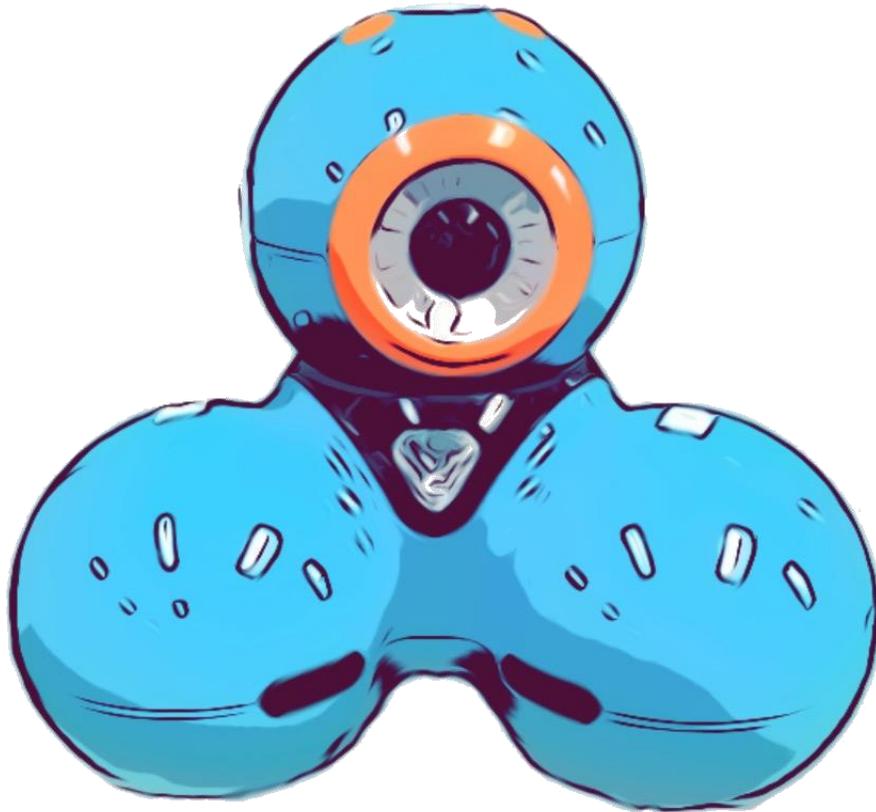


Programme d'apprentissage au codage avec Dash et Dot

Niveau E



NIVEAU E (4^e et 5^e année)

- E1 Boucles : répéter jusqu'à
- E2 Conditions : si... alors/sinon
- E3 Activité débranchée : les Fonctions
 - E4 Fonctions : partie 1
 - E5 Fonctions : partie 2



Subject: Re: Learn to code curriculum

Case Number: 77472

Wonder Workshop | Feb 28, 2019 08:47AM PST

Demers Marie-Hélène

Subject: Learn to code curriculum

FEB 28, 2019 | 08:48AM PST

Jeremy replied:

Hey Demers,

We really appreciate your translation efforts and you have our approval to provide the content to your teaching network.

We'd also like to get regular feedback from your community on the curriculum and we would ask that the teachers who are using the content register their products if they haven't done so already.

Please let us know if you need anything else.

Best regards,

Jeremy

How satisfied were you with the resolution we provided today?



FEB 17, 2019 | 12:10PM PST

Original message

Demers wrote:

Hello,

I am a francophone teacher in Canada, specifically in Lévis in the province of Quebec. In the last year, our government has promoted massive robot purchase for schools. Several have purchased several Dash robots and some Dot.

With other professionals from my school board, we are trying to train the teachers who will work with these wonderful robots. For my part, I discovered Dash and Dot this fall and I love them. My students and I have a lot of fun learning coding with them. I discover with enthusiasm all the resources that you offer on your site. On the teacher portal, I particularly appreciate your "Learn to Code Curriculum", offered online, free to teachers. I think for many teachers anxious about novelty or technology, this progression of activities directed with several tips for teachers can reassure and convince many of my colleagues to embark on the adventure.

Unfortunately, Quebec is a francophone province and most of my colleagues are not as comfortable with English as I am.

I have translated, for my own use and professional development, a part of "Learn to Code Curriculum" as well as some additional material in order to use it in my class. I would now like to know if Wonder Workshop would allow me to distribute these translations to my colleagues so that they can also benefit from your expertise. Of course, I always take care to quote you as creator of the product. All I want is to make it accessible to other teachers who are passionate about Dash but have little English. If you allow me, I can also undertake to send you the result of my work as you go so that you can judge the respect of your product, copyrights and the quality of my work. Currently, nothing is offered in French for Dash and Dot and I am struggling to provide my colleagues with resources that could help them get started in the adventure of Dash and Dot.

Thank you

This message was sent to demersm81@cnavigateurus.qc.ca in reference to Case #: 77472.

E1

Durée : 90 minutes

Boucles : répéter jusqu'à



by Wonder Workshop

Share

Tweet



La description

Niveau E - Leçon 1

Les élèves apprendront comment et pourquoi utiliser les blocs *Répéter jusqu'à*. Ils utiliseront les programmes préprogrammés Blockly et les cartes de défis pour mettre en pratique leurs nouvelles compétences.

Les objectifs

Les élèves vont :

- Identifiez les différences entre les blocs *Répéter*, *Répéter sans cesse* et *Répéter jusqu'à*.
- Différencier les situations dans lesquelles un bloc *Répéter*, *Répéter sans cesse* ou *Répéter jusqu'à* doit être utilisé.
- Utilisez les blocs *Répéter jusqu'à* pour réviser le code et résoudre les problèmes de codage dans un processus itératif.
- Construisez un programme original en utilisant un bloc *Répéter jusqu'à*.

Ce dont vous aurez besoin

- Dash et Dot
- Une tablette numérique par groupe
- Projecteur ou écran interactif avec possibilité de projection à l'écran du TNI
- Cartes de défis: E 1.1, E 1.2, E 1.3 (facultatif)
- Un tableau et un marqueur effaçable à sec par groupe (facultatif)
- Accès facultatif à Seesaw ou une autre plateforme pour diffuser le travail réalisé

Expérience préalable

Il est recommandé aux élèves d'avoir complété:

- Toutes les leçons des niveaux A, B, C et D
- Cartes de défis des niveaux A, B, C et D: toutes [facultatif]

Avant de débiter...

- Chargez complètement les tablettes et les robots.
- Installez l'application Wonder Workshop **Blockly** sur chaque tablette si ce n'est pas déjà fait.

Activité créée par Wonder Workshop et traduite par Marie-Hélène Demers, enseignante à la CSOU.

Traduction autorisée par Wonder Workshop

<https://education.makewonder.com/curriculum/learn-to-code>

Mise en situation

Espace Puzzles de Dash

1. Demandez aux élèves de compléter le défi de Dash **Patinage**. Vous pouvez trouver des indices, des solutions et des extensions pour le défi ici:
<https://drive.google.com/open?id=1-QdTMHbWcZ2D-kp1g8-i6mejV7LfIe3W>
2. Demandez aux élèves de compléter le défi de Dash **Concours de danse**. Vous pouvez trouver des indices, des solutions et des extensions pour le défi ici:
<https://drive.google.com/open?id=OBOWfNM4W+K2oWUphMGIES3FowFE>
3. Demandez aux élèves de mettre à jour leurs listes de contrôle Blockly Puzzle.
4. Remarque:
Si les précédents défis sont verrouillés, montrez aux élèves comment les déverrouiller rapidement à l'aide de cette stratégie: <https://help.makewonder.com/customer/en/portal/articles/2085752-how-do-i-skip-blockly-challenges->.

