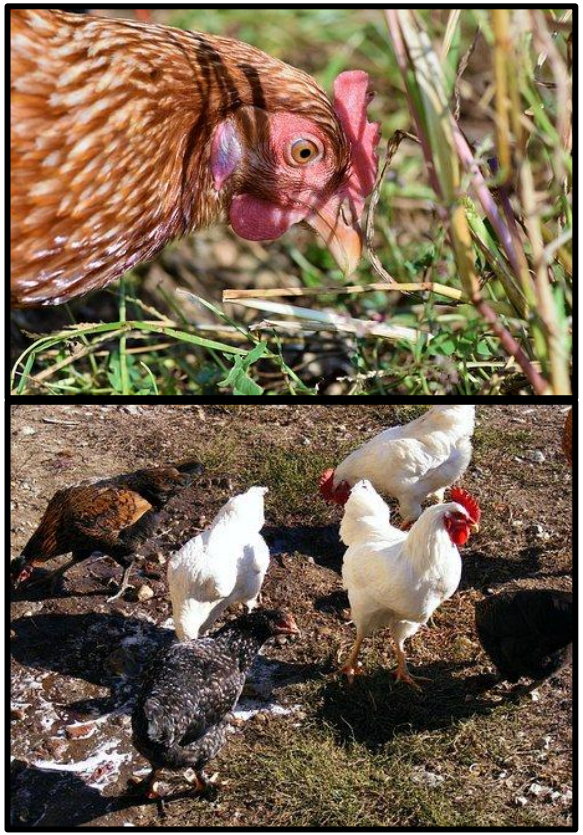
Ran Regev @regev_ran regev.ran@gmail.com

UNMAINTAINED OUTCOME

Long Development Times

Huge Learning Curves

Untouchable Areas



UNMAINTAINED OUTCOME

UNMAINTAINED OUTCOME

Frustration

UNMAINTAINED OUTCOME

Frustration

UNMAINTAINED OUTCOME

Frustration

UNMAINTAINED OUTCOME

Frustration

TRE17

UNMAINTAINED OUTCOME

Frustration

TRE17

UNMAINTAINED OUTCOME

Frustration

Slow Entrance

TRE17

UNMAINTAINED OUTCOME

Frustration

Slow Entrance

TRE17

UNMAINTAINED OUTCOME

Frustration

Slow Entrance

Long Breaks

TRE17

UNMAINTAINED OUTCOME

Frustration

Slow Entrance

Long Breaks

TRE17

UNMAINTAINED OUTCOME

Frustration

Slow Entrance

Long Breaks

Disruptions

TRE17

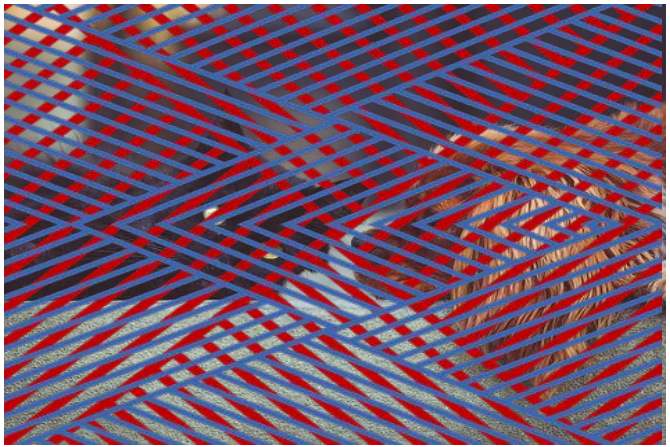
UNMAINTAINED OUTCOME

Frustration

Slow Entrance

Long Breaks

Disruptions



TRE17

UNMAINTAINED OUTCOME

Frustration

Slow Entrance

Long Breaks

Disruptions

TRE17

UNMAINTAINED OUTCOME

Frustration

Slow Entrance

Long Breaks

Disruptions

TRE17

UNMAINTAINED OUTCOME

Frustration

Slow Entrance

Long Breaks

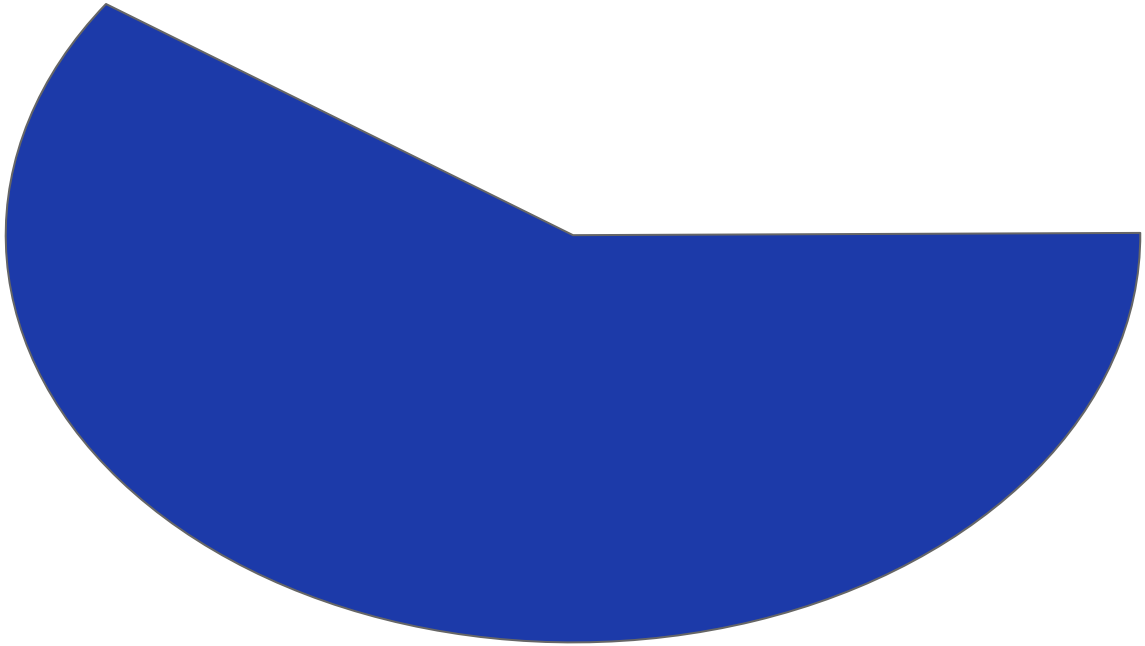
Disruptions

TRE17

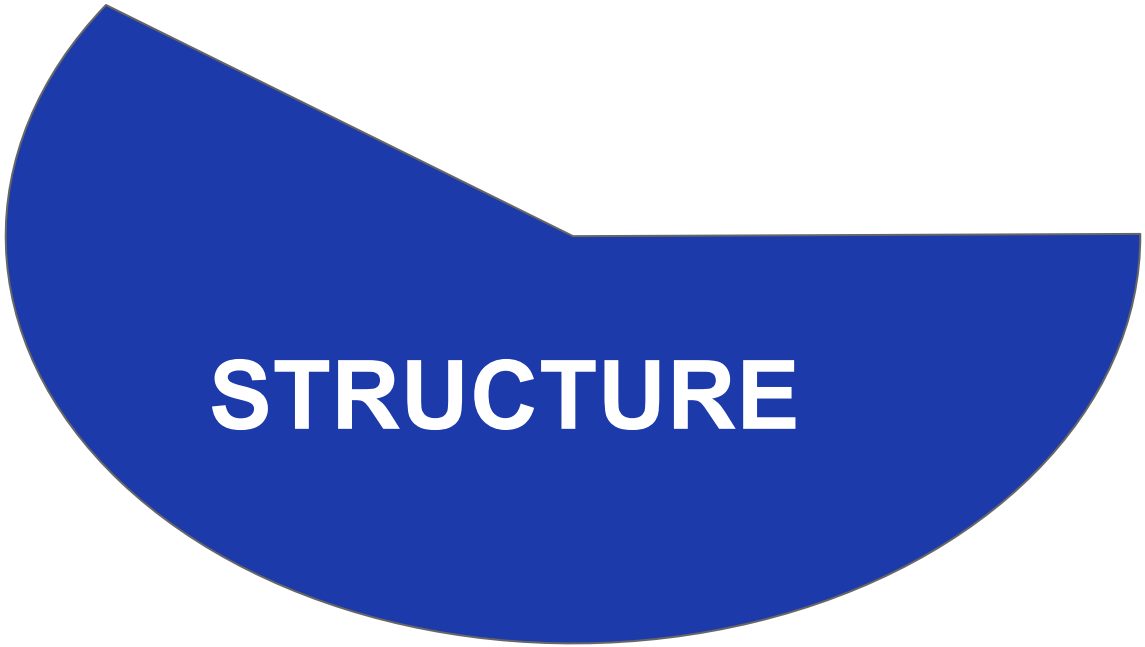
Ran Regev @regev_ran regev.ran@gmail.com

