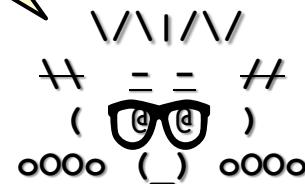


Version longue

# Exploration du Report Writing Interface de SAS®

**Code SAS  
inclus  
+ bonus !**



**Un secret bien gardé de l'Output Delivery System?**

0000  
0000 ( )  
( ) ) /  
\ ( ( /  
\ ) -  
-



Alain Voyer, 9 novembre 2022

**Club des utilisateurs  
SAS de Québec**

# Parcours proposé

- Qu'est-ce que le Report Writing Interface (RWI)?
  - et pourquoi?
- Exemple de base
- Le contexte orienté-objet
  - et la classe « odsout »
- Exemple de base expliqué
- Mise en garde
- Les différents types de méthodes
- Méthodes utilisées pour TABLE
- Syntaxe d'une méthode TEXT
- Méthodes utilisées pour LAYOUT
- LAYOUT() vs ODS LAYOUT
- Exemples: le résultat et le code SAS
- Conclusion, références, etc.

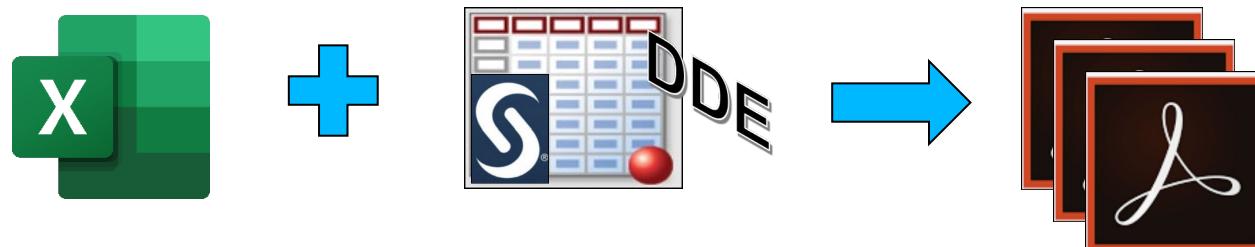
# Qu'est-ce que le RWI?

- Mécanisme avancé pour produire des rapports et documents **hautement personnalisés**
  - à partir des données disponibles dans une étape **DATA**
- Pleinement intégré dans **l'Output Delivery System**
  - avec une approche de programmation **Orientée-Objet**
- **Multiplie** les possibilités de personnalisation de l'approche traditionnelle avec "**DATA \_NULL\_**"
  - en tirant profit de tout ce qu'on utiliser dans une étape **DATA** comme la logique conditionnelle, le formatage, le traitement en sous-groupes, les vecteurs, etc.

# Pourquoi le RWI?

- Pour satisfaire aux exigences élevées
  - Dépassant les possibilités offertes notamment par
    - PROC REPORT
    - DATA \_NULL\_ (FILE PRINT ODS)
- Offrir un moyen de **convertir** des rapports produits par d'autres outils ou moyens comme
  - Excel, Word ou PowerPoint
  - Le Dynamic Data Exchange (**DDE**) – **Excel+SAS** → PDF

*“DDE is a legacy method, and SAS does not encourage using DDE”*



# Exemple de base...

- Liste des étudiant-e-s dans un document PDF

```
ods pdf file="C:\Temp\SAS\RWI\ExempleDeBase-ODS-RWI.pdf" notoc;

data _null_;
  set sashelp.class end=fin;
  if _n_=1 then
    declare odsout obj();

obj.format_text(data: "Nom: ",
                 data: Name, inline_attr: "color=blue",
                 data: "Genre: ",
                 data: Sex, inline_attr: "fontstyle=italic
                                         color=green");

if fin then
  obj.delete();

run;

ods pdf close;
```

	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69	112.5
2	Alice	F	13	56.5	84
3	Barbara	F	13	65.3	98
4	Carol	F	14	62.8	102.5

# Exemple de base

- Résultat :

```
obj.format_text(data: "Nom: ",  
                 data: Name, inline_attr: "color=blue",  
                 data: "Genre: ",  
                 data: Sex, inline_attr: "fontstyle=italic  
                                         color=green");
```

	Name	Sex	Age	Height	Weight
1	Alfred	M	14	69	112.5
2	Alice	F	13	56.5	84
3	Barbara	F	13	65.3	98
4	Cynthia	F	12	62.8	102.5
5	Henry	M	14	63.5	102.5
6	James	M	12	57.3	83
7	Jane	F	12	59.8	84.5
8	Janet	F	15	62.5	112.5
9	Jeffrey	M	13	62.5	84
10	John	M	12	59	99.5
11	Joyce	F	11	51.3	50.5