Activation des connaissances antérieures : les boucles

1. Passez brièvement en revue avec les élèves la définition et les utilisations des boucles en demandant:
« À quoi servent les boucles? »
Exemple de réponse: « L'action de faire quelque chose encore et encore. »
« Pourquoi utilisons-nous des boucles dans le codage? »
Exemple de réponse: « Nous utilisons des boucles pour rendre nos codes plus efficaces et plus faciles à écrire. »
2. Dites: « Vous avez appris différents types de boucles: les boucles **Répéter** et **Répéter sans cesse**. Qu'est-ce qui différencie les deux types de boucles? »
Exemple de réponse: « Les boucles de répétition répètent un nombre de fois spécifique, tandis que les boucles de répétition sans cesse se répètent jusqu'à ce que l'utilisateur leur dise de s'arrêter. »
3. Donnez un exemple d'utilisation de chaque type de boucle dans la vie réelle.
 - Une boucle de répétition: « Vous mettez une chaussette et répétez cette action une seconde fois. »
 - Une boucle **Répéter sans cesse**: « Vous inspirez et expirez. Vous répétez cette action encore et encore à moins que quelqu'un vous dise de retenir votre souffle. »

Connexions code.org [facultatif]

1. Passez en revue le cours D de Code.org, Leçon 8: <https://curriculum.code.org/csf/coursed/8/>
2. Dites: « Dans cette activité, vous avez pu aider le producteur à utiliser des blocs de répétition. Puis, demandez: « En quoi votre code aurait-il été différent si vous n'aviez pas été en mesure d'utiliser les blocs de répétition? »
Exemple de réponse: « Le code aurait été beaucoup plus long. »

Brève vérification

- « Quels sont les avantages à utiliser des boucles dans le code? »
Exemple de réponse: « L'utilisation de boucles rend notre code plus rapide et plus efficace en écriture. »

- « Dans quelle situation voudriez-vous utiliser une boucle de répétition par opposition à une boucle de répétition permanente? »

Exemple de réponse: « Lorsque vous écrivez votre code, vous utilisez une boucle de répétition si vous souhaitez que quelque chose se répète un certain nombre de fois. Vous utilisez une boucle de répétition permanente lorsque vous souhaitez que quelque chose se répète jusqu'à ce que le programme s'arrête. »

Réalisation

Introduction : Cache-cache

1. Projetez l'écran de votre tablette, ouvrez l'application **Blockly** et accédez au menu situé en haut à gauche de l'écran.
2. Accédez au menu **Créer nouveau** et sélectionnez le programme prédéfini **Cache-cache**. Puis, appuyez sur **Créer**.
3. Demandez aux élèves de regarder le programme et de prédire ce qui, selon eux, se passera quand il sera exécuté.

Exécutez le programme pour voir si les prédictions des élèves étaient correctes.

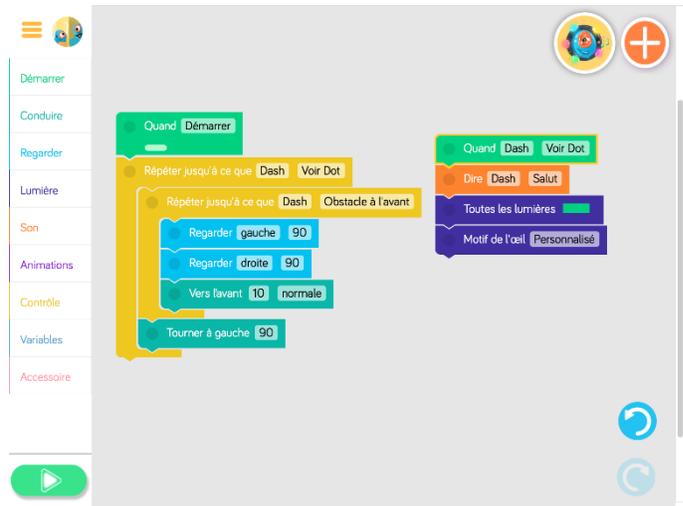
4. Pointez sur le bloc **Répéter jusqu'à** du programme et dites: « Il s'agit d'un nouveau type de bloc de répétition. Avec ce bloc, Dash ou Dot répète ce qu'il y a dans la boucle jusqu'à ce qu'un évènement spécifique se produise. »

Demandez à la classe: « Quelle action Dash va-t-il répéter jusqu'à ce qu'il y ait un obstacle devant? »

Exemple de réponse: « Dash cherchera Dot. Ensuite, Dash avancera. »

Demandez à la classe: « Que se passera-t-il quand Dash verra un obstacle devant lui? »

Exemple de réponse: « Dash regardera à gauche. »



Brève vérification

- « Quels évènements sont utilisés pour les blocs **Répéter jusqu'à** dans ce programme? Pourquoi pensez-vous que les programmeurs ont choisi ces évènements? »
- « Auriez-vous pu utiliser un bloc **Répéter sans cesse** pour lire le programme **Cache-cache**? Pourquoi? »

Activité collective

Cache-cache

1. Demandez à la classe: « Comment pouvons-nous changer le code pour que Dash cherche Dot jusqu'à ce que nous appuyions sur le bouton du haut? »

Sélectionnez un élève volontaire pour modifier le code en appuyant sur le bloc **Répéter jusqu'à** et en sélectionnant l'icône **Bouton du haut**.

Puis, lancez le programme pour que la classe puisse voir comment le programme a changé.

2. Continuez à sélectionner des élèves volontaires pour modifier le bloc *Répéter jusqu'à* de la même manière. Encouragez les élèves à:
 - Modifier l'évènement *Répéter jusqu'à entendre un clap* ou un *objet derrière*.
 - Changer les blocs à l'intérieur du bloc *Répéter jusqu'à*.

Bref retour

- « Quels sont les différents types d'évènements que vous pouvez utiliser pour le bloc *Répéter jusqu'à*? »

Expérimentation par les élèves

Répartissez les élèves en petits groupes (idéalement deux à trois élèves par robot). Encouragez-les à partager le temps passé sur leur tablette et leur robot. Demandez-leur d'établir et de faire la rotation entre des rôles tels que:

- Programmeur principal: tient la tablette et manipule les blocs de code.
- Technicien: récupère et replace le robot après chaque tentative de programme.
- Documentariste: enregistre les résultats, les réflexions et les progrès du groupe. Illustre les dessins et les idées du groupe.

Cache-cache

Demandez aux élèves d'utiliser la *Feuille de planification* pour concevoir des moyens d'ajouter ou de modifier le programme prédéfini de *Cache-cache*. Ils peuvent:

- Changer les paramètres pour Dash.
- Changer et/ou ajouter plus de lumières, de regards ou de sons.
- Ajouter plus de blocs de conduite.
- Changer le gestionnaire d'évènement.

Après avoir terminé la révision du programme, demandez aux élèves de:

- Remplir une feuille de travail de réflexion.
- Prendre une capture d'écran de leur code Blockly.
- Prendre une vidéo de Dash pendant que le code est en cours d'exécution.

Cartes de défis

Vous pouvez acheter nos ensembles de cartes de défis (Learn to Code Challenge Card) et notre guide pédagogique *Apprendre à coder* (Curriculum Guide) ici: <https://store.makewonder.com/#/education>

Demandez aux élèves de compléter les cartes de défis suivantes:

E1.1: Danse en ligne

E1.2: Danse avec Dot

E1.3: Piste de danse encombrée

Pour chaque défi, encouragez les élèves à:

- Utiliser la *Feuille de planification de Dot* pour expliquer comment ils peuvent relever le défi.
- Consulter la *Feuille de dépannage* et celle de *Résolution de problèmes et débogage* s'ils rencontrent des problèmes avec leur code (*Guide pédagogique papier payant*).

Après avoir terminé chaque carte de défi, demandez aux élèves de:

- Remplir une feuille de travail *Wonder Journal: Réflexion*.
- Prendre une capture d'écran de leur code Blockly.
- Prendre une vidéo de Dash pendant que le code est en cours d'exécution.
- Mettre à jour leurs listes de contrôle de carte de défis.

