

RAPPORT DE TEL PROJET

Élèves 1A

Titre choisi pour mon rapport en Quarto

Étudiant-e-s :
Blanche Neige
Peter Pan
Mickey Mouse
Elsa d'Arendelle

Responsable :
la Fée bleue

Encadrant(e) :
Jiminy Cricket

Table des matières

# Titre de premier niveau	3
## Titre de deuxième niveau	3
## Autre titre de deuxième niveau	3
### Exemple	3
### Dernier exemple	3
Utilité de Quarto et utilisation	4
Mix entre texte et code intégré	4
Quelques conventions propres à Quarto pour la typographie	4
Affichage de caractères spéciaux	4
Gras et italique	5
Parler « code »	5
Puce	5
Tiret	5
Présentation des paragraphes	5
Passage à la ligne	5
Création d'un nouveau paragraphe	5
Renvois	5
Notes de bas de page	5
Renvois vers une autre section du rapport	6
Utilisation des références	6
Références	8
Annexes	9
Résumé	10

Pour faire apparaître cette page, il faut saisir la ligne « `\tableofcontents` » à l'endroit souhaité. Par défaut, la page est appelée « Table des matières », si on veut la renommer « Sommaire », on fait précéder la ligne précédente par la ligne « `\renewcommand{\contentsname}{Sommaire}` ».

→ Quelle différence entre un sommaire et une table des matières ?

Généralement, un sommaire est plus ou moins détaillé (parfois juste avec les grandes parties et sans numéro de page) et toujours en début de document quand une table des matières, plus souvent en fin de document, rentre dans les détails de chaque section toujours liée à la page où elle débute. Dans les faits, les deux termes sont souvent employés l'un pour l'autre, surtout quand c'est détaillé (ce qui est plus habituel d'une table des matières) tout en étant à la fois en début de document (ce qui est plus habituel d'un sommaire). Étant ce qui est attendu ici, le nom de cette page sera à la convenance personnelle des rédacteurs du rapport.

Dans le PDF généré en ligne, les titres sont cliquables et renvoient aux sections concernées plus bas dans le document.

Attention : contrairement aux logiciels de traitement de texte les plus couramment utilisés (comme Microsoft Word ou Libre Office Writer), la page de titre n'est pas prise en compte pour la numérotation des

pages (un peu comme s'il s'agissait de la couverture d'un livre). Pour ça, pour ceux qui connaissent, c'est pareil que pour un document tapé avec LaTeX sur Overleaf. La page 1 est la première page après la page de titre soit ici la table des matières.

Titre de premier niveau

Titre de deuxième niveau

Un titre de premier niveau est introduit dans le langage Quarto par 1 croisillon (#), un titre de deuxième niveau par 2 croisillons (##), de troisième niveau par 3 croisillons (###), etc. Attention, seuls les titres des trois premiers niveaux paraissent dans la table des matières ; pas les titres de quatrième niveau ou au-delà.

Il faut un espace entre les croisillons et le titre choisi de la section.

Autre titre de deuxième niveau

Tulaliloo ti amo, chika banana ! Minion power, kaboom finale !

Exemple

Bee-do bee-do ! Fire in the pancako ! To the rescue-o, pronto-pronto !

Exemple dans l'exemple

Bananaaaa... oh no, le banana kaput ! Omelette du fromage... plan B.

Autre exemple

Papoy boom-boom! Hahaha, tutu balala ! Bello ! Me want banana gelato subito !

Dernier exemple

Tank yu veddy mucho, boss man ! Me zip-zap finisho la missiono !

Utilité de Quarto et utilisation

Un document Quarto sur R vous permet de mêler du texte et du code exécutable pour tout regrouper dans un seul document finalisé, comme ici un rapport en format PDF. Pour en savoir plus sur Quarto : site officiel <https://quarto.org> (en anglais). Pour faire un lien cliquable sur une page externe (comme dans la phrase précédente), il faut encadrer le lien par des chevrons simples (« < » et « > » respectivement avant et après le lien).

L'avantage par rapport aux autres logiciels de traitement de texte est que cela permet d'intégrer du code (ici R) directement dans le document et d'afficher les résultats (graphiques, tableaux, etc.) automatiquement. Cela favorise une reproductibilité scientifique : on peut voir et exécuter le code à l'origine des résultats.

Les résultats sont ainsi générés dynamiquement à partir des données. Si les données changent, il suffit de recompiler le document : tout se met à jour automatiquement (textes, graphiques, analyses...). Cela évite les erreurs de copier-coller ou les résultats obsolètes.

Mix entre texte et code intégré

Lorsqu'on clique sur le bouton **Render** de RStudio (ou combinaisons des touches Ctrl+Maj+K), un PDF est généré incluant à la fois le texte et le code intégré. On peut intégrer du code comme suit en l'encadrant entre une première ligne « `{r}` » et une dernière « `}` » (« ``` » est le symbole accent grave, appelé aussi *backtick* ou *backquote* depuis ses noms en anglais).

```
1 + 1
```

```
[1] 2
```

Ces sections de code intégré au texte et exécutable sont appelées des *chunks*. Quand on clique sur le bouton **+c** de RStudio (ou Ctrl+Maj+I), un *chunk* est automatiquement rajouté après l'emplacement du curseur.

