

close(fd);

With great power comes great responsibility.

Envisioning Tomorrow's Code

Sally Trivino

16 May 2024

close(fd);

With great power comes great responsibility.

Envisioning Tomorrow's Code

Sally Trivino

16 May 2024

close(fd);

With great power comes great responsibility.

Envisioning Tomorrow's Code

Sally Trivino

16 May 2024

close(fd);

With great power comes great responsibility.

Envisioning Tomorrow's Code

Sally Trivino

16 May 2024

Building a better working world

Before we start

My stance on the vi vs. Emacs as well as Tab vs. Space discussions:

```
17 set expandtab  
NORMAL .vimrc
```


The death of Coding

```
LANGUAGE SQLSCRIPT  
OPTIONS READ-ONLY
```

```
USING zdemo_abap_carr_ve.
```

```
* Simple data selection
```

```
carr_tab = SELECT carrid, carrname, currcode, url  
FROM "ZDEMO_ABAP_CARR_VE"  
ORDER BY carrid;
```

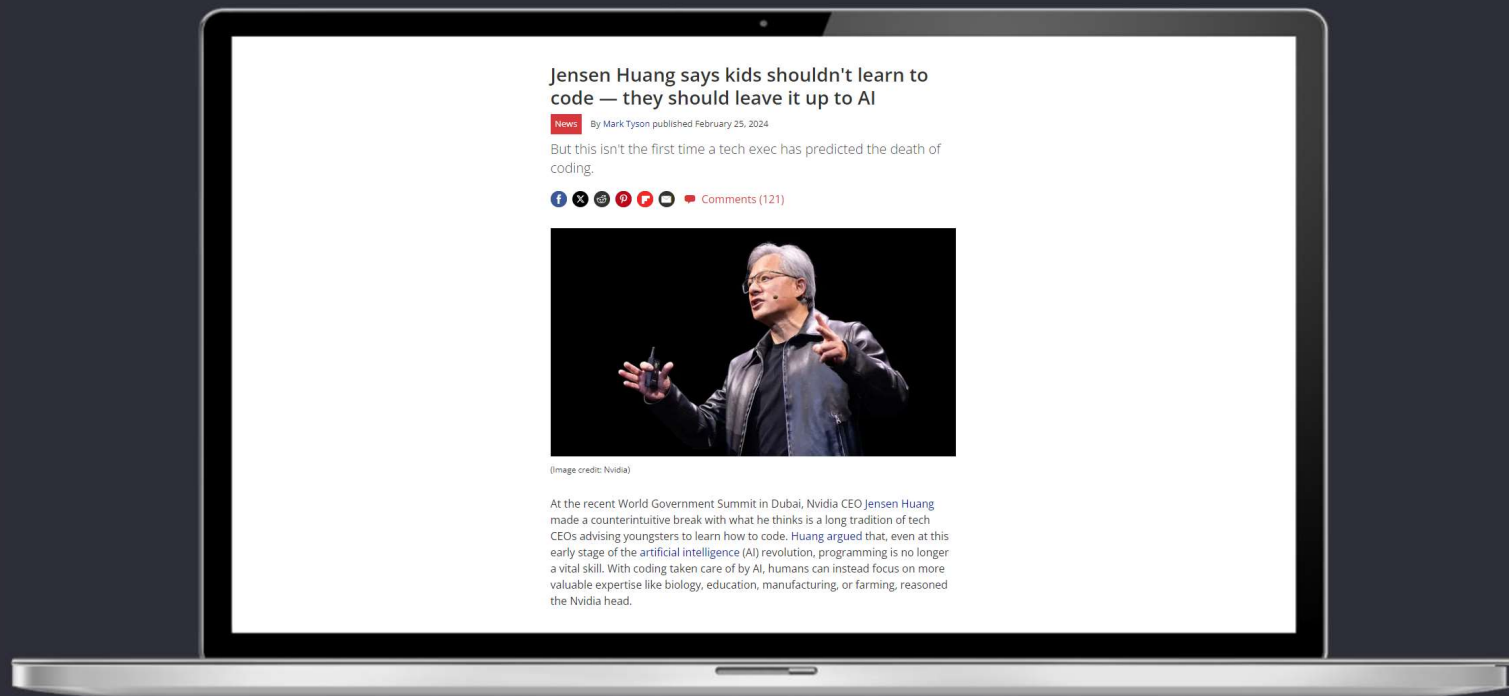
```
ENDMETHOD.
```

```
METHOD select_get_carr_fli  
BY DATABASE PROCEDURE  
FOR HDB
```