IMPACT

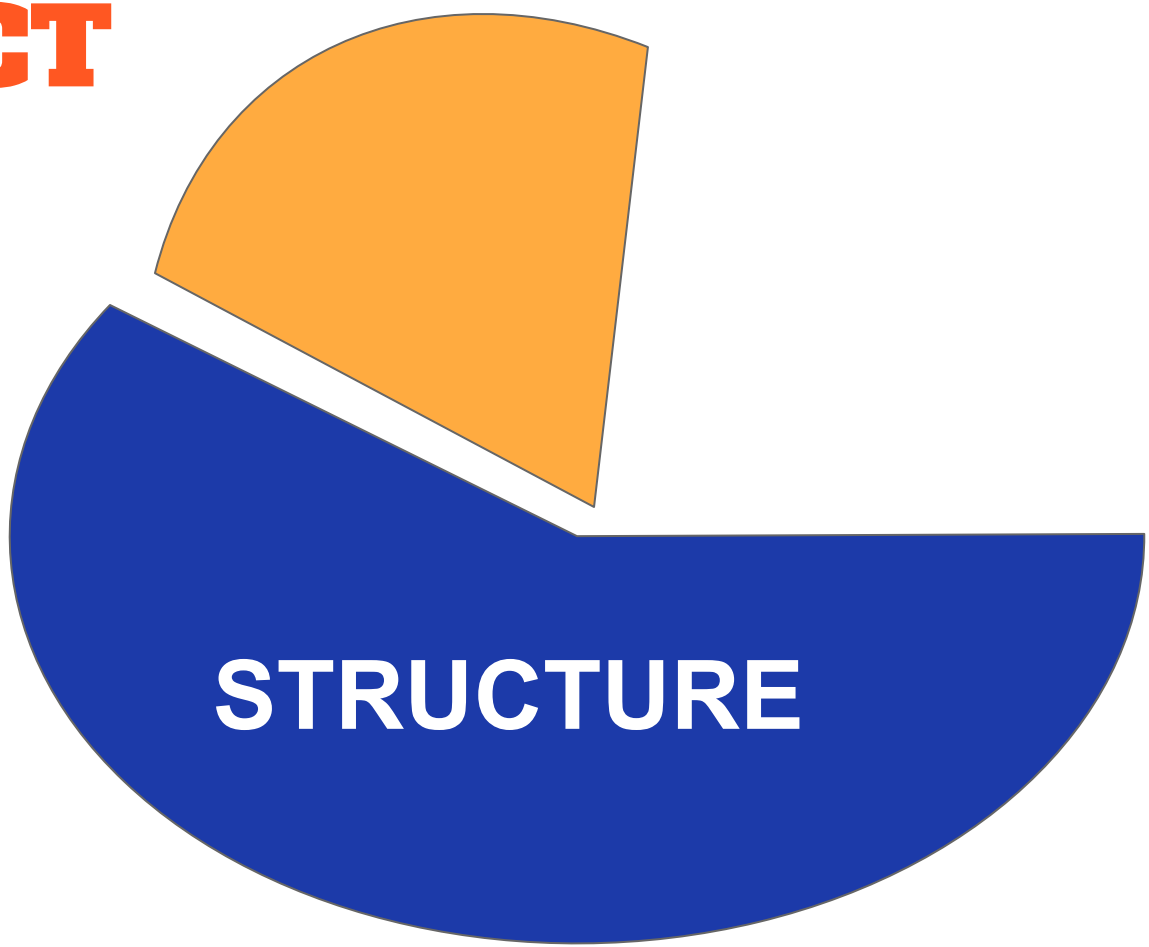
IMPACT



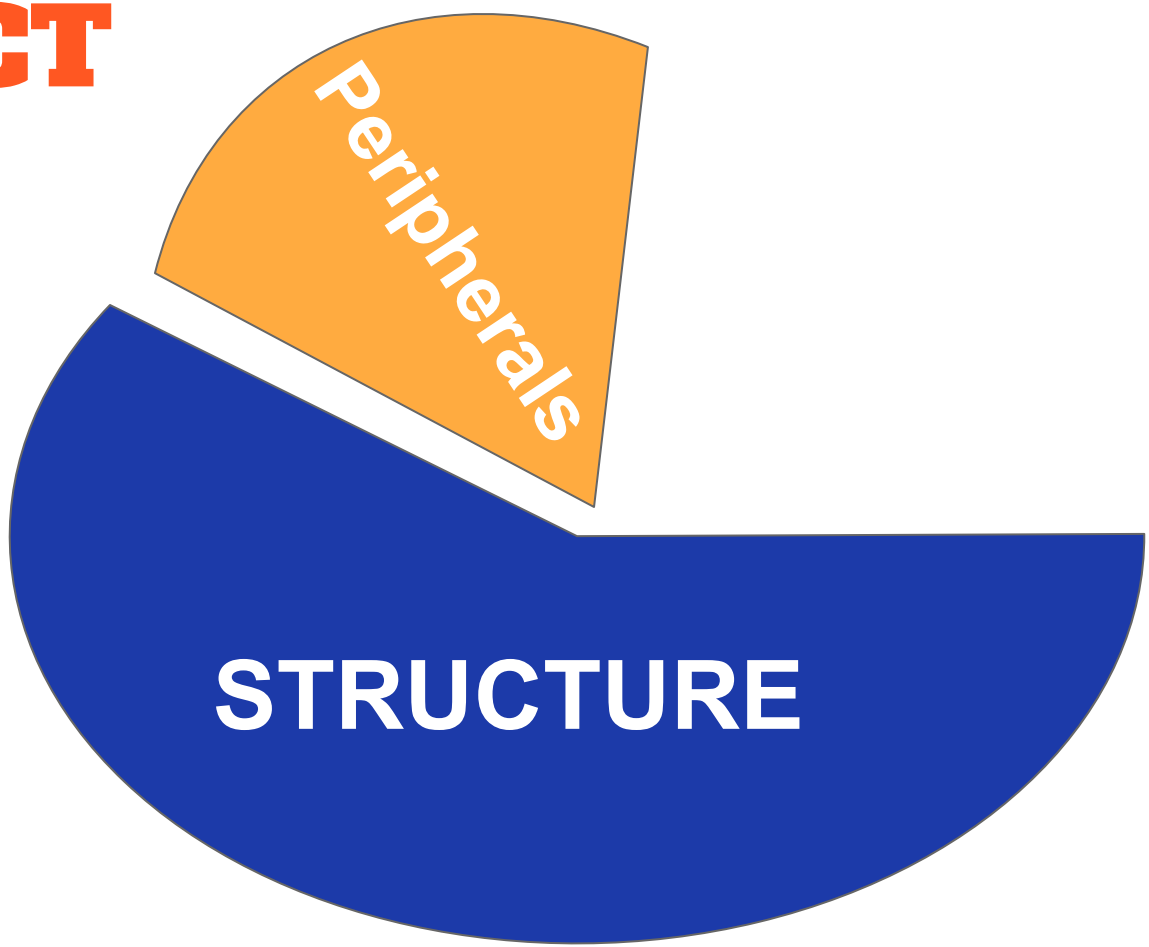
IMPACT



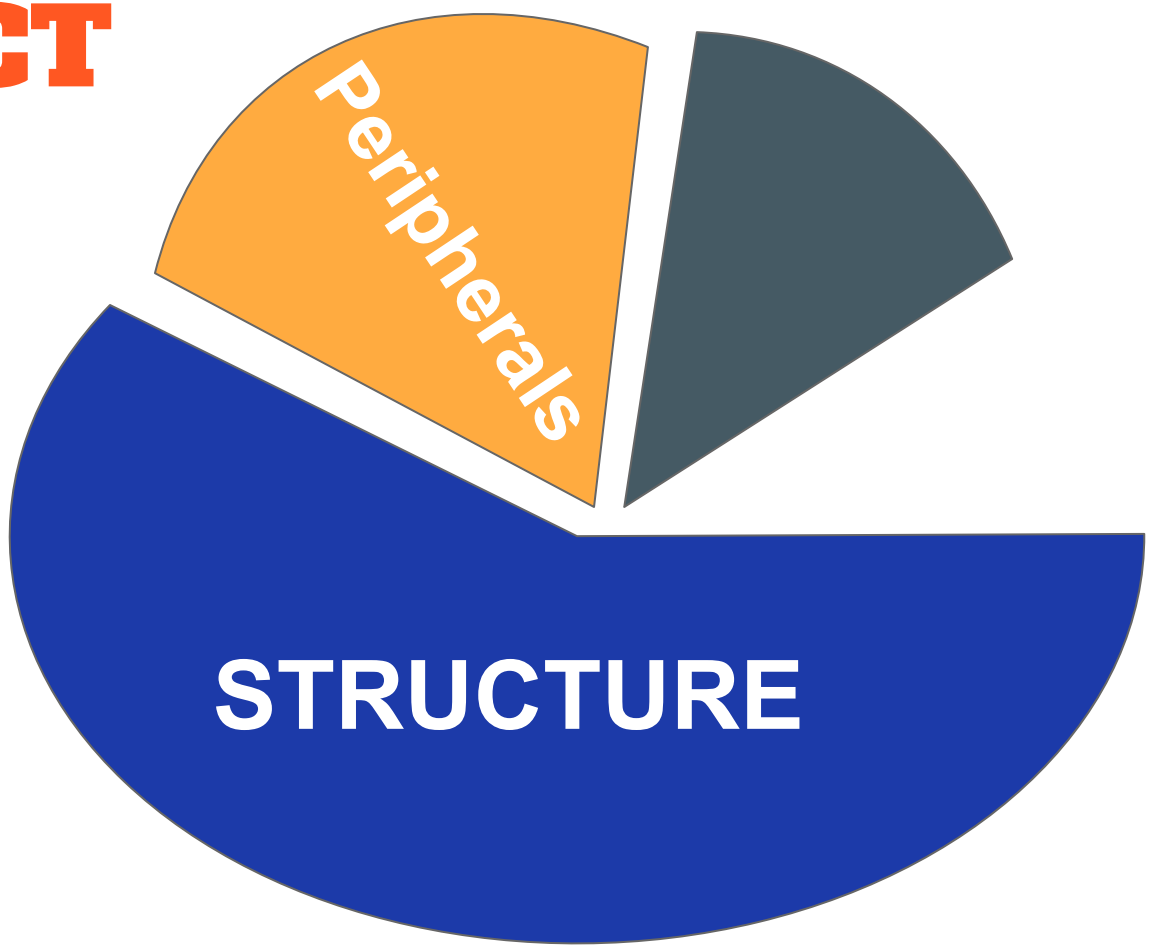
IMPACT



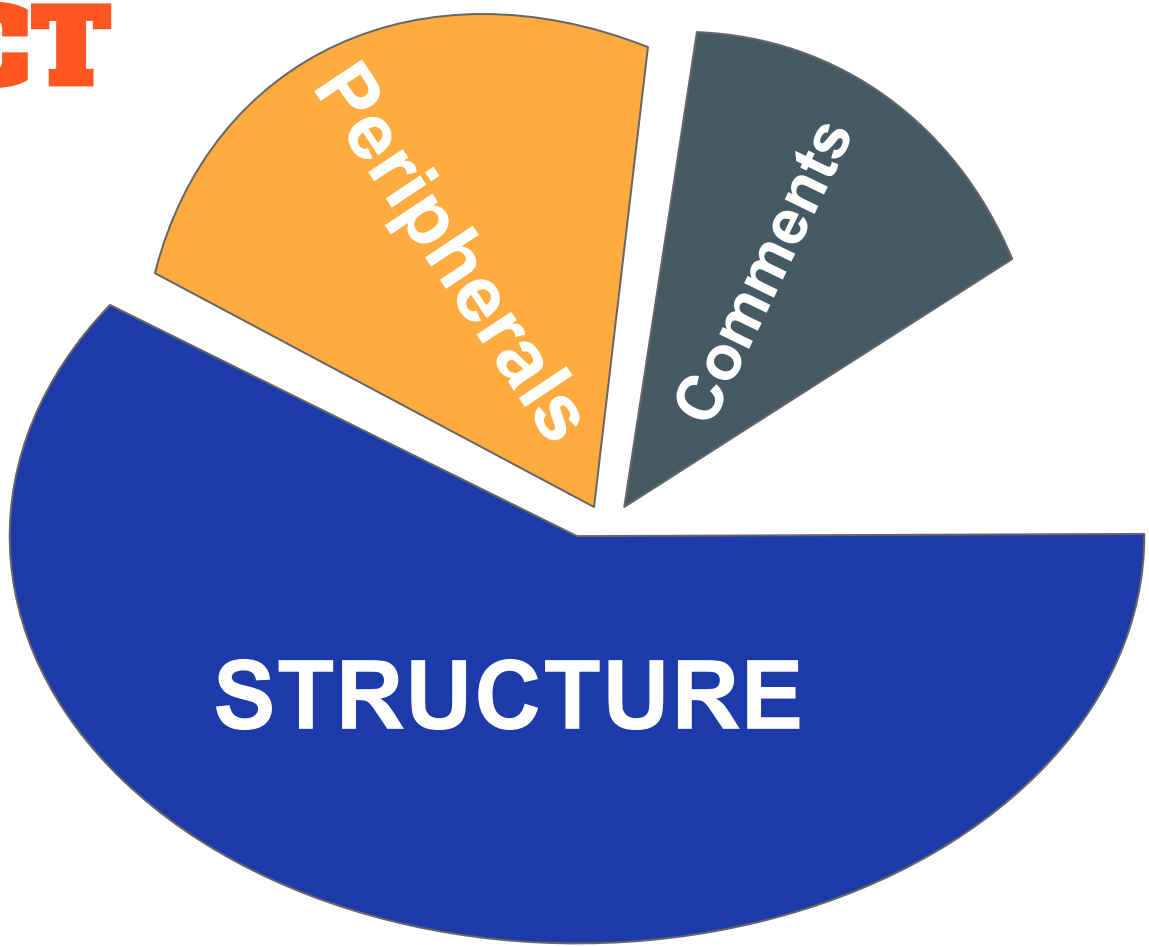
IMPACT



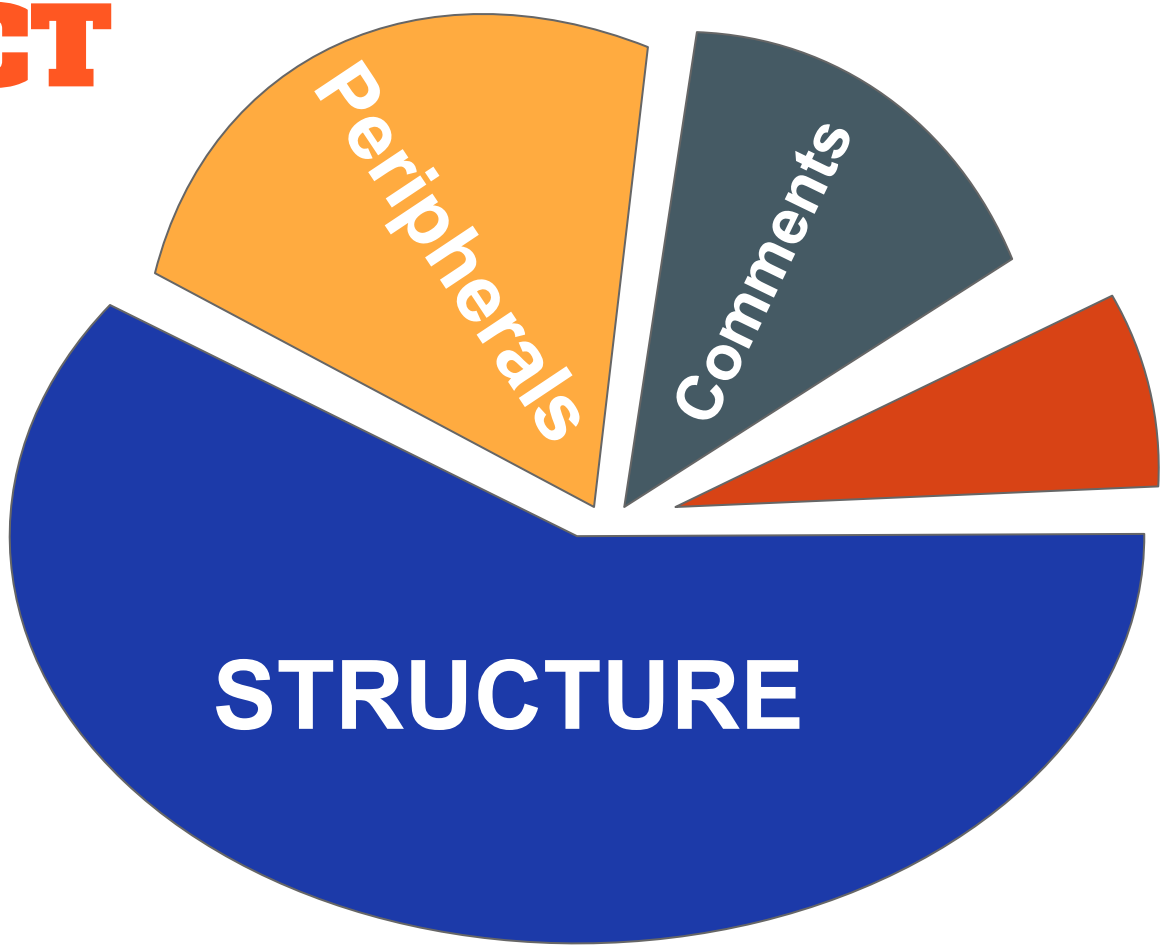
IMPACT



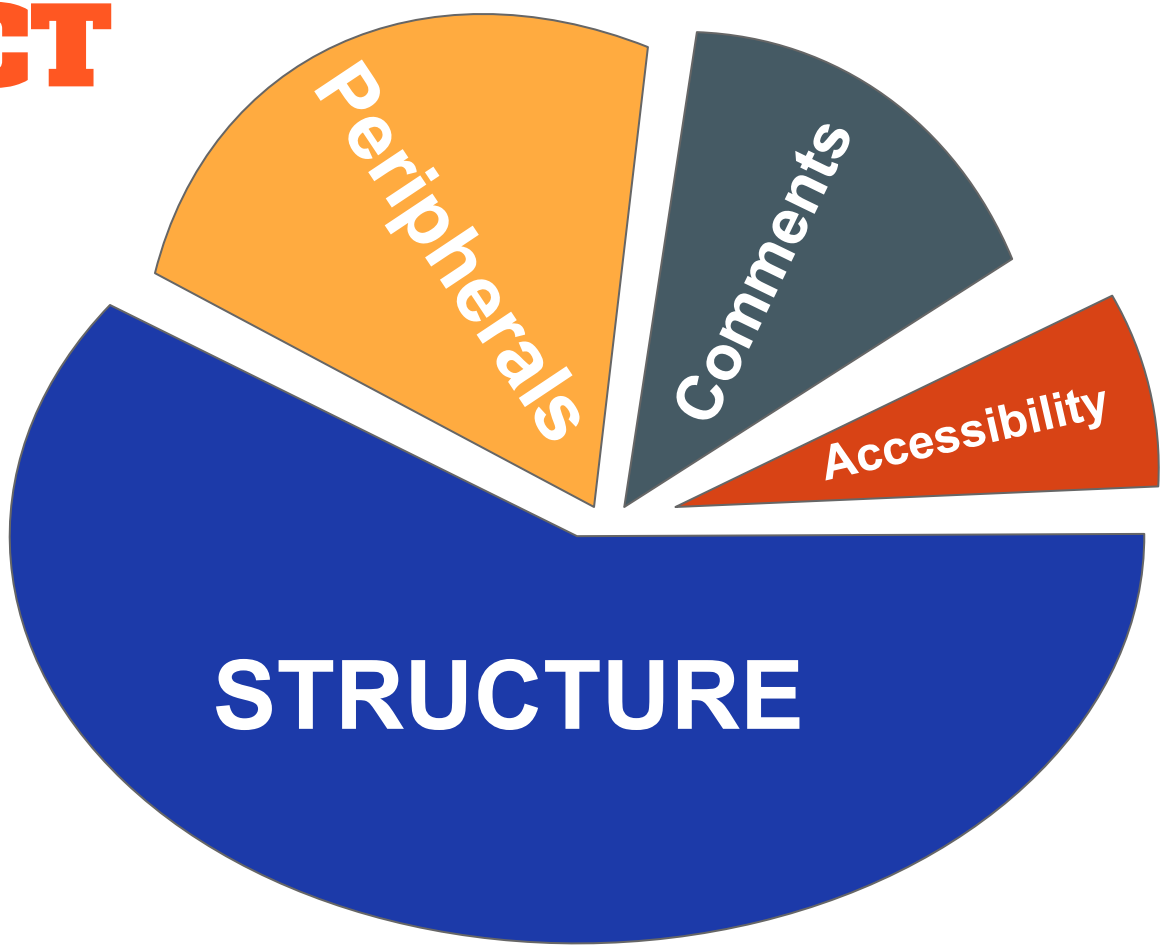
IMPACT



IMPACT



IMPACT



STRUCTURE

STRUCTURE

A

STRUCTURE

A B

STRUCTURE

A B C

STRUCTURE

A

B

C

STRUCTURE

A B C

STRUCTURE

A B C

STRUCTURE

STRUCTURE

STRUCTURE

STRUCTURE

STRUCTURE

Ran Regev @regev_ran regev.ran@gmail.com

STRUCTURE

```
using namespace vacation;
```

STRUCTURE

```
using namespace vacation;
```

STRUCTURE

```
using namespace vacation;
using namespace amazing;
```

STRUCTURE

```
using namespace vacation;
using namespace amazing;
```

STRUCTURE

```
using namespace vacation;