Table SAS sashelp.class

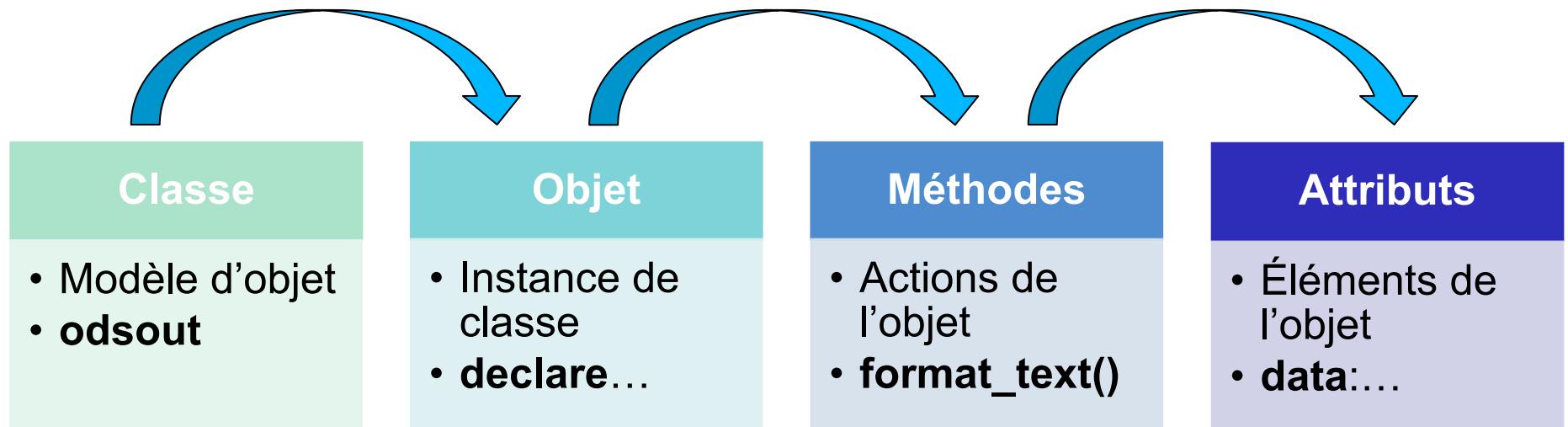
Nom: Alfred Genre: M  
Nom: Alice Genre: F  
Nom: Barbara Genre: F  
Nom: Cynthia Genre: F  
Nom: Henry Genre: M  
Nom: James Genre: M  
Nom: Jane Genre: F  
Nom: Janet Genre: F  
Nom: Jeffrey Genre: M  
Nom: John Genre: M  
Nom: Joyce Genre: F

Résultat en PDF

# Le contexte orienté-objet

- Quelques éléments de vocabulaire de l'orienté-objet:
  - Un objet est une entité composée d'un ensemble **d'attributs** (les éléments) et **d'actions** (les méthodes) à effectuer sur l'objet en question.
  - Une classe est un **modèle** ou un prototype à partir duquel des objets sont créés.
  - Une **instance** est un objet réel créé lors de l'exécution à partir d'une classe
  - Une **méthode** est l'équivalent d'une **fonction** dans le monde orienté-objet et ne s'applique qu'à l'objet qui a été défini par sa classe

# Le contexte orienté-objet



# La classe « odsout »

- Crédit d'une instance de la classe **odsout**:
  - Énoncé **DECLARE** (DCL) ou l'opérateur **\_NEW\_**

```
declare odsout obj(); /* une étape ou */  
declare odsout obj; /* deux étapes: 1 */  
obj = _new_(odsout); /* et 2 */ /*
```
  - Ici, "**obj**" c'est l'instance de la classe "**odsout**" qui est créé.  
"**obj**" c'est le nom que nous pouvons choisir.
  - Par convention le nom d'une instance est souvent composé de 1 à 3 caractères (32 caractères max.)
  - La création se fait habituellement une seule fois, généralement au début de l'étape DATA (**if \_n\_=1**)
  - Idéalement on doit aussi supprimer l'objet en fin de traitement avec la méthode **DELETE()**

# Exemple de base expliqué

- Liste des étudiant-e-s dans un document PDF
- Création (**instanciation**) de l'objet au début du traitement:

```
if _n_=1 then declare odsout obj();
```

- Utilisation de la méthode "**format\_text()**" et d'arguments:

```
obj.format_text(data: "Nom: ",  
                 data: Name, inline_attr: "color=blue",  
                 data: "Genre: ",  
                 data: Sex, inline_attr: "fontstyle=italic  
                                         color=green");
```