Vous pouvez aussi ajouter des options au code exécutable comme ceci (derrière « `#|` », c'est-à-dire un croisillon et une barre verticale) :

```
[1] 4
```

L'option « `echo: false` » désactive l'impression du code (seul le résultat final est affiché).

Quelques conventions propres à Quarto pour la typographie

Affichage de caractères spéciaux

Points typographiques : pour que le symbole accent grave ne soit pas compris par R comme du code dans du texte, on doit l'entourer de chaque côté par deux accents graves (cf. le code de la section précédente : si on est actuellement dans le mode **Visual**, pour regarder ce code, cliquer sur le bouton **Source** de RStudio ou Ctrl+Maj+F4). Pour les autres caractères spéciaux (« `*` », « `\` », « `#` », « `<` »... liste non exhaustive), c'est la barre oblique inversée (« `\` ») qui sert à désactiver la fonction spéciale habituelle de ces caractères en Quarto en la mettant juste devant chacun d'eux.

Gras et italique

Les astérisques (« * ») servent à mettre en gras ou en italique du texte. En encadrant avec un astérisque de chaque côté, on met **ce texte en italique** ; avec deux astérisques, on met ****ce texte en gras**** ; avec trois, on le met à la fois ******en gras et en italique******. Pour ceux qui connaissent, cela utilise sur ce point les mêmes conventions que la plateforme Discord.

Parler « code »

Lorsque l'on encadre une portion de texte avec un seul accent grave de chaque côté, ce qui est encadré est affiché dans une police à chasse fixe, c'est-à-dire une police où chaque caractère occupe le même espace horizontal que les autres caractères. C'est, dans les textes parlant informatique, un type de police souvent utilisé pour représenter le contenu du code utilisé. Exemples :

- si on tape « ` Machin ` » dans le code, cela affiche « Machin » où le M majuscule n'est pas plus large ou le i minuscule pas moins large que les autres lettres ;
- alors que si on tape « Machin », ça affiche « Machin » dans la police choisie en début de document, idéalement dans une police où la chasse est différente pour chaque caractère.

Puce

Comme on peut le voir dans les deux exemples ci-dessus, dans le code, un saut de ligne suivi d'une ligne commençant pour un trait d'union est retranscrit par une ligne commençant par une puce.

Tiret

– Si on veut réellement commencer la ligne par un tiret, il faut entamer la ligne par deux traits d'union comme pour cette phrase.

Présentation des paragraphes

Passage à la ligne

Pour passer à la ligne sans créer de nouveau paragraphe, terminer une ligne par deux espaces et appuyer sur « Entrée » afin d'écrire la ligne suivante.

Comme ici : on peut le vérifier dans le code source en naviguant entre les deux lignes avec les flèches du clavier.

Création d'un nouveau paragraphe

Le plus simple (car le plus lisible) est de sauter une ligne en laissant une ligne vide.

Si on passe à la ligne sans sauter de ligne et sans les deux espaces à la ligne précédente, cela écrira tout à la suite. Comme ce qui est écrit ici.

Renvois

Notes de bas de page

Pour faire une note de bas de page, il faut mettre à l'endroit souhaité un nom en chiffres ou en lettres précédé d'un accent circonflexe entre crochets comme ceci : « [^5] » et cela donne ¹ puis dans un paragraphe suivant, on fait le rappel de la note de la même façon avec le texte associé après deux-points.

Test avec « [^a] »².

Test avec « [^bidule] »³.

¹Voici le contenu de la note de bas de page qui apparaîtra en bas de la même page.

²Ça marche aussi avec une lettre.

³Ça marche aussi avec une note qui porte un nom.

Test avec « `[^fan2foot]` »⁴.

Comme on le voit, le document PDF généré numérote automatiquement les notes de page quel que soit le nom de la note. Et si on passe à la page suivante, la numérotation continue, on ne recommence pas à 1.

Renvois vers une autre section du rapport

Voici un lien interne vers la [page des références](#). Le lien interne est écrit entre crochets tout de suite suivi (sans espace) entre parenthèses d'un mot-clef (mot précédé d'un croisillon) : « `[page des références] (#pagedesreferences)` ». Ce mot-clef est répété à la section souhaitée, en l'occurrence derrière le titre de la section Références et entre accolades : « `{#pagedesreferences}` ».

On peut aussi mettre ce lien interne dans une note de bas de page comme ceci⁵.

Dans l'exemple ci-dessus, c'est un cas très utile de renvoi vers les références. D'autres cas qui peuvent être très courants : renvoi vers les annexes ou tout bonnement vers une section particulière du rapport (pour comparer deux résultats notamment). Il faudra alors créer autant de mot-clefs entre accolades tout au long du texte pour chaque endroit vers lequel l'on devra pointer.