using namespace amazing;
using namespace wish;
```

STRUCTURE

```
using namespace vacation;
using namespace amazing;
using namespace wish;
```

STRUCTURE

```
using namespace vacation;
using namespace amazing;
using namespace wish;
using namespace home;
```

STRUCTURE

```
using namespace vacation;
using namespace amazing;
using namespace wish;
using namespace home;
```

STRUCTURE

STRUCTURE

STRUCTURE

Meetings

STRUCTURE

Meetings

STRUCTURE

Source Control

Meetings

STRUCTURE

Meetings

Source Control

STRUCTURE

Meetings

Source Control

File System

STRUCTURE

Source Control

Meetings

File System

Ran Regev @regev_ran regev.ran@gmail.com

STRUCTURE

Source Control

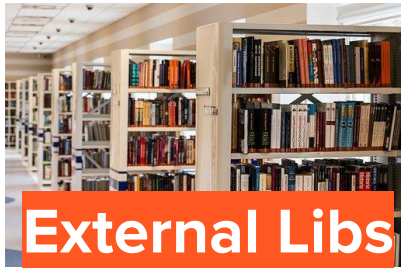
Meetings

File System

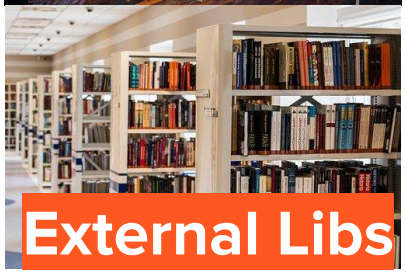
Maintained?

