

Data Analysis, Reporting, and Data Visualization

By Rose Chauvin, Clara Decros,
Camille Carraco and H  l  na Guertin

Mrs Belim, M. Bureau, M. Garnier and Mrs Dupuy

QUICK

- I - Context
 - Context and needs presentation
 - For what purpose?
 - Presentation of the dataset
- II - Organization
 - Organization and planning of work within the group
 - Tools used
- III - Project
 - Presentation interface
 - Chart



I - Context

- **Context and needs presentation**
- For what purpose?
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart

- **Public authorities**
- **Researchers and experts in domestic accident prevention**
- **Associations working on risk prevention**
- **Insurance companies**
- **The general public (parents, families, seniors)**

I - Context

- Context and needs presentation
- **For what purpose?**
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart



Risks of everyday life



Supports decisions



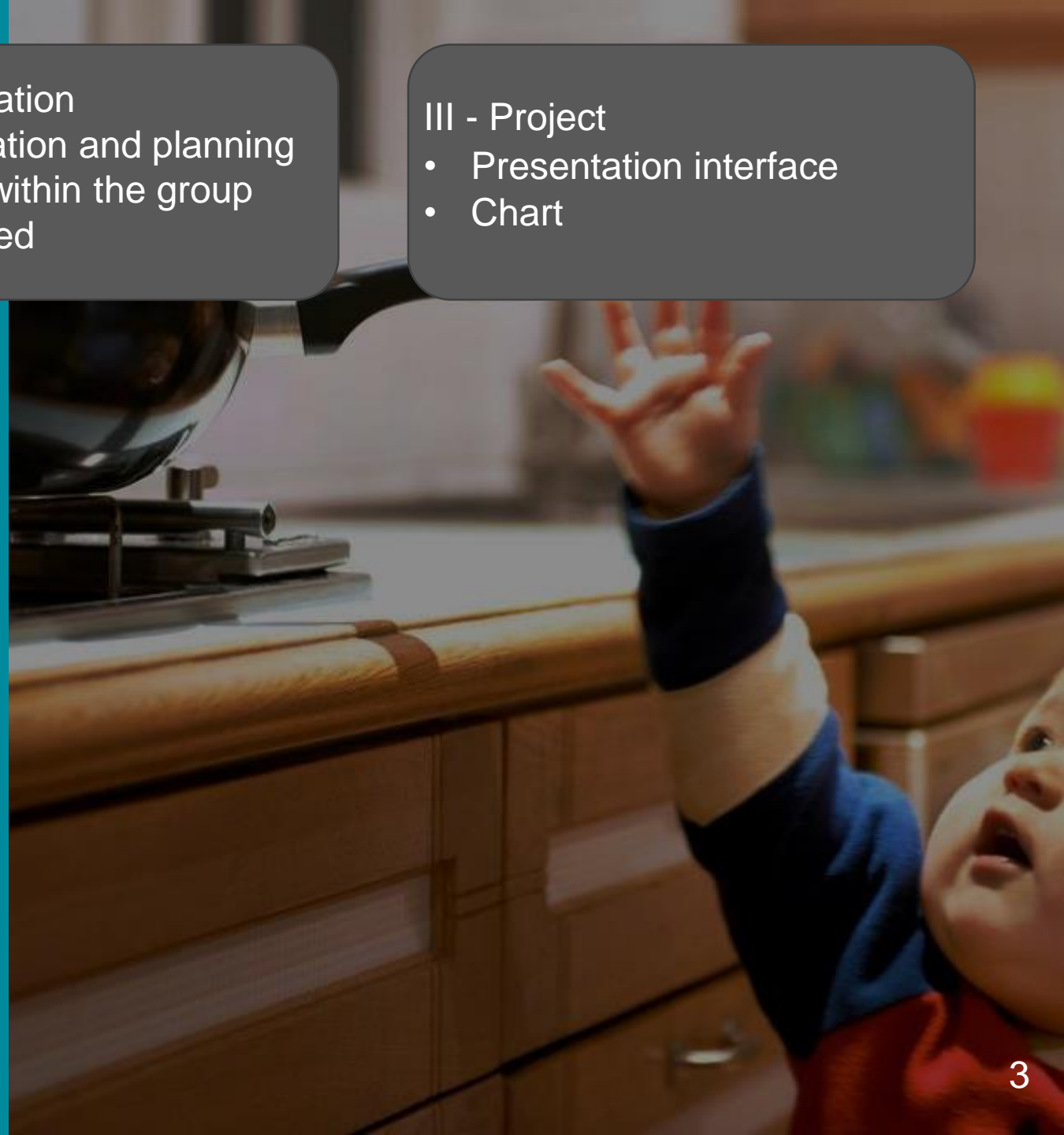
Inform and raise public awareness



Help with prevention



Create an automated tool



I - Context

- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart

- Answer: volunteering
- Personal Information and Lifestyle
- Health and accidents
- Housing environment

I - Context

- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart

Projets

SAÉ - Analyse de données, reporting et datavisualisation ...