Retour collectif

Présentation des élèves

1. Demandez aux groupes d'élèves de partager à tour de rôle l'un de leurs programmes avec la classe. Encouragez-les à:
 - Expliquer leur conception. Par exemple, « Nous voulions ajouter des animations à notre programme parce que nous voulions faire danser Dash. » ou « Nous avons utilisé le ou les blocs *Répéter jusqu'à* pour contrôler la façon dont Dash répète chaque mouvement de danse. »
 - Partager tous les obstacles et les difficultés qu'ils ont surmontés pendant l'activité. Par exemple : « Au début, nous ne savions pas pourquoi nous devrions utiliser un bloc *Répéter jusqu'à*, mais nous nous sommes rendus compte que cela nous permettait de mieux contrôler notre programme. »
 - Se demander comment ils ont atteint leurs objectifs et à se donner des informations sur leurs programmes. Les questions/commentaires possibles incluent:
 - « Comment avez-vous utilisé des indices dans votre programme? »
 - « J'aime comment tu as changé les lumières et les sons dans ton programme. »
 - « Et si vous aviez ajouté un autre gestionnaire d'évènements pour Dot? »

Questions de suivi/discussion

- « Comment les blocs *Répéter jusqu'à* ont-ils été utilisés dans ce programme? »
Exemple de réponse: « Les blocs *Répéter jusqu'à* ont permis à Dash de répéter une série de comportements placés dans le bloc *Répéter jusqu'à*. Les comportements ont continué jusqu'à ce qu'un comportement attribué indique à Dash d'arrêter ou de faire quelque chose de différent. »
- À quoi ressemblerait la vie si vous n'utilisiez que des boucles *Répéter X fois* ou *Répéter sans cesse*?
Exemple de réponse: « Lorsque vous avez faim, vous mettez de la nourriture dans votre bouche et recommencez jusqu'à ce que vous soyez rassasié. Si vous utilisiez un bloc *Répéter sans cesse*, vous ne cesseriez jamais de manger. Si vous utilisez un bloc de répétition, vous risquez de ne pas avoir assez à manger ou de trop manger. »
- « Si vous viviez dans un monde où toute votre vie avait besoin de programmation, dans quelles situations devriez-vous utiliser un bloc *Répéter jusqu'à*? »
Exemple de réponse: « J'utiliserais un bloc *Répéter jusqu'à* pour mes devoirs. Je répéterais le travail jusqu'à ce qu'il soit terminé. »

Évaluation

- Utilisez notre rubrique d'évaluation pour examiner le travail et les présentations des élèves.
- [Facultatif] Partagez le résultat de leur travail avec le monde entier en utilisant @wonderworkshop et #dashanddot!

E2

Durée : 50 minutes

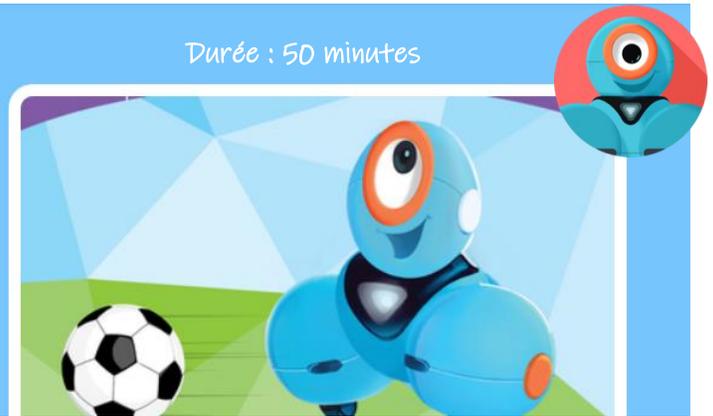
Conditions : si...alors/sinon



by Wonder Workshop

f Share

🐦 Tweet



La description

Niveau E - Leçon 2

Les élèves apprendront comment et quand utiliser les blocs *Si... alors/sinon*. Ils utiliseront les programmes préprogrammés Blockly et les cartes de défis pour mettre en pratique leurs nouvelles compétences.

Les objectifs

Les élèves vont :

- Définir les blocs *Si... alors/sinon*.
- Utiliser les blocs *Si... alors/sinon* pour résoudre un problème.
- Différencier les situations où les blocs *Si... alors/sinon* doivent être utilisés à la place des blocs *Si*.

Ce dont vous aurez besoin

- Dash
- Une tablette numérique par groupe
- Projecteur ou écran interactif avec possibilité de projection à l'écran du TNI
- Cartes de défis: E 2.1, E 2.2, E 2.3 (facultatif)
- Dot (facultatif)
- Un tableau et un marqueur effaçable à sec par groupe (facultatif)
- Accès facultatif à Seesaw ou une autre plateforme pour diffuser le travail réalisé

Expérience préalable

Il est recommandé aux élèves d'avoir complété:

- Toutes les leçons des niveaux A, B, C et D
- La leçon 1 du niveau E
- Cartes de défis des niveaux A, B, C et D: toutes [facultatif]
- Cartes de défis du niveau E : 1.1 à 1.3

Avant de débiter...

- Chargez complètement les tablettes et les robots.
- Installez l'application Wonder Workshop **Blockly** sur chaque tablette si ce n'est pas déjà fait.

Activité créée par Wonder Workshop et traduite par Marie-Hélène Demers, enseignante à la CSOU.

Traduction autorisée par Wonder Workshop

<https://education.makewonder.com/curriculum/learn-to-code>

Mise en situation

Activation des connaissances antérieures: les conditions

1. Examinez brièvement la manière dont les élèves ont utilisé les conditions dans les leçons précédentes en demandant:
 - « Quels blocs conditionnels avez-vous utilisés dans le passé? »
Exemple de réponse: « *Si*. »
 - « Quel est le but des blocs conditionnels? »
Exemple de réponse: « Pour permettre aux programmes informatiques de faire des choix. »

Connexions code.org [étape facultative]

1. Visionnez la vidéo de Code.org sur les blocs *Si*: <https://www.youtube.com/watch?v=fVUL-vzrTcM>.
2. Passez en revue le cours D, leçon 10 de Code.org sur les conditions: <https://studio.code.org/s/coursed-draft/stage/10/puzzle/1>. Puis, demandez: « Comment avez-vous utilisé les conditions dans l'activité de l'abeille? » Exemple de réponse: « Les conditions préalables étaient utilisées pour indiquer s'il y a une fleur sous le nuage, collectez le nectar. » Auriez-vous pu terminer ce défi SANS condition? Exemple de réponse: « Non, je n'aurais pas pu terminer l'activité sans utiliser un bloc de condition, car je ne savais pas ce qu'il y avait sous le nuage. L'utilisation d'un bloc de condition a permis au programme de prendre une décision. »

Réalisation

Introduction

1. Dites: « Nous avons appris des choses sur les blocs de condition au cours des leçons précédentes. Nous savons que les blocs conditionnels aident les programmes informatiques à prendre des décisions. Dans cette leçon, nous étudierons un nouveau type de blocs conditionnels: les blocs *Si/Autre* (le plus souvent nommés *Si... alors/sinon* en programmation) pour prendre des décisions d'une manière différente de celle des blocs *Si*. »
2. Dites: « Les blocs *Si* indiquent à l'ordinateur d'exécuter une action si une condition est remplie. Si cette condition n'est pas remplie, l'ordinateur n'effectue aucune action. Les blocs *Si/Autre* (*Si... alors/sinon*) indiquent à l'ordinateur de prendre une décision entre l'action A et l'action B. L'ordinateur vérifie la présence d'une condition. Si la condition est remplie, le programme exécute l'action A. Toutefois, si la condition n'est pas remplie, le programme exécute l'action B. Dans les deux cas, le bloc *Si/Autre* (*Si... alors/sinon*) entraîne l'exécution d'une action par l'ordinateur. »
3. Passez en revue une déclaration réelle *Si/Autre* (*Si... alors/sinon*) telle que:
Si le menu du dîner de la cafétéria indique qu'ils serviront une pizza alors, j'achèterai mon dîner à l'école. Sinon, j'apporterai mon dîner de la maison.
4. Demandez: « Pourriez-vous me donner des exemples de la façon dont on peut utiliser les déclarations *Si/Autre* (*Si... alors/sinon*) dans votre vie? »

Brève vérification

- « Quel est le rôle des conditions? »
- « Où trouve-t-on des conditions dans Blockly? »
- « En quoi les blocs *Si/Autre* (*Si... alors/sinon*) sont-ils différents des blocs *Si*? »

Activité collective

Fais ton choix

1. Projetez l'écran de votre tablette, ouvrez l'application **Blockly** et accédez au menu situé en haut à gauche de l'écran.
2. Allez dans le menu **Créer nouveau** et sélectionnez le programme prédéfini **Fais ton choix**. Puis, appuyez sur **Créer**.

3. Demandez aux élèves de regarder le programme et de prédire ce qui, selon eux, se passera quand il sera exécuté.

Exécutez le programme pour voir si les prédictions des élèves étaient correctes.