STRUCTURE File System Example

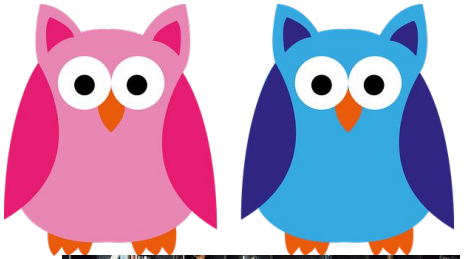
STRUCTURE File System Example

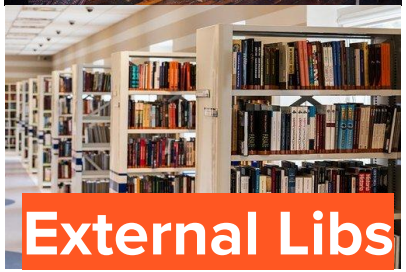


STRUCTURE File System Example

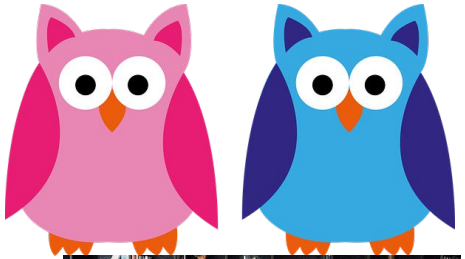


STRUCTURE File System Example

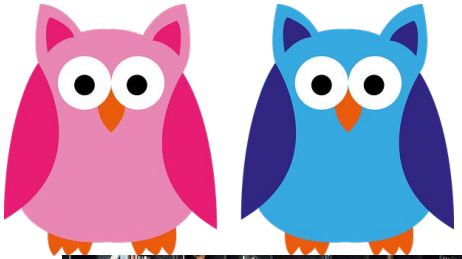




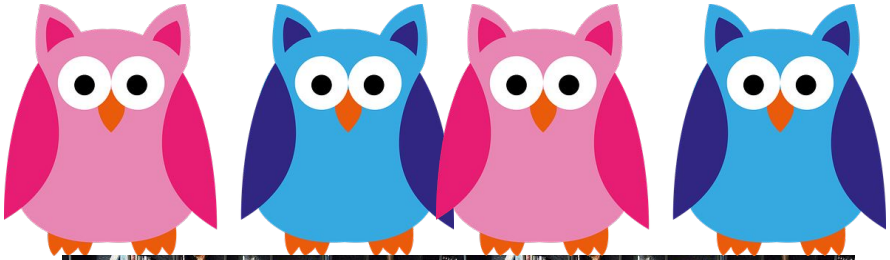
STRUCTURE File System Example



STRUCTURE File System Example

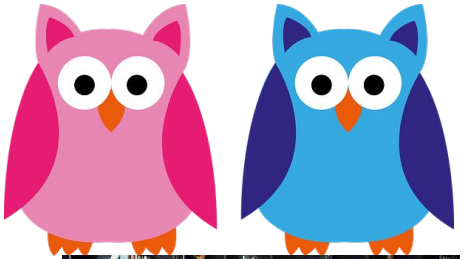


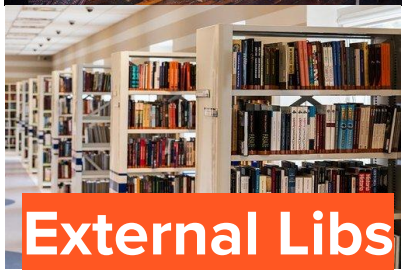
STRUCTURE File System Example

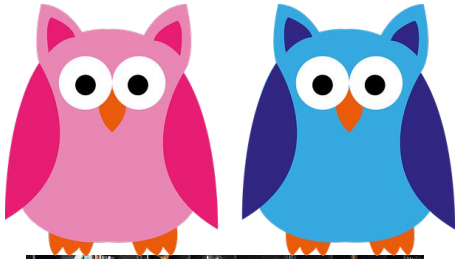


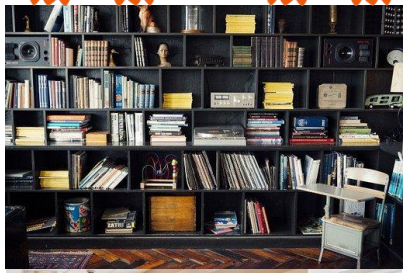
STRUCTURE

File System Example

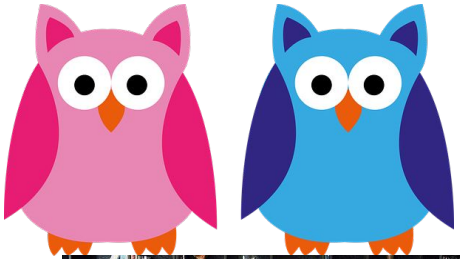


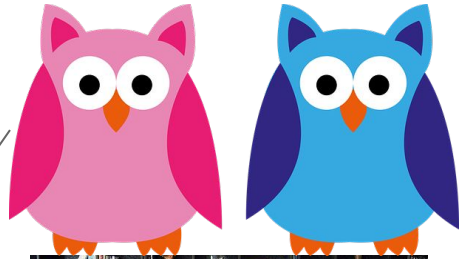


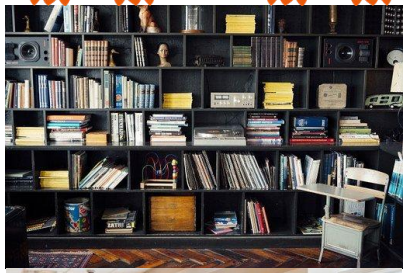




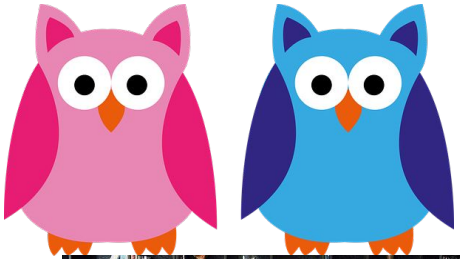
STRUCTURE File System Example

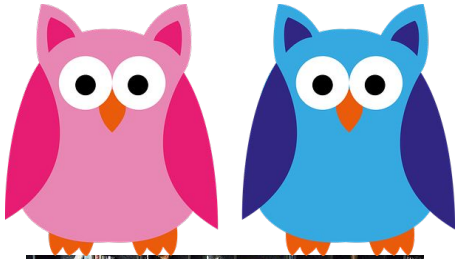


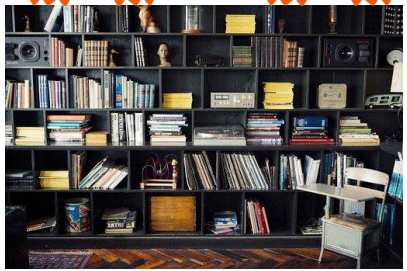


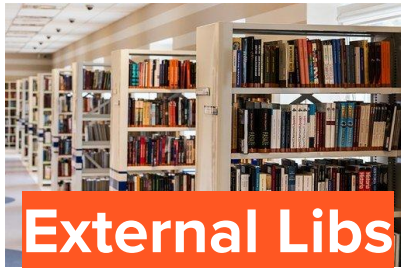


STRUCTURE File System Example

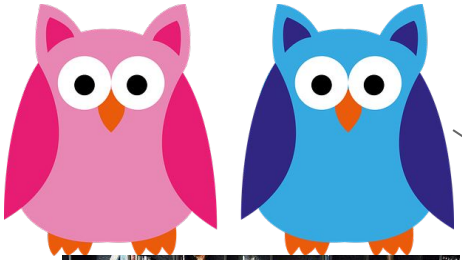


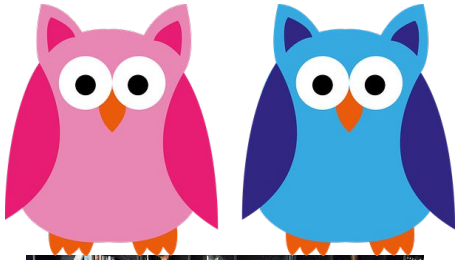


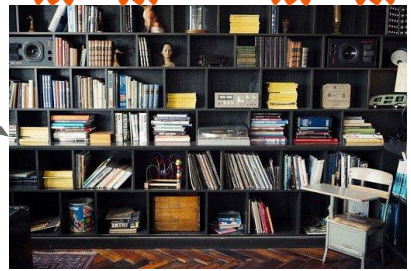


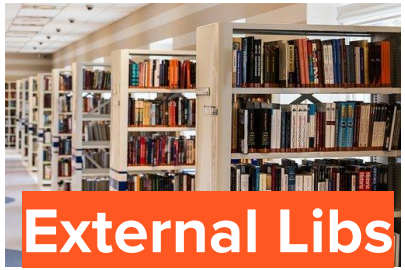


STRUCTURE File System Example

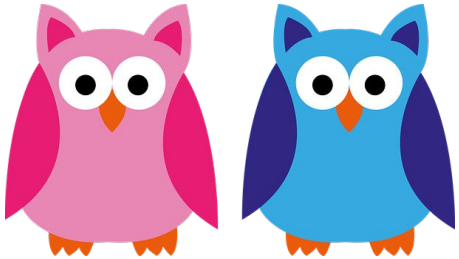




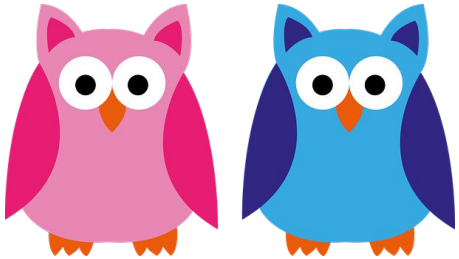


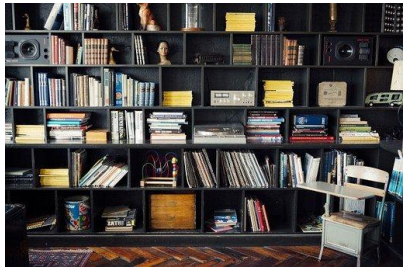


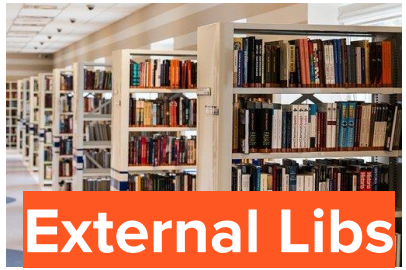
STRUCTURE



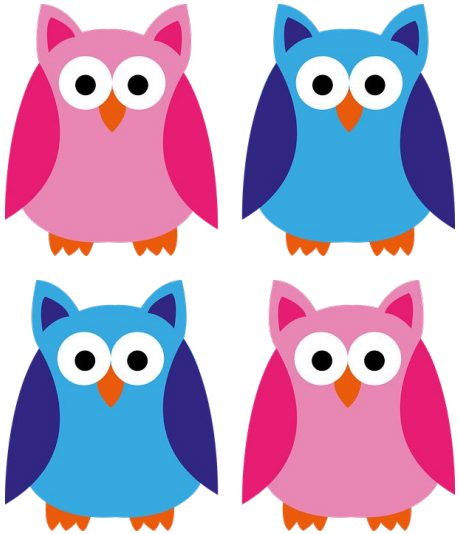
File System Example

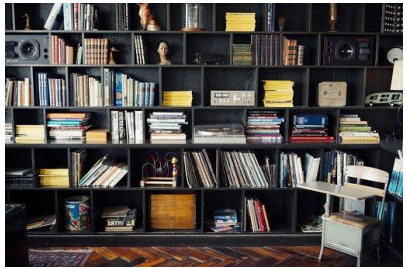


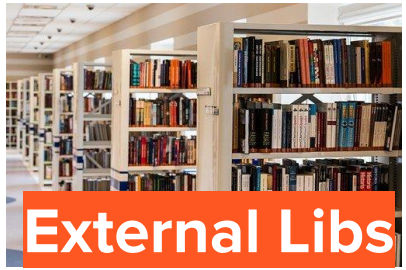




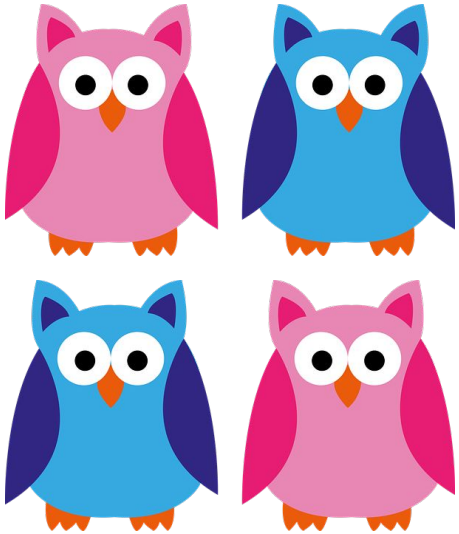
STRUCTURE File System Example



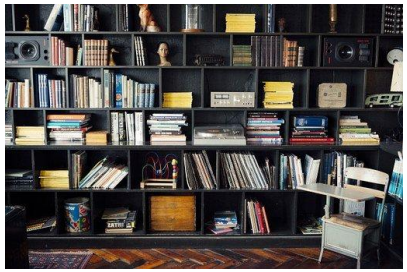


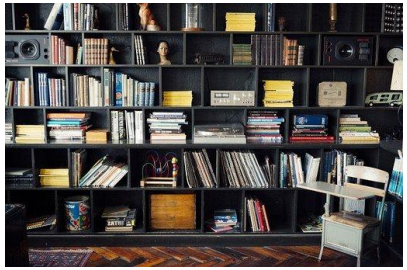


STRUCTURE

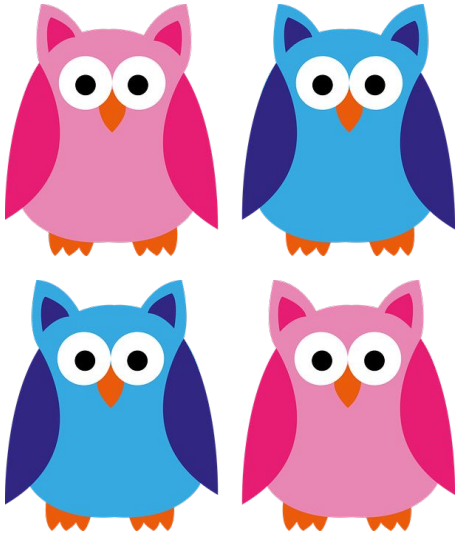


File System Example

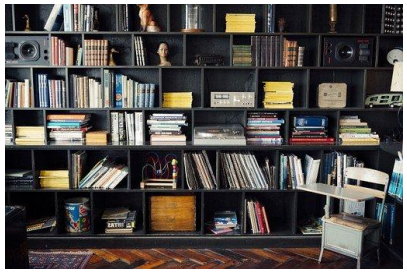


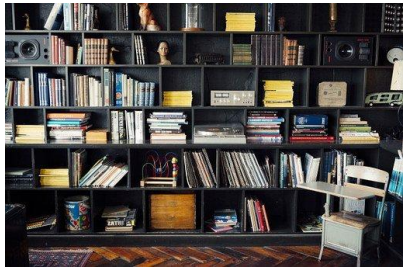


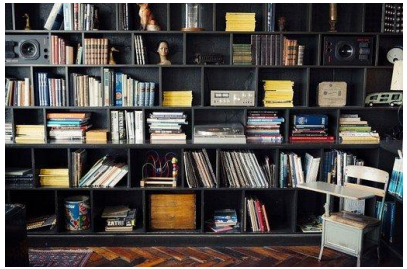
STRUCTURE



File System Example

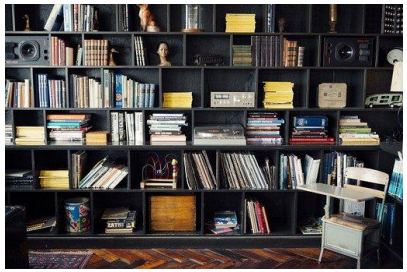


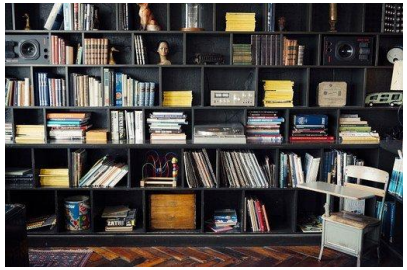


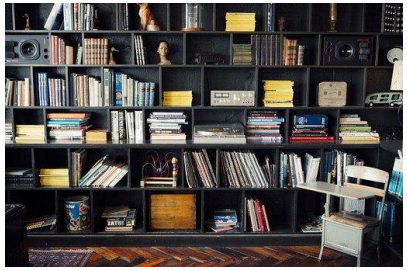


STRUCTURE

File System Example







Peripherals

Peripherals

Documents

The MainClass class

After initializing the program (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find the maximum parameters the user wants. After that we will build our DescribeUserIntentionClass that will call extendAbilities() for each user. Then we will call startProcessingInfo() (only if errorHandle() wasn't called)

The MainClass class

After initializing the pro (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find imum parameters the user wants. After that we will build our DescribeUserInte ss that will call extendAbilities() for each user. Then we will call startProce Handle() wasn't called)

Intent/Background

The MainClass class

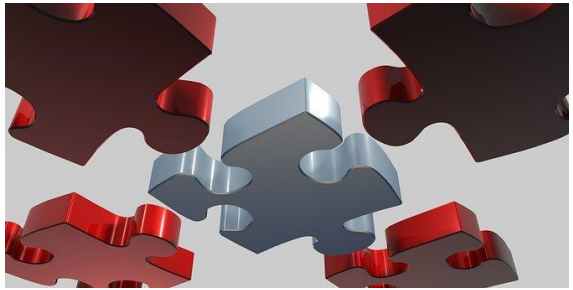
After initializing the pro (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find imum parameters the user wants. After that we will build our DescribeUserInte ss that will call extendAbilities() for each user. Then we will call startProce Handle() wasn't called)