- "Ménage" optionnel à la fin du traitement:

```
if fin then obj.delete();
```

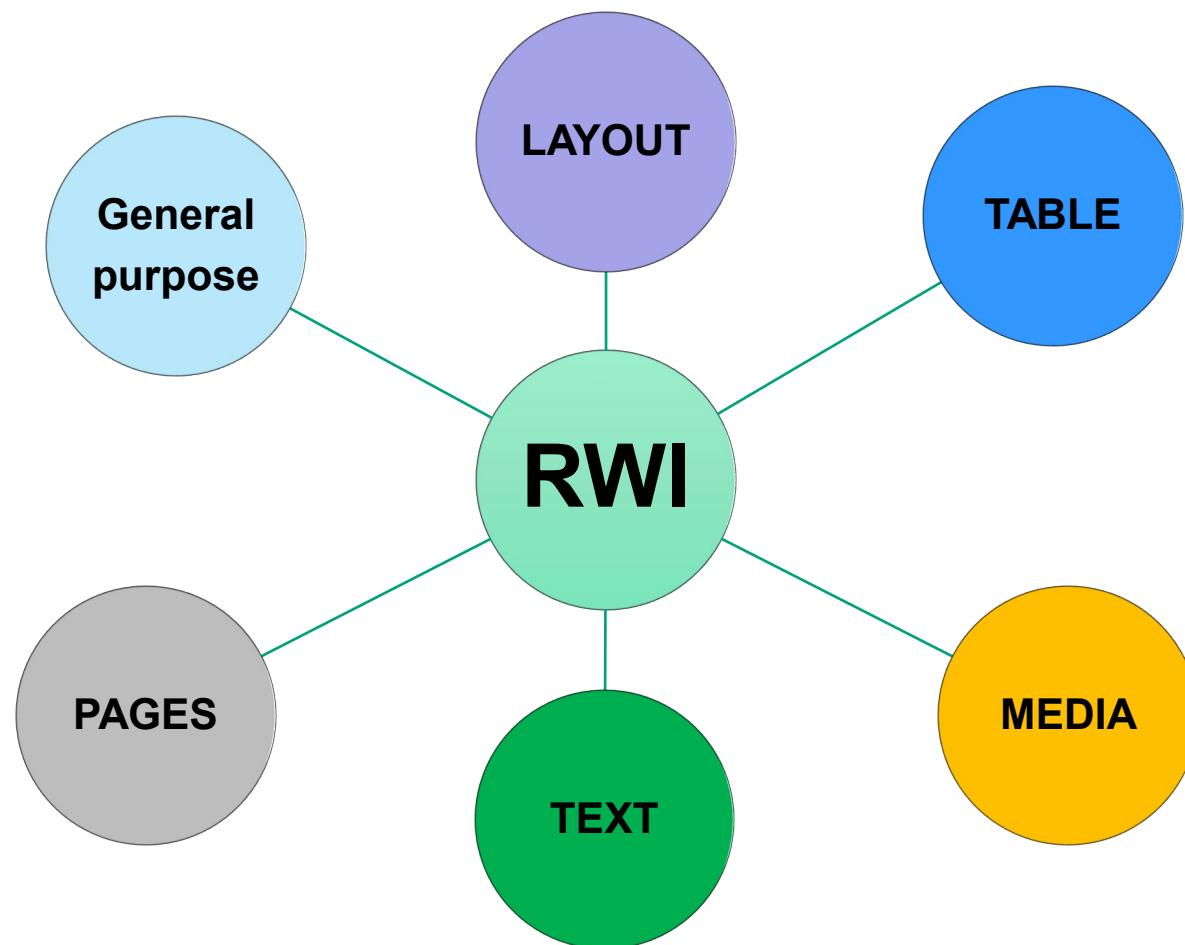


# Mise en garde



- Le **R**eport **W**riting **I**nterface de l'ODS s'appuie sur diverses fonctionnalités pour prendre pleinement en charge toutes les méthodes documentées. Cependant, **toutes les destinations de sortie ne prennent pas en charge toutes les fonctionnalités.**
- Comme toutes les destinations ne prennent pas en charge le même ensemble de fonctionnalités, **certaines méthodes peuvent avoir une prise en charge restreinte dans certaines destinations.**
- Au cours de son développement le **RWI** a progressivement changé et c'est pourquoi quelques méthodes et paramètres présents dans des articles des forums SAS habituels (SUGI, SAS Global Forum, forums régionaux, etc.) **ne se retrouvent pas dans la documentation officielle.**
  - Exemples:
  - L'argument **OVERRIDES** a changé pour **STYLE\_ATTR**
  - La méthode **OPEN\_DIR** n'est plus documentée.

# Les différents types de méthodes



# Les différents types de méthodes...

- On peut regrouper les méthodes en 6 différents types:



## Méthodes générales - **General purpose**

- Pour détruire une instance, créer un URL ou insérer une ligne dans la sortie
- Méthodes ***DELETE***, ***HREF*** et ***LINE***



## Méthodes de disposition ou d'agencement - **Layout**

- 2 variantes possibles: Absolute et Gridded
- Pour contrôler la disposition des éléments sur une page dans différentes régions
- Méthodes ***LAYOUT\_ABSOLUTE***, ***LAYOUT\_GRIDDED***, ***LAYOUT\_END*** et ***REGION***

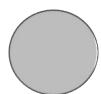


## Méthodes Média

- Permet d'utiliser des fichiers audios et vidéos ou d'insérer des images
- Méthodes ***AUDIO***, ***VIDEO*** et ***IMAGE***

# Les différents types de méthodes...

- On peut regrouper les méthodes en différents types:



## Méthodes Pages

- Pour contrôler les caractéristiques de l'entièreté d'une page
- Méthodes *PAGES*, *TITLE* et *FOOTNOTE*