Utilisation des références

Dans l'en-tête YAML de ce document, on a « `bibliography: references.bib` » qui va servir à faire une liste pour la bibliographie. Pour faciliter son utilisation, cette liste va être rédigée dans un fichier annexe appelé « `references.bib` ». Pour créer ce fichier, aller dans le menu de R : `File` → `New File` → `Text File` ; une fois ouvert, sauvegarder le fichier sous le nom souhaité.

Pour l'exemple, voici un extrait le contenu du fichier « `references.bib` » (extrait car la partie centrale remplacée par `[...]`) :

```
@article{calhamer1974,
  author    = {Allan B. Calhamer},
  title     = {The Invention of Diplomacy},
  journal   = {Games and Puzzles},
  year      = {1974},
  number    = {21}
}

@book{sharp1978,
  author    = {Richard Sharp},
  title     = {The Game of Diplomacy},
  year      = {1978},
  publisher = {Arthur Baker}
}

[...]

@book{kostick2015,
  author    = {Conor Kostick},
  title     = {The Art of Correspondence in the Game of Diplomacy},
  year      = {2015},
  publisher = {Curses & Magic}
}

@book{vanmechelen2019,
  author    = {Erik van Mechelen},
  title     = {How to Be a Diplomacy Player: A Philosophy of Face to Face Diplomacy},
```

⁴Ça marche aussi avec une note qui porte un nom qui mixe chiffres et lettres.

⁵Pour plus d'informations, se référer à la [page des références](#).

```
year      = {2019},
publisher = {Lean Publishing}
}
```

On peut voir dans la [page des références](#), la présentation finale. On avait choisi de lister les références dans `references.bib` dans l'ordre chronologique ; attention, par défaut, cela les affiche dans le PDF dans l'ordre alphabétique des auteurs, cela ne respecte pas l'ordre initialement choisi (si on avait choisi de les ranger d'une certaine façon).

Toujours dans l'en-tête YAML, « `nocite: "@*"` » permet de lister dans le PDF la totalité des références mentionnées dans `references.bib`. Sans cette ligne, seuls les éléments de la bibliographie listés simultanément dans `references.bib` et cités dans le corps du document seront listés dans le PDF.

Pour citer un élément de la bibliographie dans le corps du document, il faut faire précéder le mot d'appel à ce document précédé d'un arobase et le tout entre crochets. Le mot d'appel est le terme composé des caractères juste derrière l'accolade du document dans `references.bib` donc ici « `calhamer1974` » pour l'article d'Allan B. Calhamer publié en 1974, « `sharp19784` » pour le livre de Richard Sharp édité en 1978, « `kostick2015` » pour celui de Conor Kostick en 2015 ou encore « `vanmechelen2019` » pour celui d'Erik van Mechelen en 2019.

Cet appel dans le corps du texte sera cliquable et renverra à la ligne de la publication dans le PDF si on choisit cette option dans l'en-tête YAML : « `link-citations: true` ».

Exemples avec « `[@calhamer1974]` » et « `[@kostick2015]` » :

Allan B. Calhamer, l'auteur américain du jeu de société *Diplomatie*, célèbre jeu basé sur les talents de stratège et de négociateur des participants et conçu dans les années 1950, a écrit deux décennies plus tard dans la presse un article très développé expliquant le processus l'ayant amené à le créer ([Calhamer 1974](#)).

Conor Kostick, historien irlandais et champion de ce jeu, a rédigé en 2015 *The Art of Correspondence in the Game of Diplomacy*, un ouvrage avec de nombreux conseils pour mieux gérer les phases de négociation à *Diplomatie* ([Kostick 2015](#)).

Références

- Brassinne, Michel. 1981. « Les jeux d'alliance ». *Jeux et Stratégie*, n° 8: 59-63.
- Calhamer, Allan B. 1974. « The Invention of Diplomacy ». *Games and Puzzles*, n° 21.
- . 1999. *Calhamer on Diplomacy: The Boardgame Diplomacy and Diplomatic History*. AuthorHouse.
- Eskin, Blake. 2004. « World Domination: The Game ». *The Washington Post*, W16.
- Hill, David. 2014. « The Board Game of the Alpha Nerds ». *Grantland*.
- Kostick, Conor. 2015. *The Art of Correspondence in the Game of Diplomacy*. Curses & Magic.
- McClellan, Joseph. 1986. « Lying and Cheating by the Rules ». *The Washington Post*.
- McClelland, Edward. 2009. « All in the Game ». *Chicago*.
- Mechelen, Erik van. 2019. *How to Be a Diplomacy Player: A Philosophy of Face to Face Diplomacy*. Lean Publishing.
- Sharp, Richard. 1978. *The Game of Diplomacy*. Arthur Baker.
- Walker, Rod. 1979. *The Gamer's Guide to Diplomacy*. The Avalon Hill Co.

Annexes

Résumé

Dans certains projets, il pourra être demandé de rédiger un résumé.