🌐 Résumé **Tableau** 📋 Liste 📅 Calendrier ⌚ Chronologie 📄 Formulaires 📄 Pages 📎 Pièces jointes

🔍 Rechercher dans le ... 👤 C HG Filtre ▼

À FAIRE 5	EN COURS 3	TERMINÉ(E) ✓ 4
<p>déposer le projet</p> <p>📅 6 juin 2025</p> <p>☑ SAE-5 =</p>	<p>Power Point Oral d'Anglais</p> <p>🕒 4 juin 2025</p> <p>☑ SAE-2 = HG</p>	<p>Recherche des Données externes afin de comparer les résultats</p> <p>📅 27 mai 2025</p> <p>☑ SAE-10 ✓ = HG</p>
<p>compte rendu clara</p> <p>☑ SAE-11 =</p>	<p>Bilan global du groupe</p> <p>☑ SAE-16 = HG</p>	<p>Nettoyage des données</p> <p>📅 28 mai 2025</p> <p>☑ SAE-8 ✓ = HG</p>
<p>compte rendu rose</p> <p>☑ SAE-12 =</p>	<p>Créer des graphs pour la comparaison</p> <p>☑ SAE-17 = C</p>	<p>compte rendu Hélène</p> <p>☑ SAE-14 ✓ =</p>
<p>compte rendu camille</p> <p>☑ SAE-13 =</p>		
<p>+ Créer</p>		

ABBoud



I - Context

- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart



I - Context

- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart

Library import

```
8 import pandas as pd
9 import json
10 from datetime import datetime
11 import re
12 import tkinter as tk
13 from tkinter import scrolledtext, filedialog, messagebox
14 import os
15 import subprocess
16 import time
17 import requests
18 import datetime as date
19 from pywinauto import Application
20 from datetime import date
```

Some def

```
141 def extraire_heure(date_str):
142     try:
143         date = pd.to_datetime(date_str, errors='coerce', dayfirst=True)
144         if pd.notna(date):
145             return date.strftime('%H:%M')
146     except Exception:
147         return ""
148     return ""
149
150 def get_tranche_age(age):
151     if pd.isna(age):
152         return "" # laisser vide si âge inconnu
153     elif age < 18:
154         return "<18"
155     elif 18 <= age <= 35:
156         return "18-35"
157     elif 36 <= age <= 50:
158         return "36-50"
159     elif 51 <= age <= 70:
160         return "51-70"
161     elif age > 70:
162         return "70+"
163     else:
164         return ""
```


I - Context

- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart

Data cleaning

```
507 # ----- Suppression des colonnes ----- #
508
509 df_BD.drop([col for col in df_BD.columns if col.lower().startswith("colonne")], axis=1, inplace=True)
510 # Supprimer les colonnes commençant par "Colonne"
511 df_Acc = df_Acc.drop([col for col in df_Acc.columns if col.startswith("colonne")], axis=1)
512 df_BD = df_BD.drop([col for col in df_BD.columns if col.startswith("colonne")], axis=1)
513
514 # ----- Modification dans df_BD ----- #
515
516 if 'taille_cm' in df_BD.columns:
517     df_BD['taille_cm'] = df_BD['taille_cm'].apply(convertir_taille)
518
519 if 'annee_naissance' in df_BD.columns:
520     df_BD['annee_naissance'] = df_BD['annee_naissance'].apply(convertir_annees_naissance)
521     df_BD['annee_naissance'] = pd.to_numeric(df_BD['annee_naissance'], errors='coerce')
522     # Calcul de l'âge
523     annee_actuelle = date.today().year
524     df_BD['age'] = annee_actuelle - df_BD['annee_naissance']
525     # Tranche d'âge
526     df_BD["Tranche_Age"] = df_BD["age"].apply(get_tranche_age)
527
528 # Séparer 'DATE DE REMPLISSAGE' en 'DATE' et 'HEURE' pour df_BD (avec secondes)
529 if 'date_remplissage' in df_BD.columns:
530     df_BD['DATE'] = df_BD['date_remplissage'].apply(convertir_date)
531     df_BD['HEURE'] = df_BD['date_remplissage'].apply(convertir_heure)
532     # Supprimer la colonne
533     # Supprimer la colonne
534     df_BD.drop(columns=['date_remplissage'], inplace=True)
535
```

