Utiliser des sous-programmes pour  
 organiser votre code

**L’algèbre**

**Unité 3, Fiche 8a**

Partie 1 : Une application qui calcule l’aire d’un rectangle

Revoyons l’application qui a déterminé l’aire d’un rectangle et réorganisons le code en créant des sous-programmes.

Un ***sous-programme*** contient un petit ensemble d’instructions pour accomplir une tâche et est référencé et activé par le programme principal. Lorsque nous créons un nouveau bloc dans Scratch, nous créons un sous-programme.

Retrouvez le lien de l’application que vous avez utilisée pour déterminer les aires des rectangles de la Fiche 7 :

<https://scratch.mit.edu/projects/1194399829/editor/>

Si vous avez un compte Scratch, sauvegardez le projet dans votre compte en sélectionnant Remix en haut de l’écran.

Une connexion n’est pas nécessaire pour travailler avec le code, mais vous ne pourrez pas sauvegarder vos modifications sans elle.

Ce pseudo-code montre comment nous allons réorganiser l’application en 3 sous-programmes et un programme principal qui les appelle.

*Sous-programme Obtenir les données d’entrée*

|  |
| --- |
| sous-programme obtenirDonnéesEntrée  sortie « Entre la longueur du rectangle en centimètres : »  **longueur** = entrée de l’utilisateur  sortie « Entre la largeur du rectangle en centimètres : »  **largeur** = entrée de l’utilisateur |

*Sous-programme Calculer l’aire*

|  |
| --- |
| sous-programme calculerAire  **aire** = **longueur** \* **largeur** |

Utiliser des sous-programmes pour  
 organiser votre code (suite)

**L’algèbre**

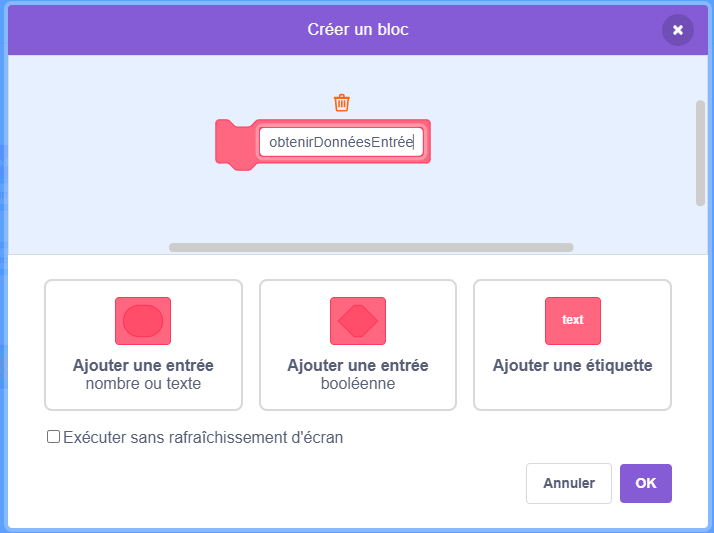
**Unité 3, Fiche 8b**

*Sous-programme Sortie de résultats*

|  |
| --- |
| sous-programme sortieRésultats  sortie « L’aire est de “ **aire** “ centimètres carrés. » |

*Programme principal*

|  |
| --- |
| sortie « Je vais calculer l’aire de ton rectangle. »  exécute le sous-programme obtenirDonnéesEntrée  exécute le sous-programme calculerAire  exécute le sous-programme sortieRésultats |

1.  Commencez par créer un sous-programme qui contient le code qui demande à l’utilisateur des données sur la longueur et la largeur.

* Pour créer un sous-programme, sélectionnez Mes Blocs,   
  Créer un bloc.   
  Nommez le premier sous-programme obtenirDonnéesEntrée et sélectionnez OK.
* Séparez les blocs du programme initial.

Glisser les blocs demander et mettre appropriés de manière à ce qu’ils apparaissent sous le sous-programme obtenirDonnéesEntrée, comme illustré ici.



Utiliser des sous-programmes pour  
 organiser votre code (suite)

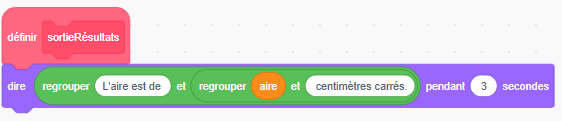
**L’algèbre**

**Unité 3, Fiche 8c**

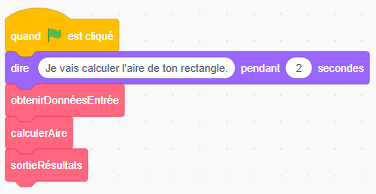
1. Créez un deuxième sous-programme nommé calculerAire.   
   Glisser le code de calcul mettre aire pour qu’il apparaisse en dessous.



3. Créez un troisième sous-programme nommé sortieRésultats.   
 Glissez le bloc dire qui indique l’aire calculée pour qu’il apparaisse en dessous.



4. Vous pouvez maintenant modifier votre programme principal pour « appeler » les sous-programmes à l’action, comme indiqué ici.



* Vous avez divisé le programme principal en étapes plus petites et plus faciles à gérer. Cela signifie qu’il y a moins de code dans le programme principal. Cela permet également de clarifier ce qui se passe dans chaque partie.
* Si vous devez déboguer ou corriger des erreurs dans l’application, vous pouvez vous concentrer sur l’un des sous-programmes plutôt que d’avoir à examiner tout le code en même temps.
* Essayez l’application avec les dimensions que vous avez utilisées dans la partie 1 de la Fiche 7 pour confirmer qu’elle affiche les aires correctes.

Utiliser des sous-programmes pour  
 organiser votre code (suite)

**L’algèbre**

**Unité 3, Fiche 8d**

Partie 2 : Une application qui calcule l’aire d’un cercle

Maintenant que vous savez comment créer des sous-programmes, vous pouvez les incorporer dans l’application que vous avez utilisée pour calculer l’aire d’un cercle de la Fiche 7. Utilisez ce lien pour accéder à l’application sans sous-programmes :

<https://scratch.mit.edu/projects/1194489665/editor/>

Créez des sous-programmes pour chaque partie de l’application.   
Suivez le pseudo-code ci-dessous. Nommez vos sous-programmes en vous basant sur les noms utilisés dans le pseudo-code. Vérifiez votre nouveau programme en utilisant les diamètres de la partie 2 de la Fiche 7.

*Sous-programme Obtenir les données d’entrée*

|  |
| --- |
| Sous-programme obtenirDonnéesEntrée  sortie « Entre le diamètre du cercle en centimètres : »  **diamètre** = entrée de l’utilisateur |

*Sous-programme Calculer le rayon*

|  |
| --- |
| Sous-programme calculerRayon  **rayon = diamètre/2** |

*Sous-programme Calculer l’aire*

|  |
| --- |
| Sous-programme calculerAire  **aire** = **pi** \* **rayon** \* **rayon** |

*Sous-programme Sortie de résultats*

|  |
| --- |
| Sous-programme sortieRésultats  sortie « L’aire est de “ **aire** “ centimètres carrés. » |

*Programme principal*

|  |
| --- |
| pi = 3.14  sortie « Je vais calculer l’aire de ton cercle. »  exécute le sous-programme obtenirDonnéesEntrée  exécute le sous-programme calculerRayon  exécute le sous-programme calculerAire  exécute le sous-programme sortieRésultats |