4. Demandez, « Qu'est-ce qui se passe quand Dash est soulevé? »

Exemple de réponse: « Dash dit: Oh oh! »

Demandez, « Qu'est-ce qui se passe quand Dash n'est pas soulevé? »

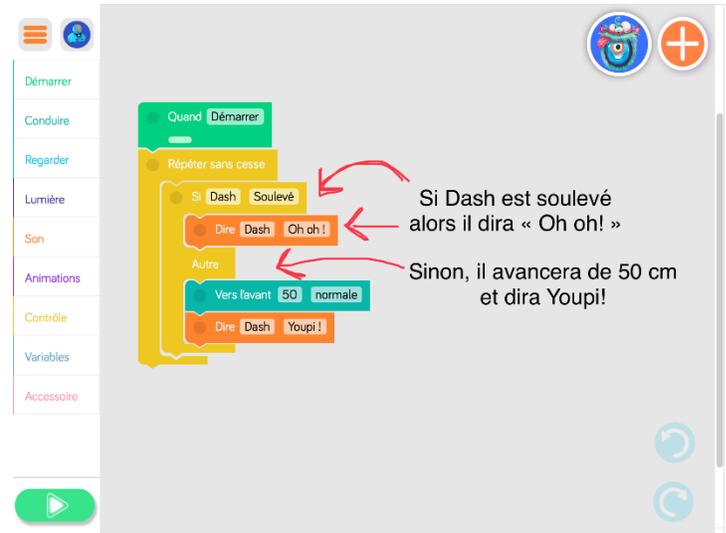
Exemple de réponse: « Le code progresse vers le code situé à l'intérieur de la partie **Autre** du bloc. »

Ensuite, Dash avance et dit *Youpi!* »

5. Demandez: « Comment changez-vous les blocs à l'intérieur du bloc **Si/Autre**? »

Exemple de réponse: « Si vous souhaitez modifier les blocs, faites glisser ceux que vous ne souhaitez pas utiliser vers la corbeille en bas à droite de l'écran. Ensuite, choisissez les blocs que vous souhaitez ajouter et placez-les à l'intérieur de la partie **Si** ou **Autre** du bloc. »

6. Demandez: « Pourquoi l'utilisation d'un bloc **Répéter sans cesse** est-elle utile dans ce programme? »
Exemple de réponse: « Un bloc **Répéter sans cesse** est utile car cela signifie que Dash recherchera en permanence un signal à capter. »



Si Dash est soulevé alors il dira « Oh oh! »

Sinon, il avancera de 50 cm et dira Youpi!

Bref retour

- « Comment le bloc **Si/Autre** (*Si... alors/sinon*) a-t-il été utilisé dans ce programme? »
- « Comment fonctionnent les blocs **Si/Autre** (*Si... alors/sinon*)? »
- « Pourquoi les blocs **Si/Autre** (*Si... alors/sinon*) sont-ils importants? »

Expérimentation par les élèves

Répartissez les élèves en petits groupes (idéalement deux à trois élèves par robot). Encouragez-les à partager le temps passé sur leur tablette et leur robot. Demandez-leur d'établir et de faire la rotation entre des rôles tels que:

- **Programmeur** principal: tient la tablette et manipule les blocs de code.
- **Technicien**: récupère et replace le robot après chaque tentative de programme.
- **Documentariste**: enregistre les résultats, les réflexions et les progrès du groupe. Illustre les dessins et les idées du groupe.

Fais ton choix

Demandez aux élèves d'utiliser la *Feuille de planification* pour concevoir des moyens d'ajouter ou de modifier le programme prédéfini *Fais ton choix*. Ils peuvent:

- Changer les paramètres.
- Ajoutez plus de sons, de lumières ou d'animations.
- Changer les mouvements.

Après avoir terminé la révision du programme, demandez aux élèves de:

- Remplir une feuille de travail de réflexion.
- Prendre une capture d'écran de leur code Blockly.
- Prendre une vidéo de Dash pendant que le code est en cours d'exécution.

Cartes de défis

Vous pouvez acheter nos ensembles de cartes de défis (Learn to Code Challenge Card) et notre guide pédagogique *Apprendre à coder* (Curriculum Guide) ici: <https://store.makewonder.com/#/education>

Demandez aux élèves de compléter les cartes de défis suivantes:

E 2.1: Suivez le leader

E 2.2: C'est le but!

E 2.3: Prêt, pas prêt, j'y vais!

Pour chaque défi, encouragez les élèves à:

- Utiliser la *Feuille de planification de Dash et/ou de Dot* pour expliquer comment ils peuvent relever le défi.
- Consulter la *Feuille de dépannage* et celle de *Résolution de problèmes et débogage* s'ils rencontrent des problèmes avec leur code (*Guide pédagogique papier payant*).

Après avoir terminé chaque carte de défi, demandez aux élèves de:

- Remplir une feuille de travail Wonder Journal: Réflexion.
- Prendre une capture d'écran de leur code Blockly.
- Prendre une vidéo de Dash pendant que le code est en cours d'exécution.
- Mettre à jour leurs listes de contrôle des cartes de défis.

Retour collectif

Présentation des élèves

1. Demandez aux groupes d'élèves de partager à tour de rôle l'un de leurs programmes avec la classe. Encouragez-les à:
 - Expliquer leur conception. Par exemple, « Nous avons ajouté une animation différente au programme parce que nous voulions faire dire à Dash: *Aidez-nous!* » « Nous avons utilisé le bloc *Si/Autre* ici pour permettre au programme de prendre plus d'une décision. »
 - Partager les obstacles et les difficultés qu'ils ont surmontés au cours de l'activité. Par exemple, « Au début, nous ne savions pas pourquoi il y avait deux blocs *Si/Autre*, nous en avons donc retiré un.

Activité créée par Wonder Workshop et traduite par Marie-Hélène Demers, enseignante à la CSOU.

Traduction autorisée par Wonder Workshop

<https://education.makewonder.com/curriculum/learn-to-code>

Cela nous a aidés à comprendre que les deux blocs étaient importants car les imbriquer a permis au programme de faire plus d'un choix. »

- Se demander comment ils ont atteint leurs objectifs et à se donner des informations sur leurs programmes. Les questions/commentaires possibles incluent:

- « Comment avez-vous programmé Dash pour réagir de cette façon? »
- « J'aime comment tu as changé les paramètres. »
- « Et si vous ajoutiez un bloc *Répéter* dans la partie *Autre* du bloc? »

Questions de suivi/discussion

- « Comment le bloc *Si/Autre* (*Si... alors/sinon*) vous a-t-il aidé dans votre programme? »
- « Auriez-vous obtenu le même résultat en utilisant simplement un bloc *Si*? »

Évaluation

- Utilisez notre rubrique d'évaluation pour examiner le travail et les présentations des élèves.
- [Facultatif] Partagez le résultat de leur travail avec le monde entier en utilisant [@wonderworkshop](#) et [#dashanddot!](#)

E3

Durée : 45 minutes

Activité débranchée : les fonctions



by Wonder Workshop

Share

Tweet



La description

Niveau E - Leçon 3

Les élèves apprendront comment et pourquoi utiliser des fonctions pour créer leur propre version d'un chant familier.

Les objectifs

Les élèves vont :

- Comprendre la définition et les utilisations d'une fonction.
- Comprendre la corrélation entre le refrain d'une chanson et une fonction.
- Écrire une chanson ou un chant original en utilisant une fonction.

Ce dont vous aurez besoin

- *Bear Hunt Reference Sheet.pdf*
- *Sample full song.jpg*
- *Bear Hunt Student Worksheet.pdf*
- Projecteur ou écran interactif avec possibilité de projection à l'écran du TNI
- Crayons et papiers
- Ciseaux
- Fiches
- Ruban autocollant
- Robot Dash (facultatif)
- Accès facultatif à Seesaw ou une autre plateforme pour diffuser le travail effectué

Expérience préalable

Il est recommandé aux élèves d'avoir complété:

- Toutes les leçons des niveaux A, B, C et D
- Les leçons du niveau E, leçons 1 et 2
- Cartes de défis des niveaux A, B, C et D: toutes [facultatif]
- Cartes de défis du niveau E: 1.1 à 2.3 [facultatif]

Activité créée par Wonder Workshop et traduite par Marie-Hélène Demers, enseignante à la CSOU.

Traduction autorisée par Wonder Workshop

<https://education.makewonder.com/curriculum/learn-to-code>

Avant de débiter...