## Méthodes Table

- Pour créer des sorties de forme tabulaire avec entêtes de rangées et de colonnes et les données dans un format structuré
- Méthodes de section de table: *BODY\_START*, *BODY\_END*, *FOOT\_START*, *FOOT\_END*, *HEAD\_START*, *HEAD\_END*
- Méthodes de table: *TABLE\_START*, *TABLE\_END*, *ROW\_START*, *ROW\_END*, *CELL\_START*, *CELL\_END* et *FORMAT\_CELL*



## Méthodes Texte - Text

- Pour créer du texte formaté et de haute qualité (fontes, couleurs, etc.)
- Méthodes ***FORMAT\_TEXT*** et ***NOTE***

# Les différents types de méthodes...

2 catégories de méthodes illustrées ici

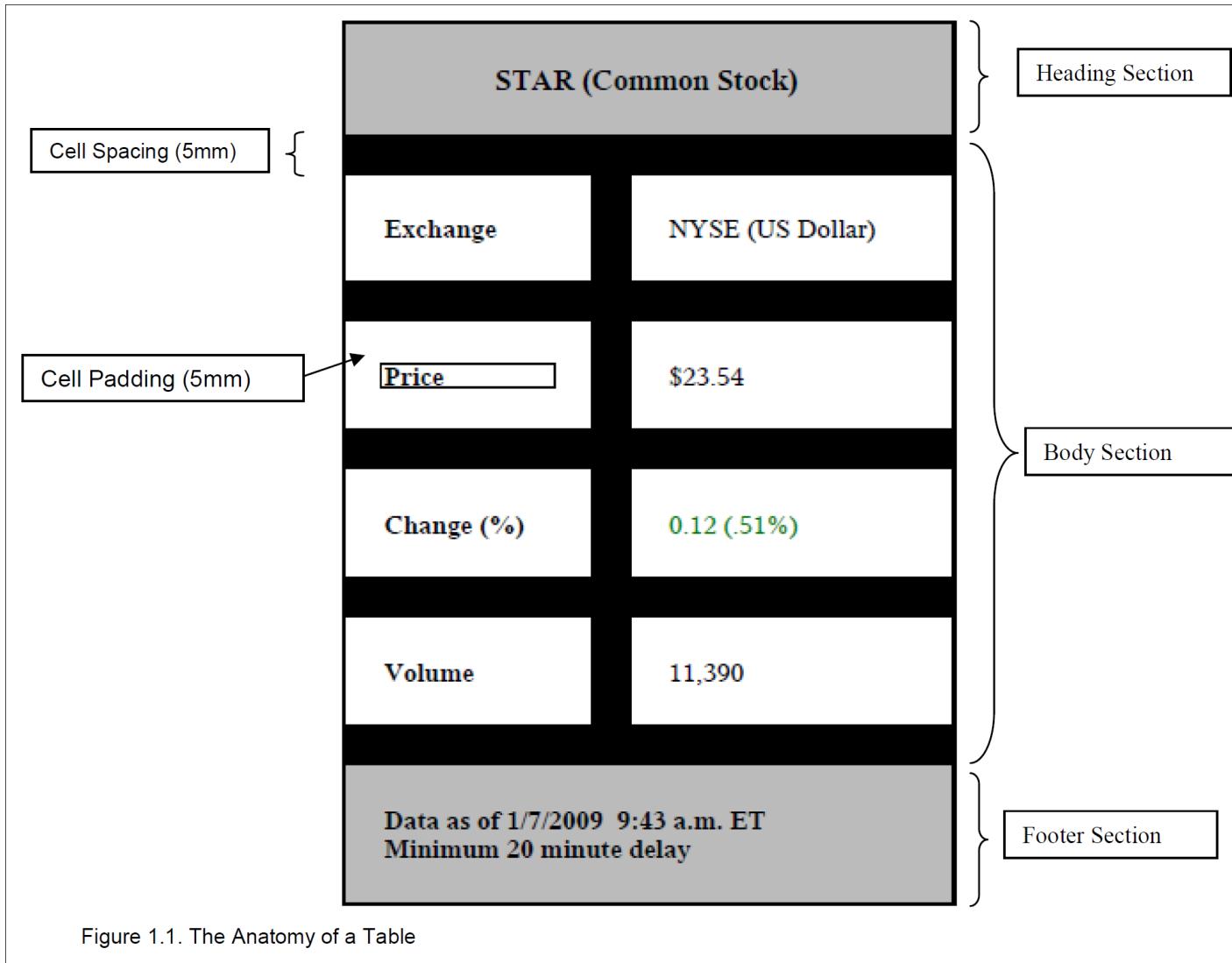
TABLE (sections)	TABLE (méthodes)	TEXT
<ul style="list-style-type: none"><li>• <code>head_start</code></li><li>• <code>head_end</code></li> <li>• <code>body_start</code></li><li>• <code>body_end</code></li> <li>• <code>foot_start</code></li><li>• <code>foot_end</code></li></ul>	<ul style="list-style-type: none"><li>• <code>table_start</code></li> <li>• <code>row_start</code></li> <li>• <code>format_cell</code></li><li>• <code>cell_start</code></li><li>• <code>cell_end</code></li> <li>• <code>row_end</code></li> <li>• <code>table_end</code></li></ul>	<ul style="list-style-type: none"><li>• <code>format_text</code></li> <li>• <code>note</code></li></ul>