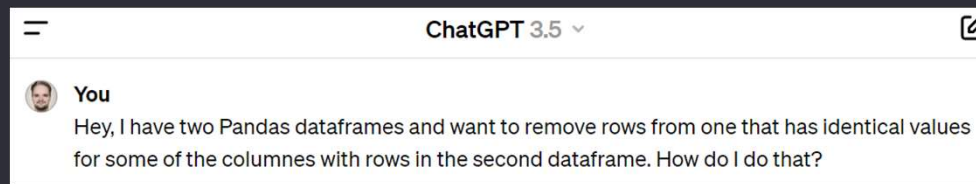
```
LANGUAGE SQLSCRIPT  
OPTIONS READ-ONLY
```

```
demo_abap_amdp=>get_carr_fli.  
an AMDP table function as specified  
ABAP_AMDP=>GET_CARR_F  
);
```


The death of Coding



A real-life example




A real-life example

ChatGPT

You can use the `merge` function in Pandas to join the two dataframes based on the common columns, and then use the `drop_duplicates` function to remove the rows that have identical values. Here's an example code:

python

 Copy code

```
import pandas as pd

# create two example dataframes
df1 = pd.DataFrame({'A': [1, 2, 3], 'B': [4, 5, 6], 'C': [7, 8, 9]})
df2 = pd.DataFrame({'A': [2, 3, 4], 'B': [5, 6, 7], 'C': [8, 9, 10]})

# merge the dataframes on columns 'A' and 'B'
merged_df = pd.merge(df1, df2, on=['A', 'B'], how='left')

# drop the rows with identical values
result_df = merged_df.drop_duplicates(subset=['A', 'B', 'C_x'], keep=False)

# result_df will have the rows from df1 that don't have identical values in columns 'A' and 'B'
```



A real-life example



You

The last step does not work. merged_df and result_df are identical. Rows with indices 1 and 2 should have been removed.


The death of Coding - but it is not this day

 ChatGPT

I apologize for the confusion. You are correct that the code I provided would not remove the rows with identical values as intended.

Here's an updated code snippet that should work for your scenario:

python

 Copy code

```
import pandas as pd

# create two example dataframes
df1 = pd.DataFrame({'A': [1, 2, 3], 'B': [4, 5, 6], 'C': [7, 8, 9]})
df2 = pd.DataFrame({'A': [2, 3, 4], 'B': [5, 6, 7], 'C': [8, 9, 10]})

# identify the rows in df1 that have identical values in the columns of interest
dup_rows = df1[df1[['A', 'B']].apply(lambda x: tuple(x), axis=1).isin(df2[['A', 'B']])]

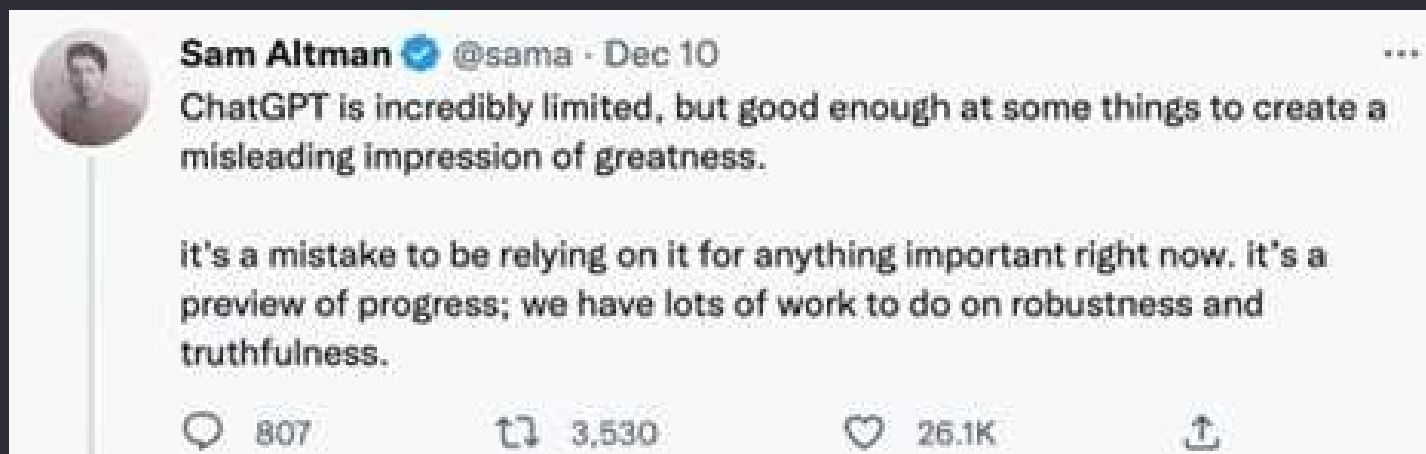
# remove the identified rows from df1
result_df = df1[~df1.index.isin(dup_rows.index)]

# result_df will have the rows from df1 that don't have identical values in columns '
```