- Découpez les quatre images de Dot à partir d'une impression du plateau de jeu de Dot. Celles-ci seront utilisées comme jetons pour le jeu.

Mise en situation

Activation des connaissances antérieures

1. Dites: « Nous avons beaucoup appris sur la programmation. Tournez-vous vers votre partenaire et dites-lui deux choses que vous avez apprises sur la programmation. »
Exemple de réponse: « J'ai appris que lorsque j'utilise une boucle, tout ce qui s'y trouve est répété. J'ai aussi appris qu'un autre mot pour une déclaration avec un si est condition. »
2. Dites: « Comment les boucles et les conditionnels peuvent-ils nous aider à rendre notre code plus efficace? »
Exemple de réponse: « Les boucles nous aident à répéter des parties du code, évitant ainsi de répertorier ces étapes encore et encore. »

Réalisation

Introduction

1. Dites: « Aujourd'hui, nous allons en apprendre davantage sur un nouveau concept de codage: les fonctions. »
2. « Vous utilisez des fonctions tous les jours dans les routines de votre vie. Souvent, lorsque vous faites des choses encore et encore, vous n'avez pas à penser à chaque partie de la tâche; votre corps sait quoi faire automatiquement. » Discutez d'exemples concrets de fonctions:
« Lorsque vous vous brossez les dents, vous ne pensez pas: Je vais mettre de l'eau sur ma brosse à dents. Je vais ouvrir le dentifrice. Je vais mettre du dentifrice sur ma brosse à dents. Je vais me brosser les dents de devant. Je vais me brosser les dents d'en arrière. Je vais mettre de l'eau dans ma bouche et recracher l'eau et le dentifrice dans l'évier. Vous pensez plutôt: Je vais me brosser les dents. C'est un exemple de fonction. »
3. Regardez la vidéo de Code.org sur les fonctions: <https://studio.code.org/s/courseee-draft/stage/6/puzzle/1>
4. Dites: « Une fonction est quand une partie d'un code est répétée et peut être appelée tout au long du programme. Les fonctions sont utilisées lorsque vous devez répéter un morceau de code plusieurs fois. »
Discutez des exemples de situations dans lesquelles vous souhaitez utiliser une fonction dans un programme. Par exemple, lorsque vous souhaitez que Dash exécute la même danse à différents moments du programme. »

Brève vérification

- « Qu'est-ce qu'une fonction et quand l'utilisez-vous? »
Exemple de réponse: « Les fonctions sont utilisées lorsque vous devez répéter un morceau de code plusieurs fois. »
- « Quand pourriez-vous utiliser une fonction dans la vie réelle? »

Exemple de réponse: « Vous pouvez utiliser une fonction lorsque vous écrivez ou chantez une chanson car les chœurs des chansons sont similaires à des fonctions. »

Activité collective

Fonctions et refrain

1. Demandez: « À quoi servent les fonctions? »

Exemple de réponse: « Les fonctions sont des morceaux de code qui peuvent être répétés tout comme le refrain d'une chanson. »

2. Dites: « Nous allons utiliser la chanson *Partir à la chasse à l'ours* pour s'exercer à utiliser des fonctions. »

Regardez la version de Michael Rosen et Helen Oxenbury de *Chasse à l'ours* sur TeacherTube:

<http://www.teachertube.com/video/we8217re-going-on-a-bear-hunt-274958>.

Demandez aux élèves de suivre la chanson plusieurs fois.

3. Répartissez les élèves en groupes de deux ou trois et distribuez la feuille de travail de l'élève à la chasse à l'ours.

Demandez aux groupes de discuter des choses qu'ils remarquent à propos de la chanson.

Exemples de réponses: « Les paroles de la chanson ne sont pas dans le bon ordre. » ou « Certaines parties de la chanson se répètent. »

Demandez aux élèves de découper les cases dans la feuille de travail de l'élève de Hunt Bear et de placer les morceaux de la chanson dans l'ordre.

4. Demandez: « Si je vous avais demandé d'écrire à la main toute cette chanson, cela prendrait beaucoup de temps. Que pourrions-nous faire pour rendre la chanson plus courte et plus rapide à écrire? »

Exemple de réponse: « Identifiez les parties de la chanson qui se répètent sous le nom de Chorus (refrain). Par conséquent, après la première utilisation, vous pouvez simplement écrire refrain à la place de ces paroles. »

5. Dites: « Un refrain de chanson et une fonction de codage sont similaires. Une fonction est un morceau de code qui se répète, et un refrain est une partie d'une chanson qui se répète. On peut faire appel à eux deux sans avoir à les réécrire à chaque fois. »

Bref retour

- Demandez: « Pourquoi est-il important de savoir utiliser des fonctions? »

Expérimentation par les élèves

Demandez aux élèves de travailler sur les activités suivantes en petits groupes. Nous recommandons deux à trois élèves par groupe.

Lorsque les élèves travaillent ensemble pendant le codage, ils peuvent s'aider mutuellement à identifier les erreurs et à développer des solutions créatives!

1. Dites: « Nous allons écrire notre propre version de *Aller à la chasse à l'ours*, mais cette fois-ci, nous utiliserons des fonctions pour faciliter la tâche. Votre chanson s'appellera *Partir à la chasse au Dash*. »

2. Au tableau, écrivez *Refrain* et écrivez le refrain de la chanson ci-dessous:

We're goin' on a Dash hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

3. Dites: « Le *refrain* de cette chanson agira comme notre fonction. Chaque fois que nous voudrions répéter le refrain, nous écrirons *Refrain*. Ainsi, nous pourrions appeler notre fonction *Refain* au lieu d'écrire tous les mots à chaque fois. »
4. Sur une fiche, demandez à chaque groupe d'élèves d'écrire leur propre vers de la chanson sur les obstacles à surmonter lors de la recherche de Dash. Écrivez les exemples de paroles suivantes au tableau pour que les élèves se réfèrent:

Oh look! It's some silly Dots!
Can't go over them,
Can't go under them,
Can't go through them,
Got to go around them!

5. Rassemblez toutes les fiches contenant les couplets et collez-les au tableau dans l'ordre. Travaillez avec la classe pour réécrire les paroles restantes de la chanson en vous basant sur un endroit effrayant où ils pourraient trouver Dash et les obstacles que les élèves ont fournis dans leurs couplets. Les paroles devraient ressembler à ceci:
6. Dites: « Nous avons déjà écrit notre fonction *Refrain*. Chaque fois que le refrain se répète, nous allons simplement écrire *Refrain* au tableau.

Uh, oh! It's dark in here.
I feel something,
It makes lots of sounds!
It lights up!
It's DASH!

Hurry back around the stack of Dots,
Back through the noisy class,
Back through the jumping monkeys,
Run in the house and lock the door.
Phew! That was close!

I'm not afraid!

Chorus

Oh look! It's a deep, dark cave!
Can't go over it,
Can't go under it,
Can't go through it,
Got to go in it.

Uh, oh! It's dark in here.
I feel something,
It makes lots of sounds!
It lights up!
It's DASH!

Hurry back around the stack of Dots,
Back through the noisy class,
Back through the jumping monkeys,
Run in the house and lock the door.
Phew! That was close!

I'm not afraid!



7. Demandez: « Après avoir écrit notre propre version de cette chanson, êtes-vous contents d'avoir utilisé des fonctions? Pourquoi? »
Exemple de réponse: « Je suis heureux d'avoir utilisé des fonctions, car elles ont rendu l'écriture de cette chanson beaucoup plus facile et plus rapide. »
8. Facultatif: Demandez à la classe d'interpréter la chanson, chaque groupe jouant son couplet et toute la classe jouant le refrain et les paroles restantes.
Invitez un élève à jouer le rôle de Dash.

Retour collectif

Présentation des élèves

Demandez aux groupes d'élèves de partager à tour de rôle leurs versets avec la classe.

Encouragez-les à:

- Expliquer leur pensée. Par exemple: « Nous avons choisi d'ajouter cet obstacle à notre couplet parce que nous l'avions trouvé amusant. »
- Se faire des commentaires sur leurs couplets.