# Méthodes utilisées pour TABLE



- Les **méthodes** de type **TABLE** créent une sortie bidimensionnelle composée d'en-têtes de lignes, d'en-têtes de colonnes et de valeurs de données dans un format structuré. Il est possible de créer cette sortie une seule cellule à la fois avec du formatage.
- On retrouve les notions suivantes concernant les méthodes utilisables:
  - **Cellule**: c'est un ensemble de données, de texte ou d'images pouvant occuper plusieurs lignes ou colonnes
  - **Remplissage de cellule (Cell Padding)**: épaisseur de chacun des 4 côtés d'une cellule
  - **Espacement des cellules (Cell Spacing)**: distance entre les cellules
  - **Dimension**: unité de mesure habituellement en CM ou en PO pour les différents éléments
  - **Colonne**: dimension verticale dans une structure tabulaire qui peut comprendre l'ensemble des cellules dans cette dimension
  - **Rangée**: dimension horizontale dans une structure tabulaire qui peut comprendre l'ensemble des cellules dans cette dimension
  - **Table**: arrangement de rangées et colonnes permettant que des catégories d'informations similaires soient affichées de manière structurée

# Méthodes utilisées pour TABLE



# Méthodes utilisées pour TABLE



**ORION STAR**  
Sports & Outdoors

STAR (Common Stock)	
<b>Exchange</b>	NYSE (US Dollar)
<b>Price</b>	\$23.54
<b>Change (%)</b>	0.12 (.51%)
<b>Volume</b>	11,390
Data as of 10/10/2022 9:43 a.m. ET Minimum 20 minute delay	

Provided to you compliments of SAS 9.4 using ODS RWI using Table methods.

# Syntaxe d'une méthode TEXT...



- Méthode **FORMAT\_TEXT()**
- **FORMAT\_TEXT**(DATA: données-à-afficher, <option1>, option2,...>);
  - Argument obligatoire:
    - **DATA**: < "chaîne" | nombre | var-car | var-num >
  - Arguments optionnels:
    - **FORMAT**: < var-car | "chaîne" >
    - **INLINE\_ATTR**: < var-car | "attribut(s)-de-style" >
    - **INLINE\_ELEM**: < var-car | "élément-de-style" >
    - **JUST**: "C" | "L" | "R" | var-car
    - **SPLIT**: < var-car | "chaîne" >
    - **STYLE\_ATTR**: < var-car | "attribut(s)-de-style" >
    - **STYLE\_ELEM**: < var-car | "élément-de-style" >
    - **VJUST**: < "B" | "M" | "T" | var-car >
    - **URL**: "chemin-URL"

# Syntaxe d'une méthode TEXT



- Option " **attribut(s)-de-style** "
  - de **INLINE\_ATTR** et de **STYLE\_ATTR**
- Dépendent de la destination:
  - **Excel, HTML, PowerPoint, PDF, RTF, Word...**
- Quelques attributs pour les **tables** en PDF:
  - **COLOR**=couleur
  - **BACKGROUNDCOLOR**=couleur
  - **FONTSIZE**=dimension (cm, in, mm, pt, etc.)
  - **FONTSTYLE=ITALIC | ROMAN | SLANT**
  - **FONTCWEIGHT=MEDIUM | BOLD | DEMI\_BOLD | LIGHT | etc.**
  - **HEIGHT**=dimension (cm, in, mm, pt, pct, px etc.)
  - **WIDTH**=dimension (cm, in, mm, pt, pct, px etc.)
  - Etc.

# Méthodes utilisées pour LAYOUT...



## 2 variantes possibles

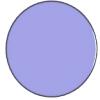
### LAYOUT (absolue)

- **layout\_absolute**
  - **region**
  - **layout\_end**

### LAYOUT (quadrillée)

- **layout\_gridded**
  - **region**
  - **layout\_end**

# Méthodes utilisées pour LAYOUT



- Les méthodes de type **LAYOUT** se déclinent en deux variantes:
  - **Absolute** et **Gridded** (absolue et quadrillée)
- Ces méthodes de mise en page contrôlent le **placement** de la sortie sur une page
- Un "**layout**" est une structure de contenu qui peut héberger des **régions**
- Une **région** peut contenir du matériel (du texte, des tables, des images) et imbriquer elle même des conteneurs **layout**
- On peut utiliser les **deux** variantes de **layout** conjointement
- La variante **Absolute** nécessite de la précision et on préfère généralement la variante **Gridded** plus souple