Peripherals

Documents

Intent/Background

Related Docs



The MainClass class

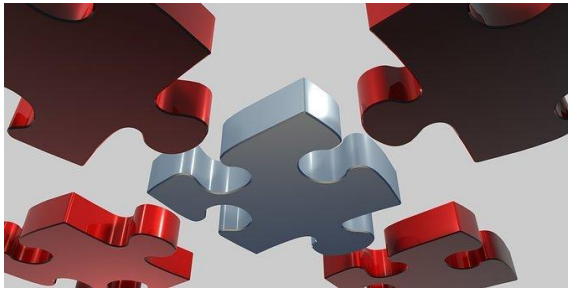
After initializing the pro (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find imum parameters the user wants. After that we will build our DescribeUserInte ss that will call extendAbilities() for each user. Then we will call startProce Handle() wasn't called)

Peripherals

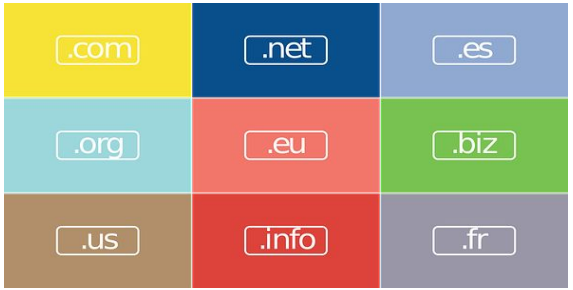
Documents

Intent/Background

Related Docs



Domain



The MainClass class

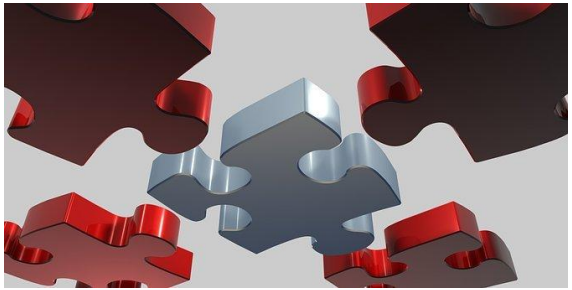
After initializing the pro (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find imum parameters the user wants. After that we will build our DescribeUserInte ss that will call extendAbilities() for each user. Then we will call startProce Handle() wasn't called)

Peripherals

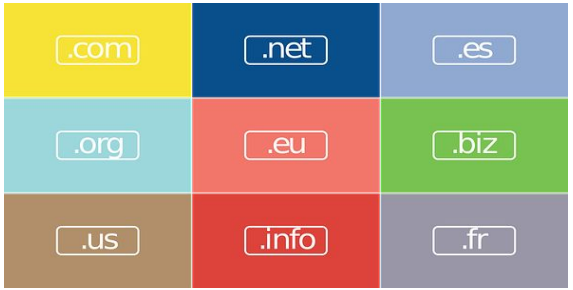
Documents

Intent/Background

Related Docs



Domain



The MainClass class

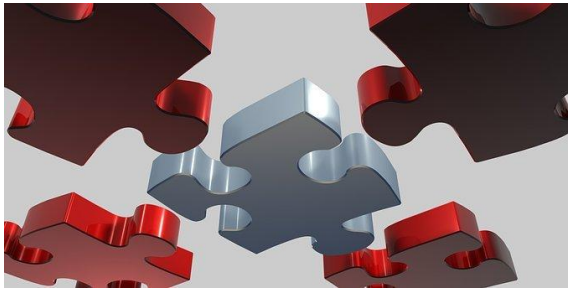
After initializing the pro (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find imum parameters the user wants. After that we will build our DescribeUserInte ss that will call extendAbilities() for each user. Then we will call startProce Handle() wasn't called)

Peripherals

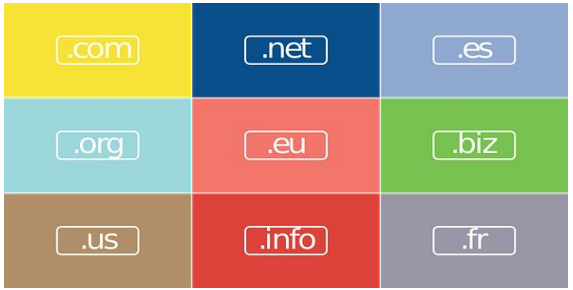
Documents

Intent/Background

Related Docs



Domain



The MainClass class

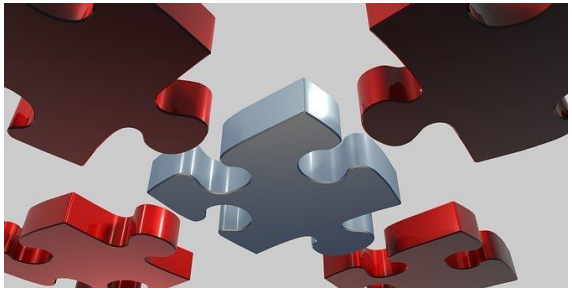
After initializing the pro (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find imum parameters the user wants. After that we will build our DescribeUserInte ss that will call leverageCapabilities() for each user. Then we will call startP rorHandle() wasn't called)

Peripherals

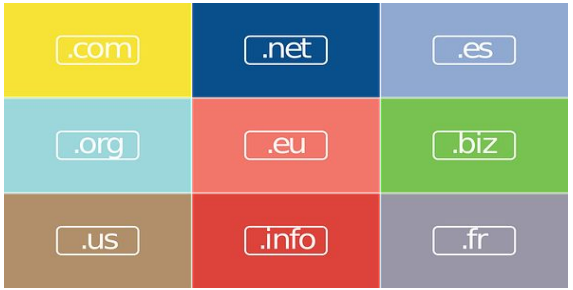
Documents

Intent/Background

Related Docs



Domain



The MainClass class

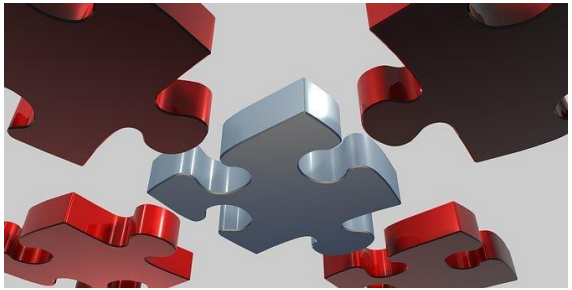
After initializing the pro (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find imum parameters the user wants. After that we will build our DescribeUserInte ss that will call leverageCapabilities() for each user. Then we will call startP rorHandle() wasn't called)

Peripherals

Documents

Intent/Background

Related Docs



Domain

The MainClass class

After initializing the program (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find the maximum parameters the user wants. After that we will build our DescribeUserIntentionClass that will call leverageCapabilities() for each user. Then we will call startProcessingInfo() (only if errorHandle() wasn't called)

Peripherals

Documents

Related Docs

Domain

The MainClass class

After initializing the program (when conf() is TRUE) we will use MAX_PARM (set by configuration) to find the maximum parameters the user wants. After that we will build our DescribeUserIntentionClass that will call leverageCapabilities() for each user. Then we will call startProcessingInfo() (only if errorHandle() wasn't called)

Peripherals

Documents

Domain

Peripherals

Documents

Peripherals

Documents

Maintained?

Peripherals

Peripherals

Code/Task

Peripherals TODO Code/Task

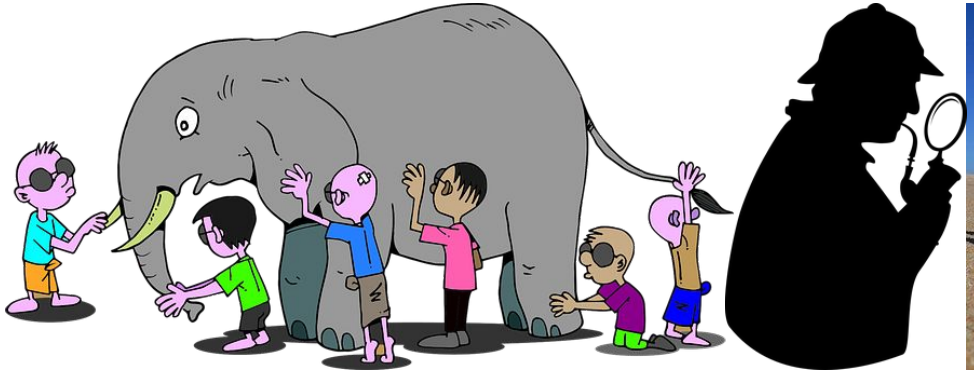
Peripherals TODO Code/Task

Findings

Peripherals TODO Code/Task

Clues Findings

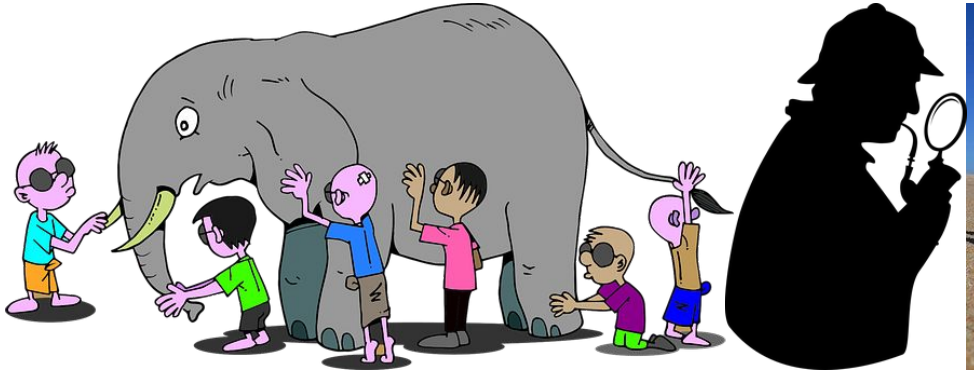
Peripherals TODO Code/Task



Assumptions Clues Findings

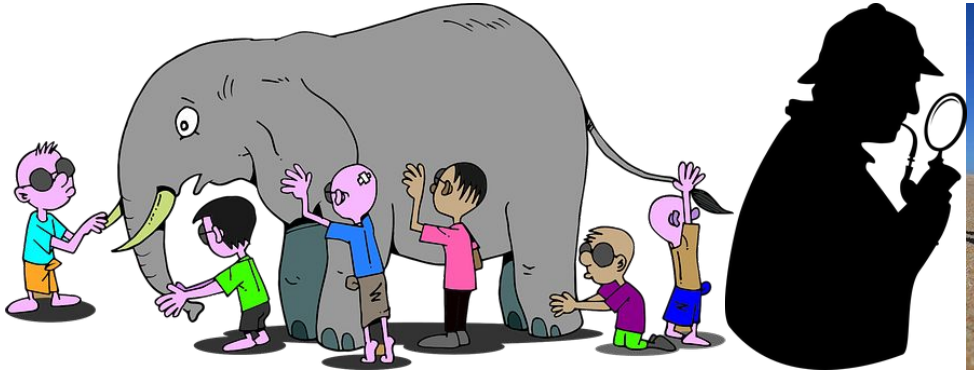
Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task



Conclusions Assumptions Clues Findings

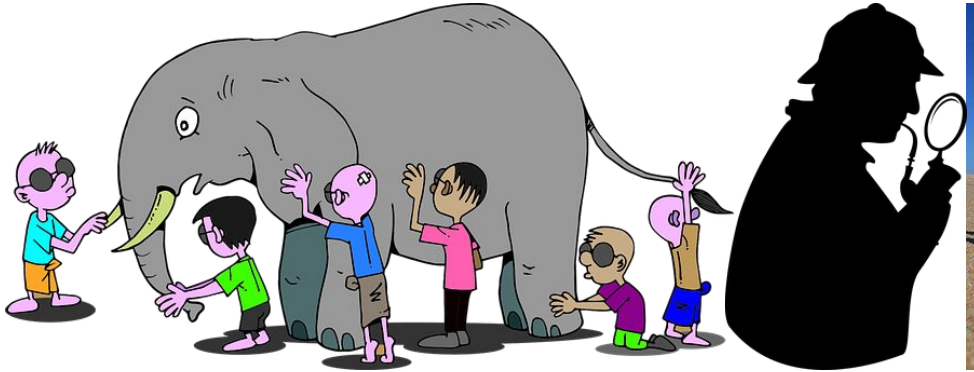
Peripherals TODO Code/Task



Conclusions Assumptions Clues Findings

```
string sInput;
int iLength, iN;
double dblTemp;
bool again = true;

while (again) {
    iN = -1;
    again = false;
    getline(cin, sInput);
    system("cls");
    stringstream(sInput) >> dblTemp;
    iLength = sInput.length();
    if (iLength < 4) {
        again = true;
        continue;
    } else if (sInput[iLength - 3] != '.') {
        again = true;
        continue;
    } while (++iN < iLength) {
        if (isdigit(sInput[iN])) {
            continue;
        } else if (iN == (iLength - 3) ) {
```