Les questions/commentaires possibles incluent:

- « Comment avez-vous choisi ces obstacles? »
- « Et si tu ajoutais un autre couplet, que se passerait-il? »

Questions de suivi/discussion

- « Comment utilisez-vous les fonctions dans votre vie réelle? »
Exemple de réponse: « J'utilise les fonctions lorsque je prends un bain ou que je m'habille. »
- « Pourquoi les fonctions sont-elles importantes? Que se passerait-il dans nos vies si nous n'utilisions pas de fonctions? »
Exemple de réponse: « Les fonctions nous aident à effectuer des tâches de base dans notre vie sans trop y penser. Si nous n'utilisions pas les fonctions, il nous faudrait beaucoup de temps pour réaliser les activités les plus élémentaires. »

Évaluation

- Utilisez notre rubrique d'évaluation pour examiner le travail et les présentations des élèves.
- [Facultatif] Partagez le résultat de leur travail avec le monde entier en utilisant @wonderworkshop et #dashanddot!

E4

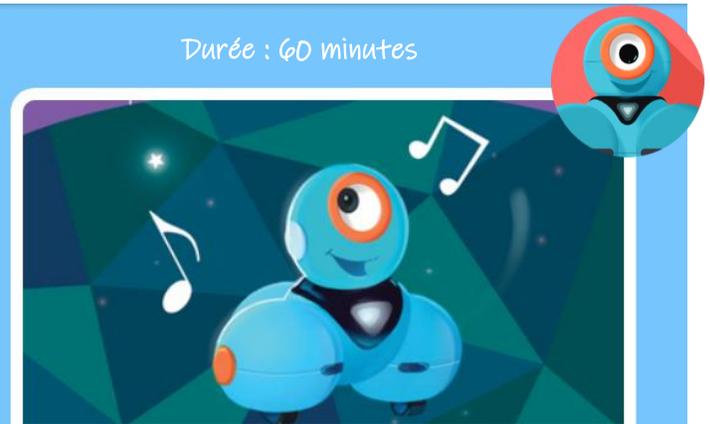
Durée : 60 minutes

Fonctions : partie 1

 by Wonder Workshop

f Share

🐦 Tweet



La description

Niveau E - Leçon 4

Les élèves examineront ce qu'ils ont appris dans la leçon précédente sur les blocs de fonctions et exploreront davantage l'utilisation de fonctions dans la programmation informatique. Ils utiliseront les programmes préprogrammés Blockly et les cartes de défis pour mettre en pratique leurs nouvelles compétences.

Les objectifs

Les élèves vont :

- Définir une fonction.
- Identifier les différences entre une fonction et une boucle et identifier les moments où une fonction doit être utilisée à la place d'une boucle.
- Utiliser les fonctions pour réviser le code et résoudre les problèmes de codage dans un processus itératif.

Ce dont vous aurez besoin

- Dash
- Une tablette numérique par groupe
- Projecteur ou écran interactif avec possibilité de projection à l'écran du TNI
- Cartes de défis: E 3.1, E 3.2, E 3.3 (facultatif)
- Un tableau et un marqueur effaçable à sec par groupe (facultatif)
- Accès facultatif à Seesaw ou une autre plateforme pour diffuser le travail réalisé

Expérience préalable

Il est recommandé aux élèves d'avoir complété:

- Toutes les leçons des niveaux A, B, C et D
- Les leçons 1,2 et 3 du niveau E
- Cartes de défis des niveaux A, B, C et D: toutes [facultatif]
- Cartes de défis du niveau E : 1.1 à 2.3 [facultatif]

Avant de débiter...

- Chargez complètement les tablettes et les robots.

Activité créée par Wonder Workshop et traduite par Marie-Hélène Demers, enseignante à la CSOU.

Traduction autorisée par Wonder Workshop

<https://education.makewonder.com/curriculum/learn-to-code>

- Installez l'application Wonder Workshop **Blockly** sur chaque tablette si ce n'est pas déjà fait.

Mise en situation

Activation des connaissances antérieures : les fonctions

1. Demander: « Qu'est-ce qu'une fonction? »
Exemple de réponse: « Une fonction est un morceau de code qui peut être appelé à plusieurs reprises. »
2. Demandez: « Comment avez-vous utilisé les fonctions dans la dernière leçon? »
Exemple de réponse: « Nous avons utilisé des fonctions pour créer une chanson/un chant. »
3. Demandez: « Pourquoi l'utilisation de fonctions était-elle importante dans cette activité? »
Exemple de réponse: « Les fonctions nous ont permis d'écrire un chant sans avoir à réécrire le refrain chaque fois qu'il était utilisé. »

Connexions code.org [étape facultative]

1. Visionnez la vidéo de Code.org sur les fonctions: <https://studio.code.org/s/course3/stage/9/puzzle/1>
2. Après avoir regardé la vidéo, demandez: « Nommez-moi des chansons que vous connaissez et qui ont un refrain. »
Exemple de réponse: « Jingle Bells »

Réalisation

Introduction

1. Dites: « Dans la dernière leçon, nous avons appris à utiliser les fonctions d'écriture de chansons. Aujourd'hui, nous allons apprendre à utiliser les fonctions d'un programme informatique en élève un programme prédéfini de Blockly. »
2. Examinez des exemples concrets dans lesquels vous utiliseriez des fonctions telles que:
 - a. Faire un événement de routine, comme se brosser les dents.
 - b. Apprendre une danse qui a des pas répétés.
3. Discutez de la manière dont les élèves pourraient utiliser des fonctions dans leur code: « Nous pouvons utiliser des fonctions pour répéter des parties de notre code encore et encore. Contrairement aux boucles, nous pouvons répéter le code de la fonction à plusieurs endroits de notre programme. »

Brève vérification

- « Comment les fonctions d'un programme sont-elles similaires au refrain d'une chanson? »
Exemple de réponse: « Le refrain d'une chanson se répète plusieurs fois au cours de la chanson. La fonction d'un code peut se répéter plusieurs fois au cours du programme. Il existe des espaces de code entre chaque instance d'une fonction dans un programme, comme il en existe en musique entre chaque instance d'un refrain dans une chanson. »
- « Comment utilisez-vous les fonctions dans votre vie? »
Exemple de réponse: « Prendre une douche, c'est comme utiliser des fonctions. Il y a plusieurs étapes différentes, mais je n'ai pas à penser à toutes les étapes. Je peux facilement effectuer ces étapes à différentes heures, comme le matin ou après l'exercice. »

Activité collective

Oui non oui

1. Projetez l'écran de votre tablette, ouvrez l'application **Blockly** et accédez au menu situé en haut à gauche de l'écran.
2. Allez dans le menu **Créer nouveau** et sélectionnez le programme prédéfini **Oui non oui**. Puis, appuyez sur **Créer**.
3. Demandez aux élèves de regarder le programme et de prédire ce qui, selon eux, se passera quand il sera exécuté.

Exécutez le programme pour voir si les prédictions des élèves étaient correctes.

4. Demandez: « Comment pourriez-vous changer les mouvements de Dash dans les fonctions? »

Exemple de réponse: « Vous pouvez ajouter plus de blocs ou modifier les angles des blocs de regard. »

5. Demandez aux élèves volontaires d'identifier les blocs d'appels dans le programme. Puis montrez-leur où trouver le bloc d'appel dans le menu Démarrer.

6. Dites: « Pourriez-vous exécuter ce programme en utilisant un bloc de répétition au lieu d'une fonction? Pourquoi? »

Exemple de réponse: « Vous ne pouvez pas utiliser un bloc de répétition car les fonctions ne se répètent pas consécutivement. Les blocs de répétition ne peuvent être utilisés que dans des espaces de code qui se répètent à la même place dans le programme. »

7. Dites: « Dans **Oui Non Oui**, nous avons utilisé des fonctions pour que Dash effectue différentes actions. Comment pouvez-vous modifier les fonctions du programme afin qu'elles fassent en sorte que Dash démontre deux comportements opposés? »



Bref retour

- « Comment les fonctions ont-elles été utilisées dans ce programme? »
- « Auriez-vous pu écrire ce programme sans fonctions? »

Exemple de réponse: « Oui, vous auriez pu écrire ce programme sans fonctions, mais le programme aurait été beaucoup plus long. »

Expérimentation par les élèves

Répartissez les élèves en petits groupes (idéalement deux à trois élèves par robot). Encouragez les élèves à partager le temps passé sur leur tablette et leur robot. Demandez-leur d'établir et de faire la rotation entre des rôles tels que:

- Programmeur principal: tient la tablette et manipule les blocs de code.
- Technicien: récupère et replace le robot après chaque tentative de programme.
- Documentariste: enregistre les résultats, les réflexions et les progrès du groupe. Illustre les dessins et les idées du groupe.