# LAYOUT() vs ODS LAYOUT

- LAYOUT se décline de 2 façons – Décryptage:
- La méthode **LAYOUT()** du Report Writing interface
  - Offre des méthodes à l'intérieur d'une étape DATA pour produire un rapport
  - Dispose des données présentes dans l'étape DATA
  - On ne peut pas inclure un énoncé **ODS LAYOUT** à l'intérieur d'une étape DATA
- L'énoncé **ODS LAYOUT**
  - Permet d'agencer plusieurs objets provenant de multiples sources
  - On peut inclure du contenu provenant du Report Writing Interface
- On peut considérer que le **ODS LAYOUT** offre de plus grandes possibilités dans la production de rapports personnalisés et sophistiqués
- À des besoins plus spécifiques, plus sophistiqués, il existe des moyens d'y répondre et l'**ODS LAYOUT** en est peut-être la réponse

# Exemple 1: le résultat



Club des utilisateurs  
SAS de Québec

## Notre organisation

 **Qui nous sommes...**

**Notre mission:**  
*Offrir des conférences périodiques autour de SAS tant au plan analytique qu'au plan du traitement des données*

**Notre vision:**  
*Ouvrir des perspectives d'utilisation qui peuvent inspirer les participant-e-s*

**Nos valeurs:**  
*Nous sommes guidés par l'accessibilité et orientés envers toutes les catégories d'utilisateurs-trices, du-de la débutant-e à l'expérimenté-e*

**Notre objectif:**  
*Nous visons l'acquisition de nouvelles compétences et le développement d'approches innovantes pour toutes et tous*

Créé en utilisant des méthodes TEXT du Report Writing Interface de l'ODS avec SAS 9.4

# Exemple 1: le code SAS...



```
* Cr ation du document avec l'objet FORMAT_TEXT() ;  
  
options nodate nonumber;  
  
options papersize=(6in 4in);  
  
ods escapechar="~";  
  
%let ChLogo='C:\Temp\SAS\RWI\LogoClubSAS.jpg';  
%let ChSmil='C:\Temp\SAS\RWI\Smiley.jpg';  
%let PDFfile=C:\Temp\SAS\RWI\ClubSAS.pdf;  
  
title1 "~~{style [preimage=&ChLogo width=100pct  
background=cx000080 color=cx87CEEB fontsize=16pt just=left]  
Notre organisation}";  
  
footnote1 "~~{style [fontsize=8pt just=center color=cx000080]  
Cr   en utilisant des m thodes TEXT du Report  
Writing Interface de l`ODS avec SAS 9.4}";
```

Cette partie de code ne fait pas appel au RWI  
mais utilise le « inline formatting »

# Exemple 1: le code SAS

```

ods pdf file="&PDFfile" notoc;

data _null_;
  set ClubSAS end=fin;
  if (_n_=1) then do;
    declare odsout O();
    O.note(data: "Qui nous sommes...",
           style_attr: "preimage=&ChSmyl
                         fontstyle=italic fontsize=10pt fontweight=bold
                         color=cx000080", just: "L");
  end;

O.format_text(data: slogan,
              style_attr: "background=cx000080 color=cx87CEEB fontsize=10pt
                            fontstyle=italic width=2.5in", just: "C");
O.format_text(data: description,
              style_attr: "background=cx87CEEB fontstyle=italic fontsize=8pt
                            width=2.5in", just: "C");

if not fin then O.format_text(data: " ", style_attr: "height=1mm");
if      fin then O.delete();
run;

ods _all_ close;

```

	▲ slogan	▲ description
1	Notre mission:	Offrir des conférenc...
2	Notre vision:	Ouvrir des perspecti...

# Exemple 1: le résultat bis

TEXT

note()  
Club des utilisateurs  
SAS de Québec

## Notre organisation

title

Qui nous sommes...

format\_text()

Notre mission:  
Offrir des conférences périodiques autour de SAS  
tant au plan analytique qu'au plan du traitement  
des données

Notre vision:  
Ouvrir des perspectives d'utilisation qui peuvent  
inspirer les participant-e-s

Nos valeurs:  
Nous sommes guidés par l'accessibilité et  
orientés envers toutes les catégories  
d'utilisateurs-trices, du-de la débutant-e à  
l'expérimenté-e

Notre objectif:  
Nous visons l'acquisition de nouvelles  
compétences et le développement d'approches  
innovantes pour toutes et tous

footnote

Créé en utilisant des méthodes TEXT du Report Writing Interface de l'ODS avec SAS 9.4

# Exemple 2: le résultat



## Nos présentateurs



Jean Hardy est une des personnes les mieux connues des utilisateurs SAS du Québec. Cumulant plus de 33 ans d'expérience en services conseils et en formations sur différents logiciels statistiques, il persiste à partager sa passion et ses connaissances à toutes celles et ceux intéressés par les mêmes sujets. Curieux et rigoureux, les échanges avec Jean sont toujours sources de nouvelles idées et solutions. Membre du CA du Club, il fait partie des membres fondateurs des quelques générations des Clubs SAS de Québec au fil des années.