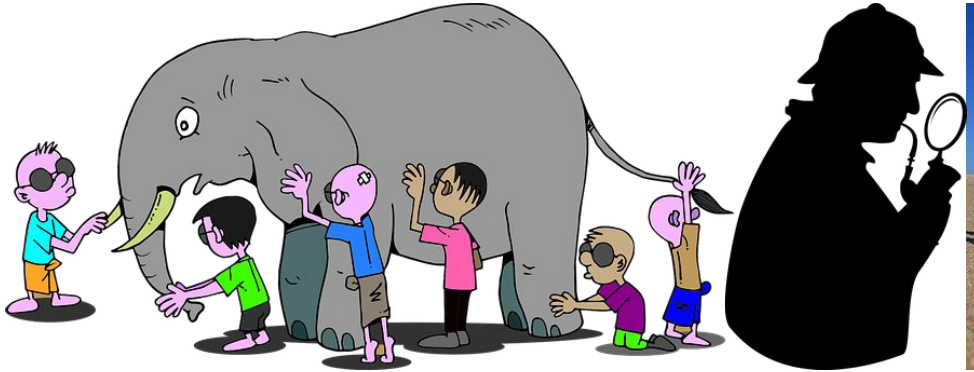
Peripherals TODO Code/Task



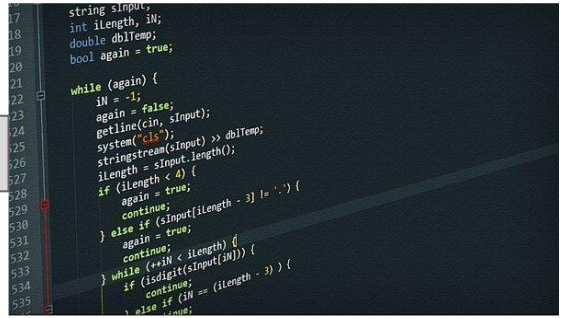
Conclusions Assumptions Clues Findings


```
string sInput;
int iLength, iN;
double dblTemp;
bool again = true;

while (again) {
    iN = -1;
    again = false;
    getline(cin, sInput);
    system("cls");
    stringstream(sInput) >> dblTemp;
    iLength = sInput.length();
    if (iLength < 4) {
        again = true;
        continue;
    } else if (sInput[iLength - 3] != '.') {
        again = true;
        continue;
    } while (++iN < iLength) {
        if (isdigit(sInput[iN])) {
            continue;
        } else if (iN == (iLength - 3) ) {
```

Peripherals TODO Code/Task

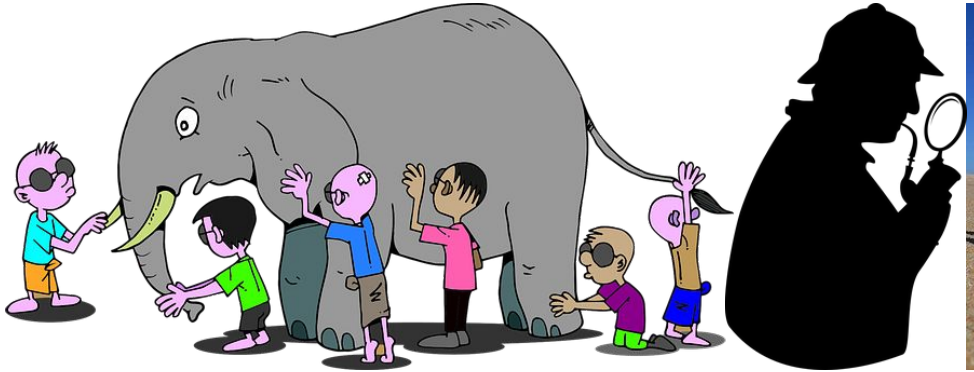


Conclusions Assumptions Clues Findings

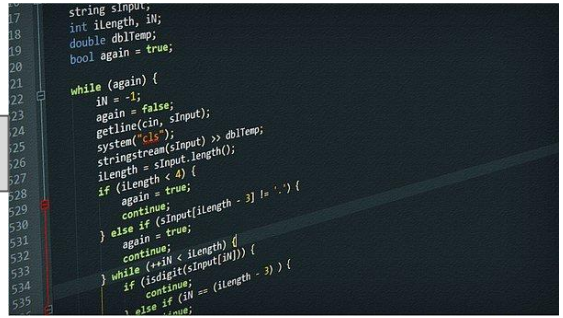


```
string sInput;
int iLength, iN;
double dblTemp;
bool again = true;

while (again) {
    iN = -1;
    again = false;
    getline(cin, sInput);
    system("cls");
    stringstream(sInput) >> dblTemp;
    iLength = sInput.length();
    if (iLength < 4) {
        again = true;
        continue;
    } else if (sInput[iLength - 3] != '.') {
        again = true;
        continue;
    } while (++iN < iLength) {
        if (isdigit(sInput[iN])) {
            continue;
        } else if (iN == (iLength - 3) ) {
```

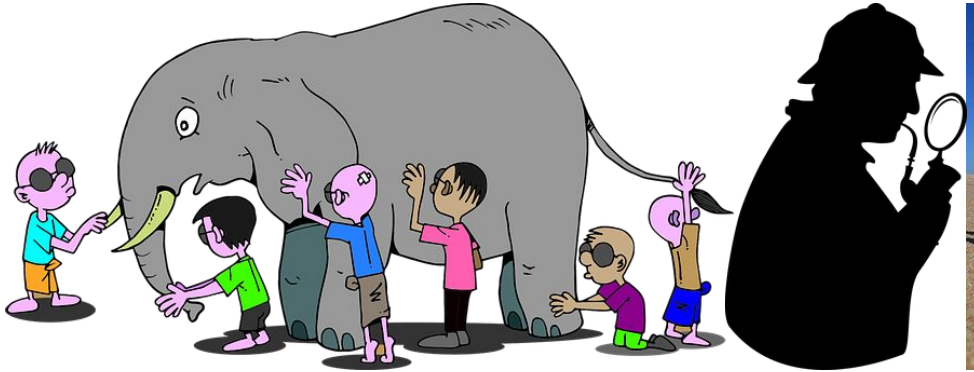
Peripherals TODO Code/Task



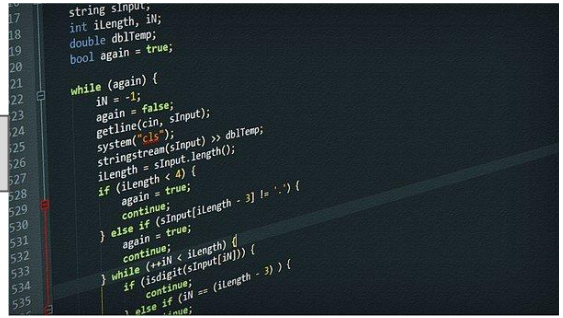
Conclusions Assumptions Clues Findings



Peripherals TODO Code/Task



Conclusions Assumptions Clues Findings



Peripherals TODO Code/Task

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

Code

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

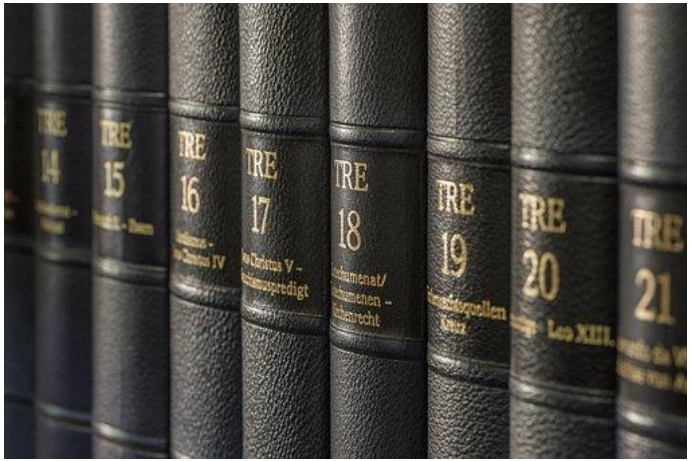
External Tool

Code

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

External Tool



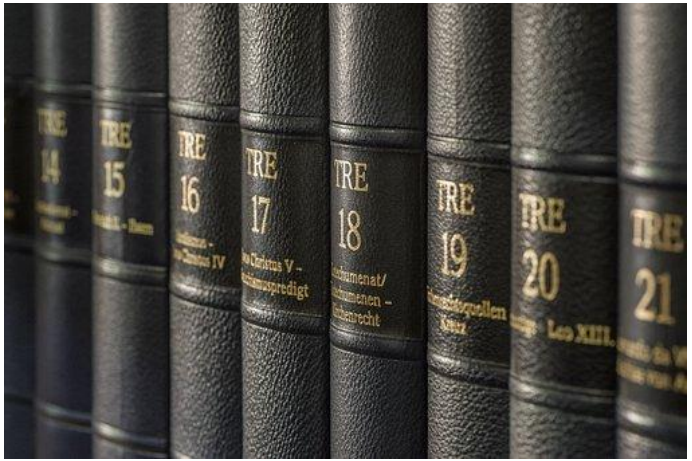
Code

```
string sInput;
int iLength, iN;
double dblTemp;
bool again = true;

while (again) {
    iN = -1;
    again = false;
    getline(cin, sInput);
    system("cls");
    stringstream(sInput) >> dblTemp;
    iLength = sInput.length();
    if (iLength < 4) {
        again = true;
        continue;
    } else if (sInput[iLength - 3] != '.') {
        again = true;
        continue;
    } while (++iN < iLength) {
        if (isdigit(sInput[iN])) {
            continue;
        } else if (iN == (iLength - 3) ) {
```

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

External Tool



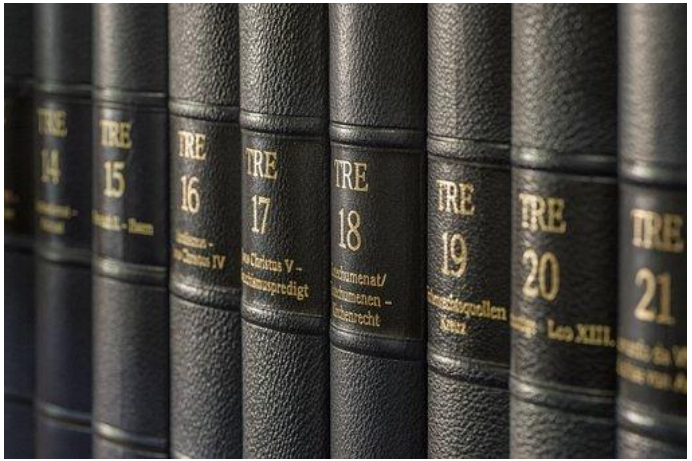
Code

```
string sInput;
int iLength, iN;
double dblTemp;
bool again = true;

while (again) {
    iN = -1;
    again = false;
    getline(cin, sInput);
    system("cls");
    stringstream(sInput) >> dblTemp;
    iLength = sInput.length();
    if (iLength < 4) {
        again = true;
        continue;
    } else if (sInput[iLength - 3] != '.') {
        again = true;
        continue;
    } while (++iN < iLength) {
        if (isdigit(sInput[iN])) {
            continue;
        } else if (iN == (iLength - 3) ) {
```