Oui non oui

Demandez aux élèves d'utiliser la *Feuille de planification* pour concevoir des moyens d'ajouter ou de modifier le programme prédéfini *Oui non oui*. Ils peuvent:

- Changer les noms des fonctions.
- Modifiez/ajoutez des lumières, des mouvements et des sons à chaque fonction.
- Ajouter une boucle.
- Ajoutez des conditions en utilisant les blocs *Si* ou *Si/Autre*.
- Changer l'ordre dans lequel les fonctions sont appelées.

Après avoir terminé la révision du programme, demandez aux élèves de:

- Remplir une feuille de travail de réflexion.
- Prendre une capture d'écran de leur code Blockly.
- Prendre une vidéo de Dash pendant que le code est en cours d'exécution.

Cartes de défis

Vous pouvez acheter nos ensembles de cartes de défis (Learn to Code Challenge Card) et notre guide pédagogique *Apprendre à coder* (Curriculum Guide) ici: <https://store.makewonder.com/#/education>

Demandez aux élèves de compléter les cartes de défis suivantes:

E 3.1: Répétition de danse

E 3.2: Danse avec fantaisie

E 3.3: Dash danse en finale

Pour chaque défi, encouragez les élèves à:

- Utiliser la feuille de planification de Dash et/ou de Dot pour expliquer comment ils peuvent relever le défi.
- Consulter la *Feuille de dépannage* et celle de *Résolution de problèmes et débogage* s'ils rencontrent des problèmes avec leur code (Guide pédagogique papier payant).

Après avoir terminé chaque carte de défi, demandez aux élèves de:

- Remplir une feuille de travail Wonder Journal: Réflexion.
- Prendre une capture d'écran de leur code Blockly.
- Prendre une vidéo de Dash pendant que le code est en cours d'exécution.
- Mettre à jour leurs listes de contrôle de carte de défis.

Espace Puzzles de Dash

1. Demandez aux élèves de résoudre le défi de Dash *Robot curieux*. Vous pouvez trouver des solutions et des extensions pour le défi ici:

<https://drive.google.com/open?id=1sqhBJIRPdpiBAEQ5UXDadSIKUFRMxPDU>.

2. Demandez aux élèves de mettre à jour leurs listes de contrôle Blockly Puzzle.

3. Remarque: si les précédents défis sont verrouillés, montrez aux élèves comment les déverrouiller rapidement à l'aide de cette stratégie:

<https://help.makewonder.com/customer/en/portal/articles/2085752-how-do-i-skip-blockly-challenges->.

Retour collectif

Présentation des élèves

1. Demandez aux groupes d'élèves de partager à tour de rôle l'un de leurs programmes avec la classe. Encouragez-les à:
 - Expliquer leur conception. Par exemple, « Nous voulions ajouter une autre fonction au programme parce que nous voulions ajouter des animations. » ou « Nous avons modifié les actions dans les blocs de fonctions ici parce que nous voulions que Dash continue à avancer. »
 - Partager tous les obstacles et les difficultés qu'ils ont surmontés pendant l'activité. Par exemple, « Au début, nous ne savions pas comment appeler une fonction, mais nous avons trouvé le bloc Appel dans le menu Démarrer. »
 - Se demander comment ils ont atteint leurs objectifs et à se donner des informations sur leurs programmes. Les questions/commentaires possibles incluent:
 - « Comment as-tu programmé Dash pour bouger comme ça? »
 - « J'aime comment tu as ajouté une autre fonction. »
 - « Et si vous changiez l'ordre des blocs d'appel, que se passerait-il? »

Questions de suivi/discussion

- « Comment le bloc fonction vous a-t-il aidé dans votre programme? »
Exemple de réponse: « Le bloc fonction m'a aidé à répéter des séquences de code non consécutives. »
- « Auriez-vous pu écrire le même programme en utilisant des boucles? »
Exemple de réponse: « Non, car les boucles répètent des lignes de code consécutives. »

Évaluation

- Utilisez notre rubrique d'évaluation pour examiner le travail et les présentations des élèves.
- [Facultatif] Partagez le résultat de leur travail avec le monde entier en utilisant @wonderworkshop et #dashanddot!

E5

Durée : 90 minutes

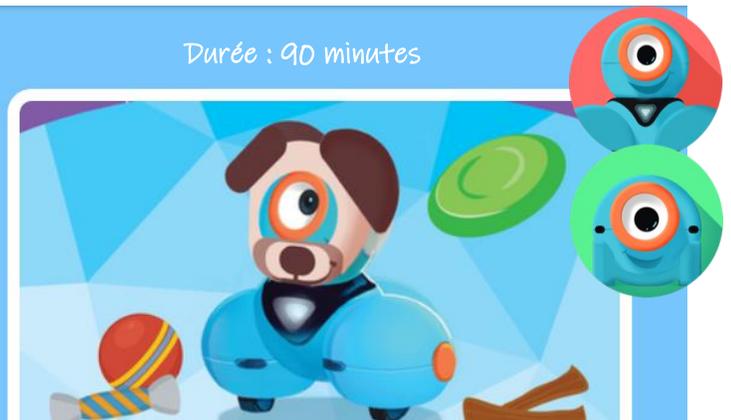
Fonctions : partie 2



by Wonder Workshop

f Share

🐦 Tweet



La description

Niveau E - Leçon 5

Les élèves utiliseront des fonctions et les combineront avec des gestionnaires d'évènements et des boucles. Ils utiliseront les programmes préprogrammés Blockly et les cartes de défis pour mettre en pratique leurs nouvelles compétences.

Les objectifs

Les élèves vont :

- Définir une fonction.
- Concevoir un programme qui combine des gestionnaires d'évènements, des boucles et des fonctions.
- Utiliser les fonctions pour réviser le code et résoudre les problèmes de codage dans un processus itératif.

Ce dont vous aurez besoin

- Dash et Dot
- Une tablette numérique par groupe
- Projecteur ou écran interactif avec possibilité de projection à l'écran du TNI
- Cartes de défis: E 3.4, E 3.5, E 3.6 (facultatif)
- Un tableau et un marqueur effaçable à sec par groupe (facultatif)
- Accès facultatif à Seesaw ou une autre plateforme pour diffuser le travail réalisé

Expérience préalable

Il est recommandé aux élèves d'avoir complété:

- Toutes les leçons des niveaux A, B, C et D
- Les leçons 1, 2, 3 et 4 du niveau E
- Cartes de défis des niveaux A, B, C et D: toutes [facultatif]
- Cartes de défis du niveau E : 1.1 à 3.3 [facultatif]

Avant de débiter...

- Chargez complètement les tablettes et les robots.
- Installez l'application Wonder Workshop **Blockly** sur chaque tablette si ce n'est pas déjà fait.

Activité créée par Wonder Workshop et traduite par Marie-Hélène Demers, enseignante à la CSOU.