Toujours de bonne humeur, Alain a fait carrière comme analyste en informatique spécialisé en infrastructure technologique de stockage et logiciels. Il a été responsable de l'installation, de l'implantation et du support technique de plusieurs logiciels incluant SAS sur plusieurs plateformes. Il a conçu, entretenu, et optimisé plusieurs traitements SAS au fil des années dans son rôle d'analyste en stockage. Mais c'est d'abord son rôle de support conseil aux utilisateurs qui a stimulé sa passion envers SAS. Présentateur occasionnel dans des groupes d'utilisateurs SAS, il poursuit sa carrière comme formateur SAS depuis 2017 heureux de partager ses connaissances et de continuer à apprendre en même temps.

**Le Club des utilisateurs SAS de Québec: actif dans le partage des connaissances du monde SAS**

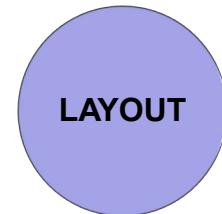
# Exemple 2: le code SAS...



```
* Cr ation du document avec l'objet LAYOUT_GRIDDED() ;  
  
%let chemin=C:\Temp\SAS\RWI;  
%let PDFfile=C:\Temp\SAS\RWI\Gridded_Layout-ClubSAS.pdf;  
  
options nodate nonumber;  
options papersize=(8in 6in);  
  
ods escapechar="~";  
  
filename flogo "&chemin\LogoClubSASFb.jpg";  
  
title1 "~~{style [preimage=flogo width=100pct background=lightgray color=darkblue  
fontsize=18pt just=left fontstyle=italic] Nos pr sentateurs}";  
  
footnote1 "~~{style [fontsize=10pt fontstyle=italic just=r color=darkblue  
backgroundcolor=yellow width=100pct] Le Club des utilisateurs SAS de  
Qu bec: actif dans le partage des connaissances du monde SAS}";
```

Cette partie de code ne fait pas appel au RWI  
mais utilise le « inline formatting »

# Exemple 2: le code SAS



```
ods pdf file="&PDFfile" notoc;

data _null_;
  set formateurs end=fin;
  if (_n_=1) then do;
    dcl odsout O();
    O.layout_gridded(columns: 3, style_attr: "just=left");
  end;
  O.region(width: "1.25in");
  O.image(file: photo, style_attr: "just=left width=25pct");
  O.format_text(data: nom, style_attr: "just=right width=100pct fontsize=10pt
                                             fontweight=bold");
  O.region(width: "0.75in");
  O.format_text(data: " ");
  O.region(width: "4.75in");
  O.format_text(data: strip(bio), style_attr: "fontweight=bold");

  if fin
    then do; O.layout_end();
            O.delete();
    end;
  else O.format_text(data: "") ;
;
run;
ods _all_ close;
```

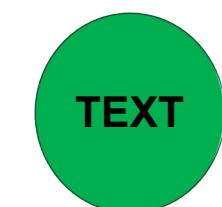
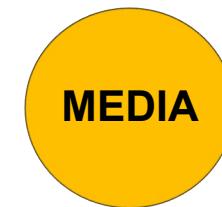


	photo	nom	bio
1	C:\Temp\SAS\RWI\JeanHardy.p...	Jean Har...	Jean Hardy est une des personnes ...
2	C:\Temp\SAS\RWI\AlainVoyer.p...	Alain Voy...	Toujours de bonne humeur, Alain ...

# Exemple 2: le résultat bis

LAYOUT



## Club des utilisateurs SAS de Québec

### Nos présentateurs

**image()**



Jean Hardy

**Format\_text()**

Jean Hardy est une des personnes les mieux connues des utilisateurs SAS du Québec. Cumulant plus de 33 ans d'expérience en services conseils et en formations sur différents logiciels statistiques, il persiste à partager sa passion et ses connaissances à toutes celles et ceux intéressés par les mêmes sujets. Curieux et rigoureux, les échanges avec Jean sont toujours sources de nouvelles idées et solutions. Membre du CA du Club, il fait partie des membres fondateurs des quelques générations des Clubs SAS de Québec au fil des années.

**region()** **region()** **region()**

**title**

**layout\_gridded()**

**footnote**

*Le Club des utilisateurs SAS de Québec: actif dans le partage des connaissances du monde SAS*

# Conclusion et perspective

- Le RWI est un **immense** terrain de jeu dont les possibilités sont extrêmement nombreuses mais avec certaines limites
- Le RWI offre une très **grande souplesse** pour la personnalisation de rapports ou documents
- Nécessite **exploration et expérimentation** intensives
  - Syntaxe élaborée
  - Nombreux paramètres et méthodes
  - Combinaisons multiples des méthodes
- Il faut **consulter souvent** la documentation
- **D'autres zones** d'exploration sont possibles:
  - Utilisation de **PROC TEMPLATE** (attributs de style)
  - Utilisation des « Cascading Style Sheets » (CSS)
  - L'énoncé **ODS LAYOUT** en complément

# Quelques références

[SAS® 9.4 Output Delivery System: Advanced Topics, Third Edition](#)

[ODS Report Writing Interface Tip Sheet](#)

[Take Home the ODS Crown Jewels: Master the New Production Features of  
ODS LAYOUT and Report Writing Interface Techniques](#)