A real-life example

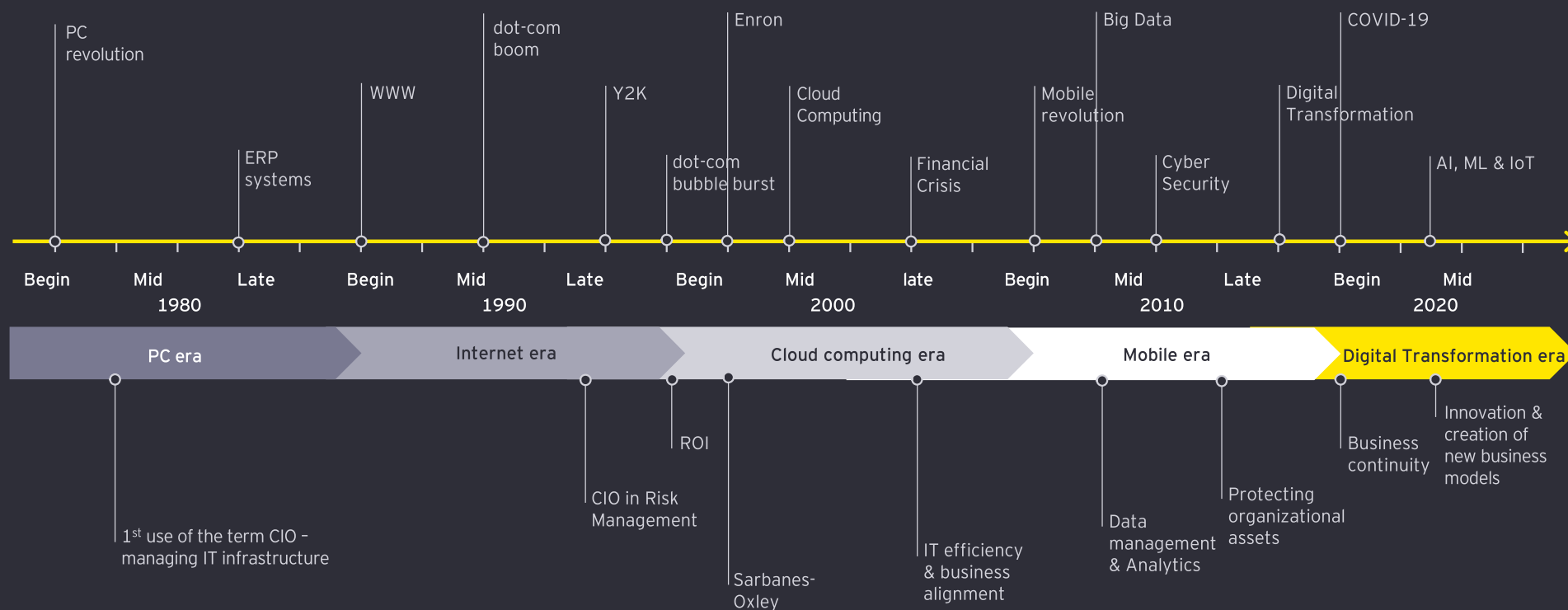
ChatGPT can make mistakes. Consider checking important information.