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

External Tool

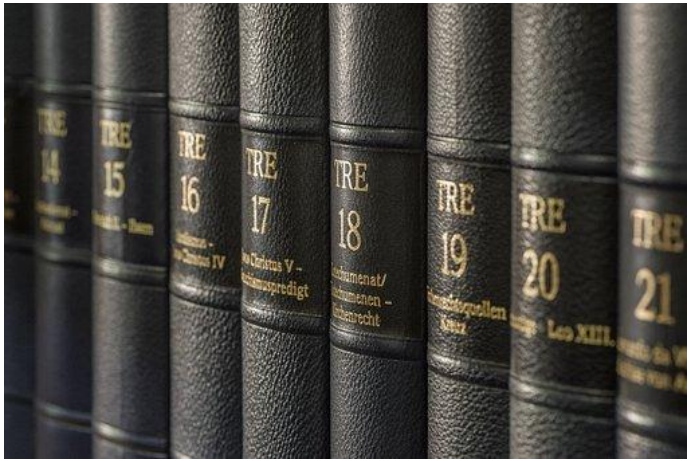


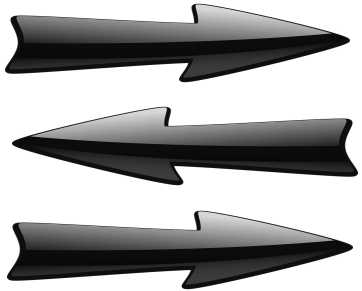
Code

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

External Tool



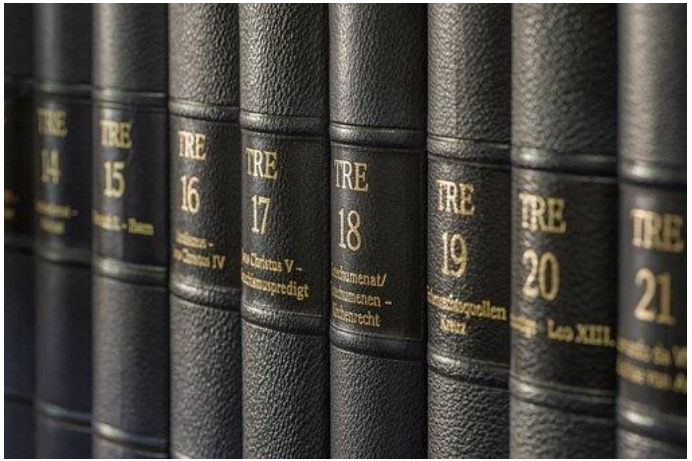


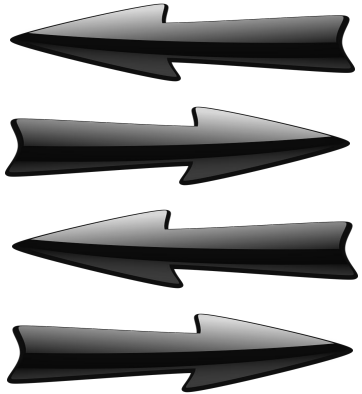
Code

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

External Tool





Code

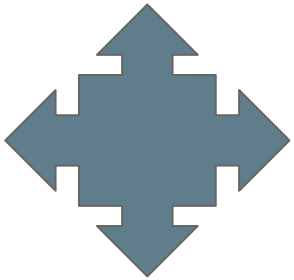
```
string sInput;
int iLength, iN;
double dblTemp;
bool again = true;

while (again) {
    iN = -1;
    again = false;
    getline(cin, sInput);
    system("cls");
    stringstream(sInput) >> dblTemp;
    iLength = sInput.length();
    if (iLength < 4) {
        again = true;
        continue;
    } else if (sInput[iLength - 3] != '.') {
        again = true;
        continue;
    } while (++iN < iLength) {
        if (isdigit(sInput[iN])) {
            continue;
        } else if (iN == (iLength - 3) ) {
```

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task



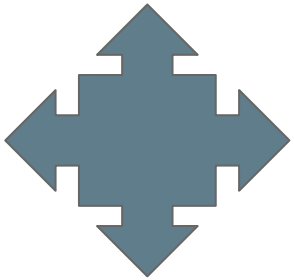
Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

TRE-17 - the dashboard shows only 12H AM/PM format. Probably because of ahhhhh

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task



TRE-17 – dashboard shows only 12H AM/PM format. Probably because of ahhhhh

External

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

```
#> git br -r
origin/MBL-22
origin/MBL-32-qa-1
origin/OFC-41
origin/OFC-44
origin/TRE-15
origin/TRE-17
```

External

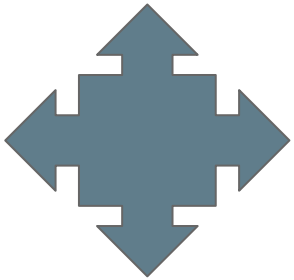
TRE-17 – dashboard shows only 12H AM/PM format. Probably because of ahhhhh

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

Branches

```
#> git br -r
origin/MBL
origin/M    2-qa-1
origin      -41
origi       C-44
ori        TRE-15
ori        n/TRE-17
```



External

TRE-17 - dashboard shows only 12H AM/PM format. Probably because of ahhhhh

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

Branches

```
#> git br -r
origin/MBL
origin/M    2-qa-1
origin      -41
origi       C-44
ori        TRE-15
ori       n/TRE-17
```

External

TRE-17 – dashboard shows only 12H AM/PM format. Probably because of ahhhhh

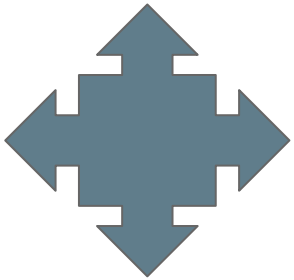
```
#> git log -3 --oneline
OFC-41 # events grid and bar - switched between 'Ok' and 'Cancel' buttons
TRE-17 # changed to 24 hour format in dashboard
MBL-22 # changed error messages in 'change password' screen
```

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

```
#> git br -r
origin/MBL
origin/M    2-qa-1
origin      -41
origi      C-44
ori       TRE-15
ori      n/TRE-17
```

Branches



```
// TRE-17 explains this strange code:
/* ... */
```

External

TRE-17 - dashboard shows only 12H AM/PM format. Probably because of ahhhhh

```
#> git log -3 --   line
OFC-41 # event id and bar - switched between 'Ok' and 'Cancel' buttons
TRE-17 # cha    to 24 hour format in dashboard
MBL-22 # c    ed error messages in 'change password' screen
```

Commits

Ran Regev @regev_ran regev.ran@gmail.com

Peripherals TODO Code/Task

```
#> git br -r
origin/MBL
origin/M  2-qa-1
origin  -41
origi  FC-44
ori   /TRE-15
ori  n/TRE-17
```

Branches

```
// TRE-17 explains this   nge code:
/* ... */
```

Code

TRE-17 –   dashboard shows only 12H AM/PM   mat. Probably because of ahhhhh

External

```
#> git log -3 --   ine
OFC-41 # event   id and bar - switched between 'Ok' and 'Cancel' buttons
TRE-17 # cha    to 24 hour format in dashboard
MBL-22 # c   ed error messages in 'change password' screen
```

Commits

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Ran Regev @regev_ran regev.ran@gmail.com

Comments

What/Why

Ran Regev @regev_ran regev.ran@gmail.com

Comments

What

What/Why

Ran Regev @regev_ran regev.ran@gmail.com

Comments

What/Why

What

Ran Regev @regev_ran regev.ran@gmail.com

Comments

What/Why

What

Ran Regev @regev_ran regev.ran@gmail.com

Comments

What/Why

What


```
// check if total is zero
if ( 0 == total )
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

What/Why

What


```
// check if total is zero
if ( 0 == total )
```

Why

Ran Regev @regev_ran regev.ran@gmail.com

Comments

What/Why

What

Why

```
// check if total is zero
if ( 0 == total )
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

What/Why

What


```
// check if total is zero
if ( 0 == total )
```

Why

Ran Regev @regev_ran regev.ran@gmail.com

Comments

What/Why

What


```
// check if total is zero
if ( 0 == total )
```

Why


```
// when the total stayed zero, we
// must take a special action
// (see MBL-22 for more details)
if ( 0 == total )
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Share

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Share

```
// need to commit received offset + 1 !!!
kafka::commit( _offset + 1 );
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Share

```
// need to commit received offset + 1 !!!
kafka::commit( _offset + 1 );
```

Investigation

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Share

```
// need to commit received offset + 1 !!!
kafka::commit( _offset + 1 );
```

Investigation

Confusion

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Share

```
// need to commit received offset + 1 !!!
kafka::commit( _offset + 1 );
```

Investigation

Confusion

Share!

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Story

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Story

```
// creates a copy of the element in the queue.
// this allows unlocking the queue before handling the element.
// handleQueueElement gets a reference to this local t.
// the queue itself may be moved.
T t = std::move( _queue.front() );
_queue.pop();
_queueLock.unlock();
_handler.handleQueueElement( t );
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Story

```
// creates a copy of the element in the queue.
// this allows unlocking the queue before handling the element.
// handleQueueElement gets a reference to this local t.
// the queue itself may be moved.
T t = std::move( _queue.front() );
_queue.pop();
_queueLock.unlock();
_handler.handleQueueElement( t );
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Story

```
// creates a copy of the element in the queue.
// this allows unlocking the queue before handling the element.
// handleQueueElement gets a reference to this local t.
// the queue itself may be moved.
T t = std::move( _queue.front() );
_queue.pop();
_queueLock.unlock();
_handler.handleQueueElement( t );
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Story

```
// creates a copy of the element in the queue.
// this allows unlocking the queue before handling the element.
// handleQueueElement gets a reference to this local t.
// the queue itself may be moved.
T t = std::move( _queue.front() );
_queue.pop();
_queueLock.unlock();
_handler.handleQueueElement( t );
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Story

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Story

```
// a note about the period -
// in each cycle we do the operation and _only then_ start the
// countdown to the end of period.
// therefore the time between operations is period-time + operation time.
// it is fine if the operation is short and the period is long.
// it might be odd with short period and long operation.
// e.g. #1: period = 1 second. operation takes 5 milli:
// the time between the _start_ of one operation and its successive would be:
// 1.005 seconds.
// e.g. #1: period = 10 milli. operation takes 15 milli:
// the time between the _start_ of one operation and its successive would be:
// 25 milli. which is *not* what we were asking for (as asked for 10).
// but what are the options? interleaving operations?
// forcing the operation object to be re-entered?
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Story

```
// a note about the period -
// in each cycle we do the operation and _only then_ start the
// countdown to the end of period.
// therefore the time between operations is period-time + operation time.
// it is fine if the operation is short and the period is long.
// it might be odd with short period and long operation.
// e.g. #1: period = 1 second. operation takes 5 milli:
// the time between the _start_ of one operation and its successive would be:
// 1.005 seconds.
// e.g. #1: period = 10 milli. operation takes 15 milli:
// the time between the _start_ of one operation and its successive would be:
// 25 milli. which is *not* what we were asking for (as asked for 10).
// but what are the options? interleaving operations?
// forcing the operation object to be re-entered?
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Story

```
// a note about the period -
// in each cycle we do the operation and _only then_ start the
// countdown to the end of period.
// therefore the time between operations is period-time + operation time.
// it is fine if the operation is short and the period is long.
// it might be odd with short period and long operation.
// e.g. #1: period = 1 second. operation takes 5 milli:
// the time between the _start_ of one operation and its successive would be:
// 1.005 seconds.
// e.g. #1: period = 10 milli. operation takes 15 milli:
// the time between the _start_ of one operation and its successive would be:
// 25 milli. which is *not* what we were asking for (as asked for 10).
// but what are the options? interleaving operations?
// forcing the operation object to be re-entered?
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

This comment is no more

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

It has ceased to be!

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

It Expired!

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Bereft of life!

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

Rests in peace!

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

It is off the twig!

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

It shuffled off its mortal coil

Ran Regev @regev_ran regev.ran@gmail.com

Comments

Maintained?

```
std::shared_ptr<ByteBuffer> DataBufferPool::GetDataBuffer()
{
    std::shared_ptr<ByteBuffer> DataBuffer;
    // [Dubin]: we found out the timeout slow the system too much.
    // when data buffer pool is full. Therefore, We don't user retries.
    DataBuffer = data_buffers_pool_->acquire();
    if ( DataBuffer == nullptr )
    {
        dropped_data_buffers_++;
    }
    return DataBuffer;
}
```

THIS IS AN EX COMMENT !

Ran Regev @regev_ran regev.ran@gmail.com

Accessibility

Ran Regev @regev_ran regev.ran@gmail.com

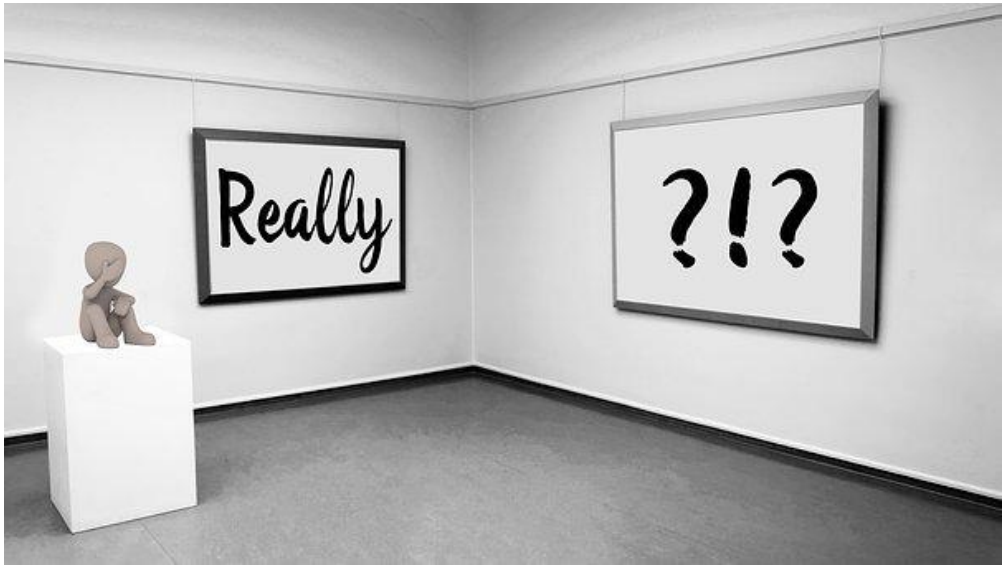
Accessibility