Interface code

```
662 # --- Création de la fenêtre ---
663
664 fenetre = tk.Tk()
665 fenetre.title("Traitement des fichiers CSV - Accident et BD")
666 fenetre.configure(bg=colors["bg"])
667
668 # Taille fenêtre
669
670 largeur, hauteur = 600, 500
671 largeur_ecran = fenetre.winfo_screenwidth()
672 hauteur_ecran = fenetre.winfo_screenheight()
673 x = (largeur_ecran // 2) - (largeur // 2)
674 y = (hauteur_ecran // 2) - (hauteur // 2)
675 fenetre.geometry(f"{largeur}x{hauteur}+{x}+{y}")
676
677
678 style_bouton = {
679     "bg": colors["card"],
680     "fg": colors["text"],
681     "activebackground": colors["accent"],
682     "activeforeground": colors["highlight"],
683     "font": ("Arial", 10, "bold"),
684     "relief": tk.FLAT,
685     "width": 40,
686     "padx": 5,
687     "pady": 5
688 }
689
690 # --- Boutons ---
691
692 btn_charger_excel = tk.Button(fenetre, text="Charger fichier Excel MAVIE", command=charger_fichier_excel, **style_bouton)
693 btn_charger_excel.pack(pady=5)
694
695 btn_traiter = tk.Button(fenetre, text="Lancer le traitement", command=lancer_traitement, **style_bouton)
696 btn_traiter.pack(pady=5)
697
```

I - Context

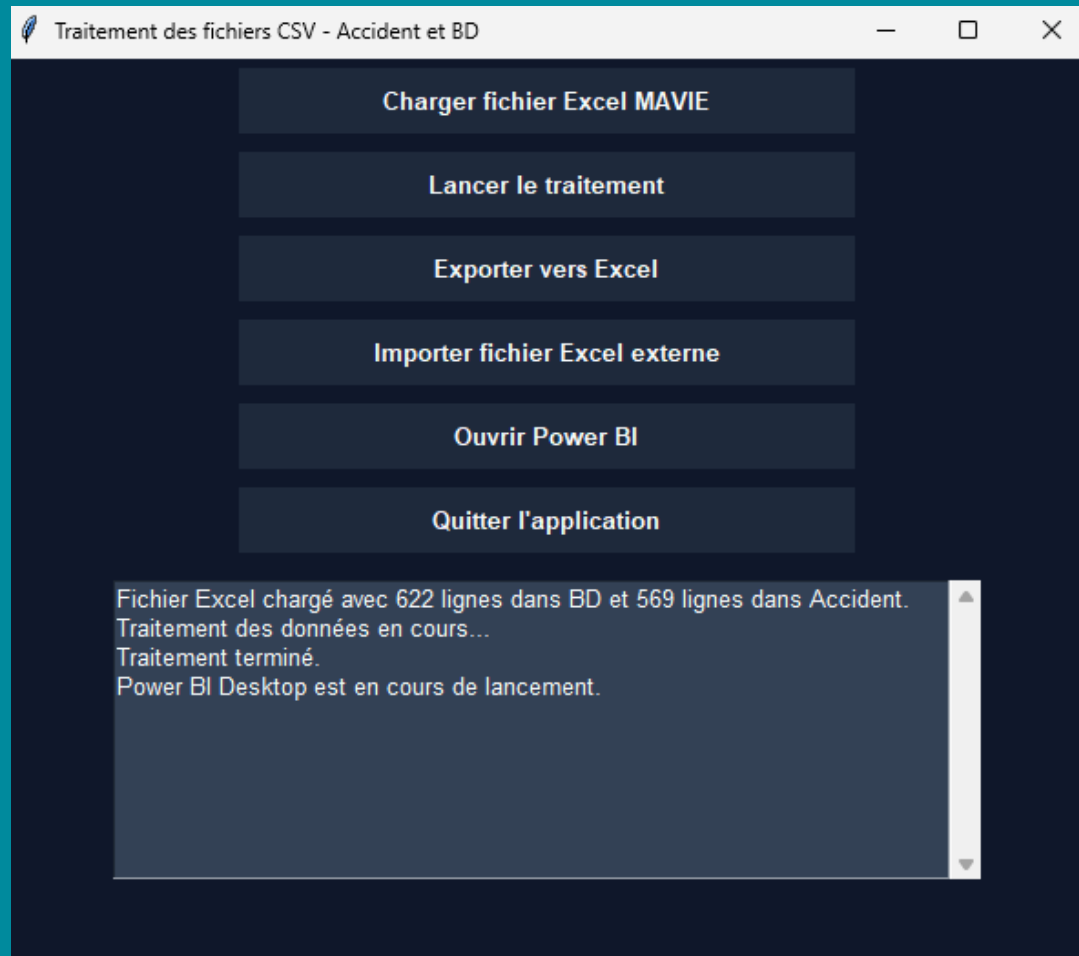
- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart



The interface

I - Context

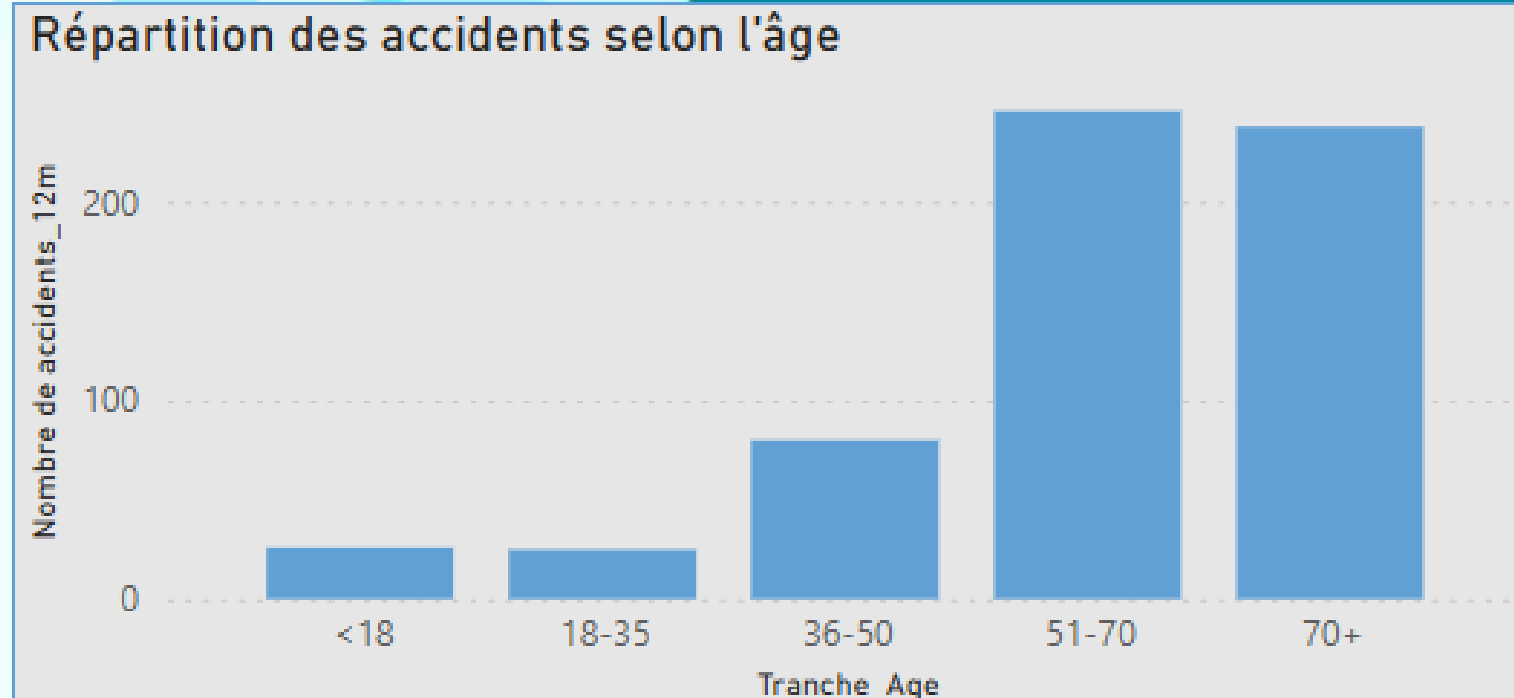
- Context and needs presentation
- For what purpose?
- Presentation of the dataset

II - Organization

- Organization and planning of work within the group
- Tools used

III - Project

- Presentation interface
- Chart





To conclude