The death of Coding - but it is not this day



Transitioning role of the CIO

How times have changed



Is your algorithm an ethical one?

```
METHOD select_carriers  
  BY DATABASE PROCEDURE  
  FOR HDB  
  LANGUAGE SQLSCRIPT  
  OPTIONS READ-ONLY  
  USING zdemo_abap_carr ve.  
  data selection  
  carr_tab = SELECT carrid, carrname, currcode, url  
              FROM "ZDEMO_ABAP_CARR_VE"  
              ORDER BY carrid;  
ENDMETHOD.
```

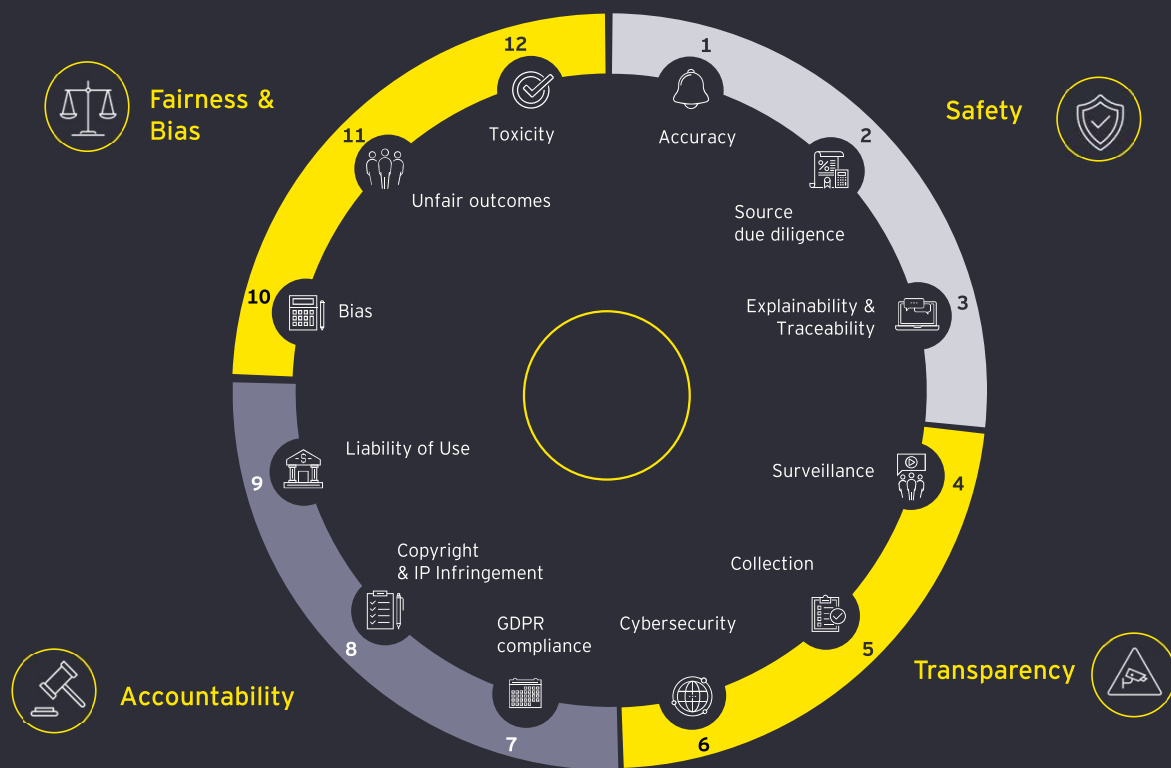
```
METHOD select_get_carr_fli  
  BY DATABASE PROCEDURE  
  FOR HDB  
  LANGUAGE SQLSCRIPT  
  OPTIONS READ-ONLY  
  USING zdemo_abap_carr fli.  
  amdp=>get_carr_fli.  
  function as specified after USING  
  GET_CARR_FLI"
```



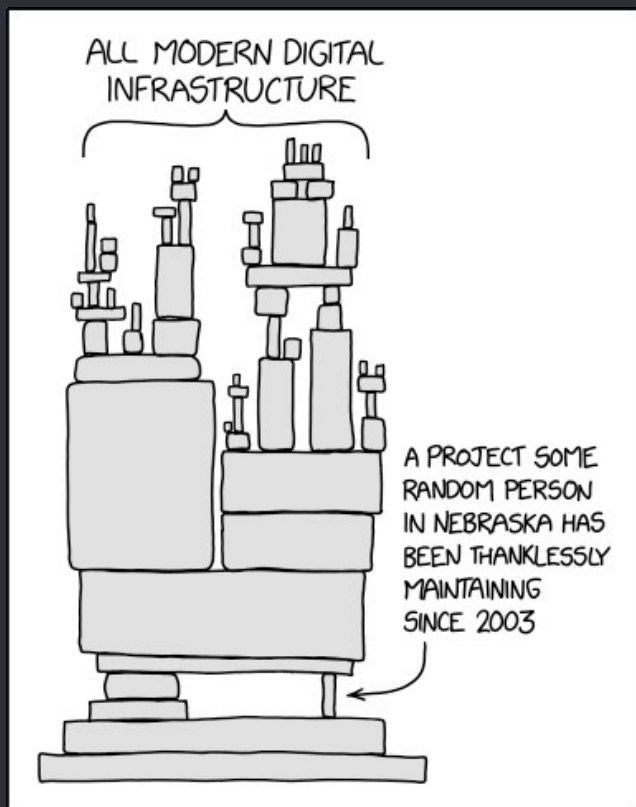
```
1
2 // 10,000 if-statements
3
4 if() {
5     if() {
6         if() {
7             if() {
8                 if() {
9                     if() {
10                        if() {
11                            if() {
12                                if() {
13                                    if() {
14                                        if() {
```