```
using intset = std::set<int>;
using intvec = std::vector<int>;

bool f( intset& exists, intset& newset  )
{
    bool changed=false;
    bool added=false;
    int newSize = newset.size();
    int existsSize = exists.size();
    intvec toRemove;

    if ( newSize <= existsSize ) {
        changed = getChanged( exists, newset, toRemove ); }
    else {
        changed = getChanged( exists, newset, toRemove );
        added=true; }

    if ( changed ) {
        // do staff with toRemove
        return true; }
    else if ( added ) {
        // do staff with newset
        return false; }
    else {
        return false; }
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Accessibility

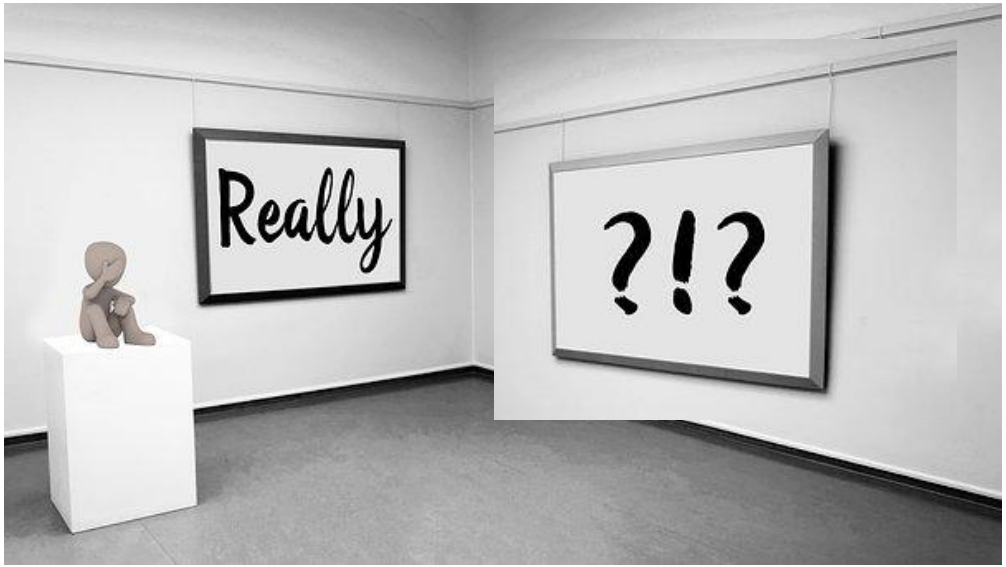
```
using intset = std::set<int>;
using intvec = std::vector<int>;

bool f( intset& exists, intset& newset  )
{
    bool changed=false;
    bool added=false;
    int newSize = newset.size();
    int existsSize = exists.size();
    intvec toRemove;

    if ( newSize <= existsSize ) {
        changed = getChanged( exists, newset, toRemove ); }
    else {
        changed = getChanged( exists, newset, toRemove );
        added=true; }

    if ( changed ) {
        // do staff with toRemove
        return true; }
    else if ( added ) {
        // do staff with newset
        return false; }
    else {
        return false; }
}
```



Ran Regev @regev_ran regev.ran@gmail.com

Accessibility

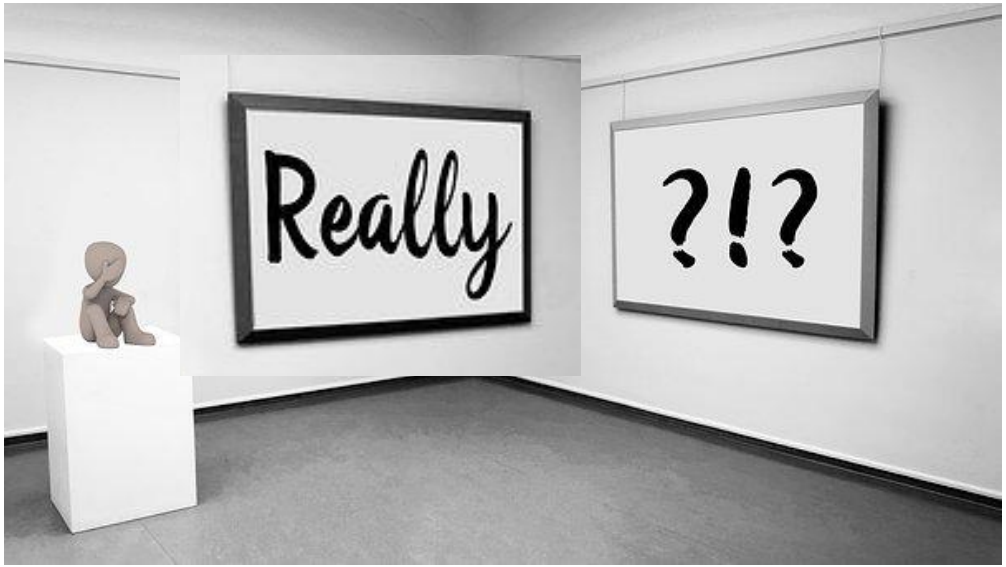
```cpp
using intset = std::set<int>;
using intvec = std::vector<int>;

bool f( intset& exists, intset& newset  )
{
    bool changed=false;
    bool added=false;
    int newSize = newset.size();
    int existsSize = exists.size();
    intvec toRemove;

    if ( newSize <= existsSize ) {
        changed = getChanged( exists, newset, toRemove ); }
    else {
        changed = getChanged( exists, newset, toRemove );
        added=true; }

    if ( changed ) {
        // do staff with toRemove
        return true; }
    else if ( added ) {
        // do staff with newset
        return false; }
    else {
        return false; }
}
```



Ran Regev @regev_ran regev.ran@gmail.com

Accessibility

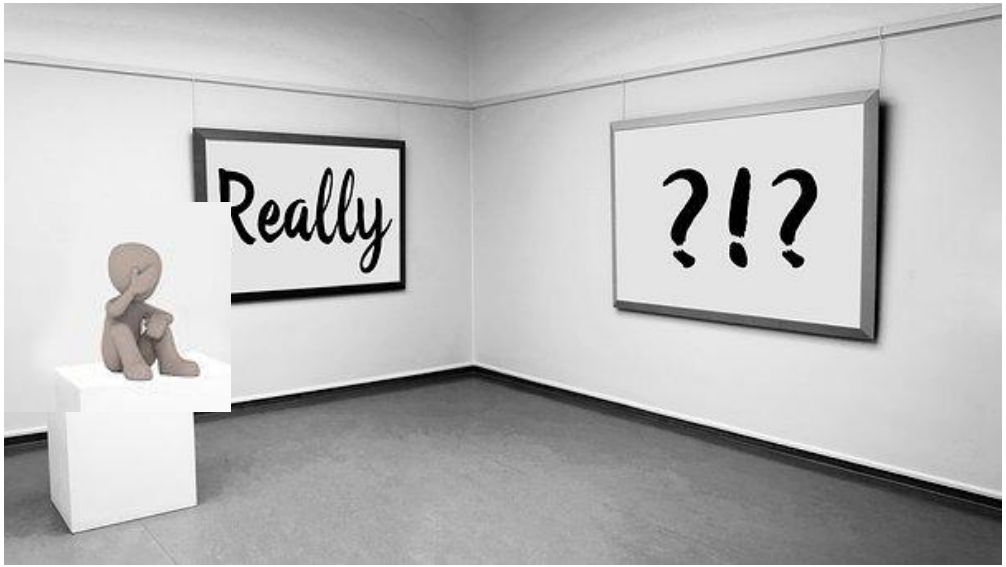
```cpp
using intset = std::set<int>;
using intvec = std::vector<int>;

bool f( intset& exists, intset& newset  )
{
    bool changed=false;
    bool added=false;
    int newSize = newset.size();
    int existsSize = exists.size();
    intvec toRemove;

    if ( newSize <= existsSize ) {
        changed = getChanged( exists, newset, toRemove ); }
    else {
        changed = getChanged( exists, newset, toRemove );
        added=true; }

    if ( changed ) {
        // do staff with toRemove
        return true; }
    else if ( added ) {
        // do staff with newset
        return false; }
    else {
        return false; }
}
```



Ran Regev @regev_ran regev.ran@gmail.com

Accessibility

```
using intset = std::set<int>;
using intvec = std::vector<int>;

bool f( intset& exists, intset& newset  )
{
    bool changed=false;
    bool added=false;
    int newSize = newset.size();
    int existsSize = exists.size();
    intvec toRemove;

    if ( newSize <= existsSize ) {
        changed = getChanged( exists, newset, toRemove ); }
    else {
        changed = getChanged( exists, newset, toRemove );
        added=true; }

    if ( changed ) {
        // do staff with toRemove
        return true; }
    else if ( added ) {
        // do staff with newset
        return false; }
    else {
        return false; }
}
```



Ran Regev @regev_ran regev.ran@gmail.com

Accessibility

```
intset& g( intset& exists, const intset& newset  )
{
    intvec toDelete;
    intvec toAdd;
    intvec common;

    std::set_difference(
        exists.begin(), exists.end(),
        newset.begin(), newset.end(),
        std::inserter( toDelete, toDelete.begin() )
    );
    std::set_difference(
        newset.begin(), newset.end(),
        exists.begin(), exists.end(),
        std::inserter( toAdd, toAdd.begin() )
    );
    std::set_intersection(
        exists.begin(), exists.end(),
        newset.begin(), newset.end(),
        std::inserter( common, common.begin() )
    );
    // do staff with toDelete
    // do staff with toAdd
    // do staff with common
    return exists;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Accessibility

```
intset& g( intset& exists, const intset& newset  )
{
    intvec toDelete;
    intvec toAdd;
    intvec common;

    std::set_difference(
        exists.begin(), exists.end(),
        newset.begin(), newset.end(),
        std::inserter( toDelete, toDelete.begin() )
    );
    std::set_difference(
        newset.begin(), newset.end(),
        exists.begin(), exists.end(),
        std::inserter( toAdd, toAdd.begin() )
    );
    std::set_intersection(
        exists.begin(), exists.end(),
        newset.begin(), newset.end(),
        std::inserter( common, common.begin() )
    );
    // do staff with toDelete
    // do staff with toAdd
    // do staff with common
    return exists;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Accessibility

```
intset& g( intset& exists, const intset& newset  )
{
    intvec toDelete;
    intvec toAdd;
    intvec common;

    std::set_difference(
        exists.begin(), exists.end(),
        newset.begin(), newset.end(),
        std::inserter( toDelete, toDelete.begin() )
    );
    std::set_difference(
        newset.begin(), newset.end(),
        exists.begin(), exists.end(),
        std::inserter( toAdd, toAdd.begin() )
    );
    std::set_intersection(
        exists.begin(), exists.end(),
        newset.begin(), newset.end(),
        std::inserter( common, common.begin() )
    );
    // do staff with toDelete
    // do staff with toAdd
    // do staff with common
    return exists;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Accessibility

```
intset& g( intset& exists, const intset& newset  )
{
    intvec toDelete;
    intvec toAdd;
    intvec common;

    std::set_difference(
        exists.begin(), exists.end(),
        newset.begin(), newset.end(),
        std::inserter( toDelete, toDelete.begin() )
    );
    std::set_difference(
        newset.begin(), newset.end(),
        exists.begin(), exists.end(),
        std::inserter( toAdd, toAdd.begin() )
    );
    std::set_intersection(
        exists.begin(), exists.end(),
        newset.begin(), newset.end(),
        std::inserter( common, common.begin() )
    );
    // do staff with toDelete
    // do staff with toAdd
    // do staff with common
    return exists;
}
```

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Resources

Pictures

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Resources

Pictures

https://pixabay.com/

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Resources

Pictures

https://pixabay.com/

Related Talks

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Resources

Pictures

https://pixabay.com/

Related Talks

Kate Gregory (@gregcons)

Summary

Resources

Pictures

https://pixabay.com/

Related Talks

Kate Gregory (@gregcons)

Classical Music

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Resources

Pictures

https://pixabay.com/

Related Talks

Kate Gregory (@gregcons)

Classical Music

Joshua Weilerstein - Sticky Notes Podcast

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Ran Regev @regev_ran regev.ran@gmail.com

Summary

Thank You

Ran Regev @regev_ran regev.ran@gmail.com