Daniel O'Connor, SAS Institute Inc., SAS Global Forum 2013, Paper 015-2013

[The Power to Show: Ad Hoc Reporting, Custom Invoices, and Form Letters](#)

Daniel O'Connor, SAS Institute Inc., WUSS 2009

[Next Generation DATA \\_NULL\\_ Report Writing using ODS OO Features:](#)

[A Powerful Tool to Create Custom Tables](#)

Alice Liqiong Li and Jacob Jen-Hao Cheng

Maryland Hospital Association - Quality Indicator Project, Elkridge MD

SAS Global Forum 2010, Paper 072-2010

[Toe to Toe: Comparing ODS LAYOUT and the ODS Report Writing Interface](#)

Daniel Kummer, SAS Institute Inc., Cary, NC, SUGI31, Paper 330

[Have it Your Way: Creating Reports with the Data Step Report Writing Interface](#)

Pete Lund, Looking Glass Analytics, Olympia, WA

SAS Global Forum 2016, Paper 2240-2016

[Have it Your Way: Using the ODS EXCEL Destination with the DATA step Report Writing Interface](#)

Pete Lund, Looking Glass Analytics, Olympia, WA, WUSS 2019, Paper 210

[Using the DATA Step to Create Bar Charts: The ODS Report Writing Interface](#)

Randy Herbison, Westat, NESUG 10, Paper BB05

**Questions ?  
Commentaires ?  
Suggestions ?**

**Merci!**

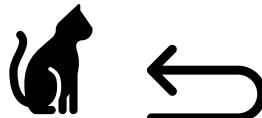


## **Exploration du Report Writing Interface de SAS®**

**Contact:**

```
1 data _null_;  
2 courriel=lowcase(cat(reverse("Alain") ,reverse("Voyer") , "@gmail.com")) ;  
3 putlog "NOTE: " courriel=;  
4 run;
```

NOTE: courriel=nialareyov@gmail.com



**Club des utilisateurs  
SAS de Québec**

### **Marques de commerce**

SAS et tous les autres noms de produits et services de SAS Institute Inc. sont des marques de commerce de SAS Institute Inc. aux États-Unis et dans les autres pays. ® indique l'inscription aux États-Unis. Les autres noms de produits ou marques sont des marques déposées ou des marques de commerce de leurs compagnies respectives.

# Noms des couleurs

White	Cornsilk	Antiquewhite	Seashell	Linen	Ivory	Floralwhite
Snow	Azure	Mintcream	Ghostwhite	Honeydew	Aliceblue	Beige
Oldlace	Bisque	Moccasin	Wheat	Navajowhite	Blanchedalmond	Tan
Gray	Lightgrey	Darkgray	Dimgray	Gainsboro	Silver	Whitesmoke
Black	Darkslategray	Slategray	Lightslategray	Lemonchiffon	Khaki	Darkkhaki
Brown	Sienna	Chocolate	Saddlebrown	Sandybrown	Burlywood	Peru
Red	Tomato	Darkred	Indianred	Mistyrose	Lavenderblush	Firebrick
Crimson	Maroon	Peachpuff	Goldenrod	Darkgoldenrod	Palegoldenrod	Lavender
Orange	Darkorange	Orangered	Forestgreen	Greenyellow	Lime	Lightgoldenrodyellow
Yellow	Lightyellow	Gold	Springgreen	Darkolivegreen	Olive	Limegreen
Green	Lightgreen	Darkgreen	Mediumseagreen	Mediumspringgreen	Palegreen	Olivedrab
Lawngreen	Chartreuse	Yellowgreen	Paleturquoise	Darkseagreen	Aquamarine	Mediumaquamarine
Teal	Lightseagreen	Seagreen	Darkblue	Mediumturquoise	Turquoise	Darkturquoise
Darkcyan	Cyan	Lightcyan	Mediumslateblue	Lightskyblue	Skyblue	Deepskyblue
Blue	Lightblue	Mediumblue	Steelblue	Darkslateblue	Powderblue	Cornflowerblue
Royalblue	Dodgerblue	Slateblue	Plum	Cadetblue	Mediumorchid	Darkorchid
Navy	Midnightblue	Lightsteelblue	Mediumvioletred	Orchid	Thistle	Rosybrown
Purple	Mediumpurple	Indigo	Fuchsia	Palevioletred	Magenta	Darkmagenta
Violet	Darkviolet	Blueviolet	Salmon	Deeppink	Coral	Lightcoral
Pink	Lightpink	Hotpink	Lightsalmon	Darksalmon		

# Hommage aux femmes en informatique

- **Ada Lovelace (10 décembre 1815 – 27 novembre 1852)**
- Augusta Ada King, comtesse de Lovelace était une **mathématicienne** et **écrivaine** anglaise, principalement connue pour son travail sur l'ordinateur mécanique polyvalent proposé par Charles Babbage: **le moteur analytique**
- Considérée comme la première femme "**programmeuse**"
- Le langage **ADA** a été créé en son honneur