12 Considerations for ethical coding and responsible AI



Accountability



Date: Fri, 29 Mar 2024 08:51:26 -0700

From: Andres Freund <andres@...razel.de>

To: oss-security@...ts.openwall.com

Subject: backdoor in upstream xz/liblzma leading to ssh server compromise

Hi,

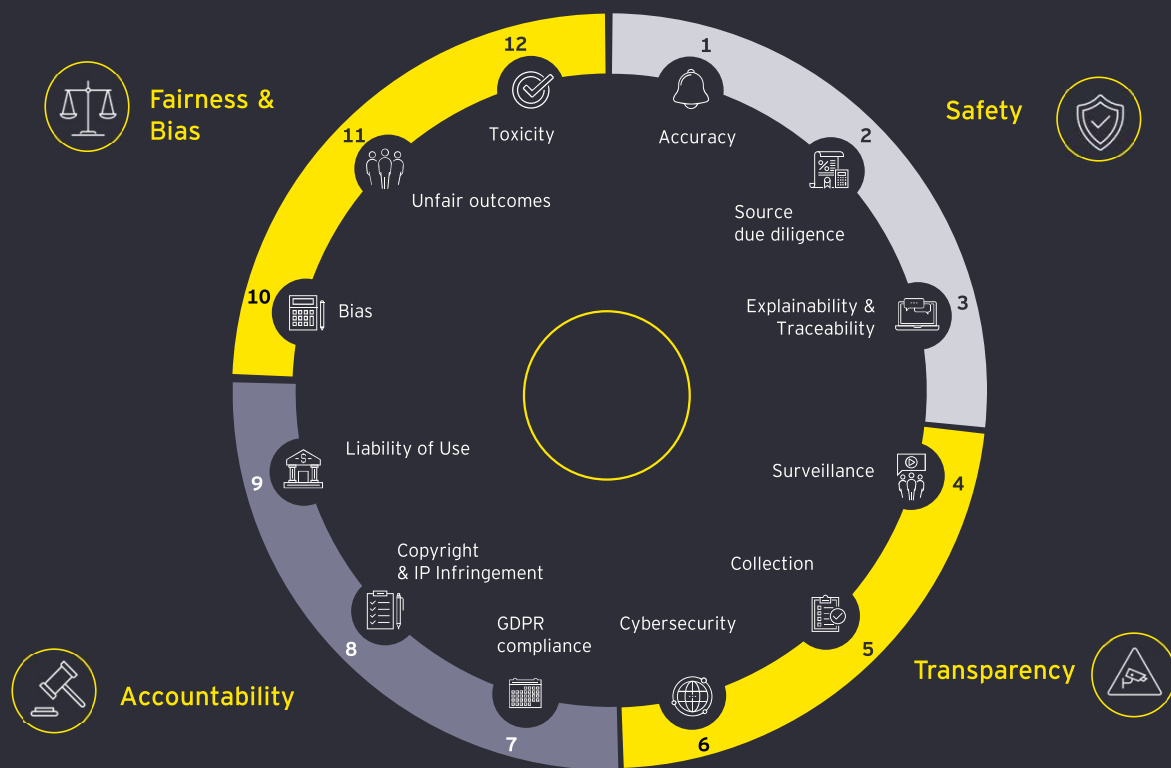
After observing a few odd symptoms around liblzma (part of the xz package) on

Debian sid installations over the last weeks (logins with ssh taking a lot of

CPU, valgrind errors) I figured out the answer:

The upstream xz repository and the xz tarballs have been backdoored.

12 Considerations for ethical coding and responsible AI



With great power comes
great responsibility!