Traduction autorisée par Wonder Workshop

<https://education.makewonder.com/curriculum/learn-to-code>

Mise en situation

Activation des connaissances antérieures : les fonctions

1. Examinez brièvement comment les élèves ont utilisé les fonctions lors de la leçon précédente en demandant:
 - « Quel est le rôle des fonctions? »
Exemple de réponse: « Les fonctions aident à rendre les programmes plus efficaces en nous permettant de répéter des morceaux de code tout au long du programme. »
 - « En quoi les fonctions sont différentes des boucles? »
Exemple de réponse: « Les boucles répètent des parties de code consécutivement, alors que les fonctions répètent des sections de code à différents endroits. »

Connexions code.org [facultatif]

1. Passez en revue la leçon de Code.org sur les fonctions et les boucles:
<https://studio.code.org/s/coursee-draft/stage/7/puzzle/1>
2. Indiquez aux élèves comment utiliser les fonctions et les boucles pour compléter la leçon.
Demandez: « Qu'est-ce qui était difficile dans cette tâche? »
Exemple de réponse: « Il était difficile de suivre les différentes parties du code. Je n'étais pas sûr de ce qui se passerait ensuite. »

Réalisation

Introduction :

1. Dites: « Nous avons effectué quelques activités à l'aide de fonctions. Nous allons maintenant combiner des fonctions avec d'autres concepts de codage connus tels que les boucles et les gestionnaires d'événements. »
2. Examinez un exemple concret dans lequel vous pouvez utiliser tous ces concepts ensemble:
« Lorsque vous allez au magasin, vous utilisez une boucle pour parcourir le magasin. Vous montez dans une allée, tournez à droite/à gauche puis répétez. Une fois dans l'allée, vous utilisez un gestionnaire d'événements en continuant à marcher jusqu'à ce qu'un événement se produise, tel que voir quelque chose qui se trouve sur votre liste d'achats. Lorsque l'événement se produit, vous appelez une fonction de **récupération** c'est-à-dire que vous tendez la main, ramassez l'objet, retirez votre main et mettez l'objet dans le panier. »

Brève vérification

- Demandez: « Qu'est-ce qu'une boucle? »
Exemple de réponse: « Une boucle est un morceau de code qui se répète. »
- Demandez: « Comment fonctionne un gestionnaire d'événements? »
Exemple de réponse: « Un gestionnaire d'événements attend qu'un événement se produise avant d'exécuter le programme sous celui-ci. »

Activité collective

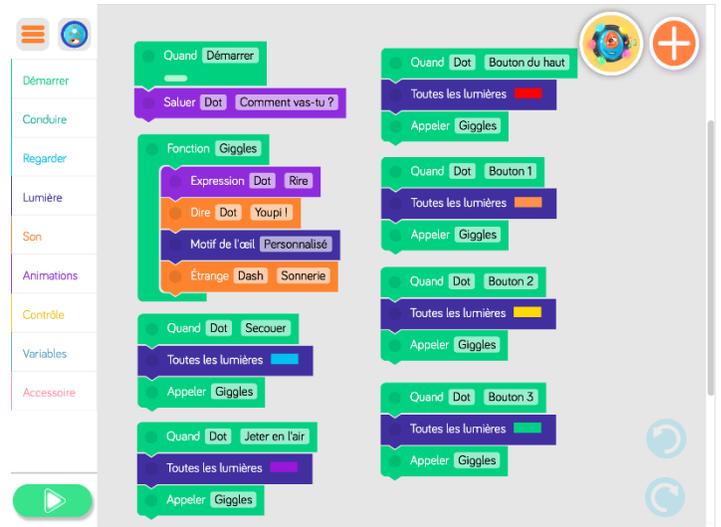
Guili-guili

1. Projetez l'écran de votre tablette, ouvrez l'application **Blockly** et accédez au menu situé en haut à gauche de l'écran.
2. Allez dans le menu **Créer nouveau** et sélectionnez le programme prédéfini **Guili-guili**. Puis appuyez sur **Créer**.
3. Dites: « Voyons comment Blockly utilise à la fois les fonctions et les gestionnaires d'évènements dans le programme prédéfini **Guili-guili**. »

Demandez aux élèves de regarder le programme et de prédire ce qui, selon eux, se passera quand il sera exécuté.

Exécutez le programme pour voir si les prédictions des élèves étaient correctes.

4. Demandez: « Combien de gestionnaires d'évènements pouvez-vous trouver dans ce programme? »
5. Demandez aux élèves volontaires de décrire ce qui se passe au début du programme.
Exemple de réponse: « Dot dit: Comment allez-vous? »
6. Demandez: « Comment voulez-vous que ce programme fonctionne pour Dash au lieu de Dot? »
Exemple de réponse: « Modifiez chaque gestionnaire d'évènements en **évènements Dash**. Modifiez les blocs de son et d'animation de Dot en sons et animations pour Dash. »
7. Dites: « Et si nous voulions que Dot rigole deux fois lorsque le bouton supérieur de Dot est enfoncé? Que devrions-nous changer? »
Exemple de réponse: « Ajouter un autre bloc d'appel. »
Demandez à un élève volontaire d'ajouter un autre bloc d'appel sous le gestionnaire d'évènements sélectionné, puis appuyez sur **Jouer** pour voir comment le programme a changé.



Bref retour

- « Combien de fonctions pouvez-vous utiliser dans un programme? »
Exemple de réponse: « Vous pouvez utiliser un nombre illimité de fonctions. »
- « Si vous utilisez plusieurs fonctions, comment votre programme saura-t-il laquelle faire jouer? »
Exemple de réponse: « Vous attribuez à chaque fonction un nom différent de sorte que, lorsque vous appelez cette fonction, le programme sache celui que vous souhaitez. »

Expérimentation par les élèves

Répartissez les élèves en petits groupes (idéalement deux à trois élèves par robot). Encouragez les élèves à partager le temps passé sur leur tablette et leur robot. Demandez-leur d'établir et de faire la rotation entre des rôles tels que:

- Programmeur principal: tient la tablette et manipule les blocs de code.
- Technicien: récupère et replace le robot après chaque tentative de programme.
- Documentariste: enregistre les résultats, les réflexions et les progrès du groupe. Illustre les dessins et les idées du groupe.

Guili-guili

Demandez aux élèves d'utiliser la feuille de planification pour concevoir des moyens d'ajouter ou de modifier le programme prédéfini de *Guili-guili*. Ils peuvent:

- Changer la fonction ou ajouter d'autres fonctions.
- Modifier ou ajoutez des lumières, des mouvements et des sons dans chaque fonction.
- Ajouter une boucle.
- Changer les repères

Après avoir terminé la révision du programme, demandez aux élèves de:

- Remplir une feuille de travail de réflexion.
- Prendre une capture d'écran de leur code Blockly.
- Prendre une vidéo de Dash pendant que le code est en cours d'exécution.

Cartes de défis

Vous pouvez acheter nos ensembles de cartes de défis (Learn to Code Challenge Card) et notre guide pédagogique *Apprendre à coder* (Curriculum Guide) ici: <https://store.makewonder.com/#/education>

Demandez aux élèves de compléter les cartes de défis suivantes:

E 3.4: Dresseur de chiens

E 3.5: Encore plus de tours!

E 3.6: Course à obstacles

Pour chaque défi, encouragez les élèves à:

- Utiliser la feuille de planification de Dot ou de Dash pour expliquer comment ils peuvent relever le défi.
- Consulter la *Feuille de dépannage* et celle de *Résolution de problèmes et débogage* s'ils rencontrent des problèmes avec leur code (Guide pédagogique papier payant).

Après avoir terminé chaque carte de défi, demandez aux élèves de:

- Remplir une feuille de travail Wonder Journal: Réflexion.
- Prendre une capture d'écran de leur code Blockly.
- Prendre une vidéo de Dash pendant que le code est en cours d'exécution.
- Mettre à jour leurs listes de contrôle de carte de défis.

Espace Puzzles de Dot

1. Demandez aux élèves de résoudre le défi de Dot *Un Robot adorable*. Vous pouvez trouver des solutions et des extensions pour le défi ici: <https://drive.google.com/open?id=10iGwYdhT+T+VlPukOGQVbptQY0ZC7EukQ>.
2. Demandez aux élèves de résoudre le défi de Dot *Boule disco*. Vous pouvez trouver des solutions et des extensions pour le défi ici: <https://drive.google.com/open?id=1H1DhoBXkr6dJQrDcMwDhsOTgYZTqWDF5k>
3. Demandez aux élèves de mettre à jour leurs listes de contrôle Blockly Puzzle.

4. Remarque: si les précédents défis sont verrouillés, montrez aux élèves comment les déverrouiller rapidement à l'aide de cette stratégie:

<https://help.makewonder.com/customer/en/portal/articles/2085752-how-do-i-skip-blockly-challenges->

Retour collectif

Présentation des élèves

1. Demandez aux groupes d'élèves de partager à tour de rôle l'un de leurs programmes avec la classe. Encouragez-les à:
 - Expliquer leur conception. Par exemple, « Nous voulions ajouter une autre fonction au programme parce que nous voulions inclure nos propres sons. » ou « Nous avons modifié les actions dans les blocs de fonctions pour que Dot ait la tête qui tourne. »
 - Partager tous les obstacles et les difficultés qu'ils ont surmontés pendant l'activité. Par exemple, « Au début, nous n'étions pas sûrs du meilleur endroit pour ajouter une boucle, mais nous avons ensuite exécuté le programme à plusieurs reprises avec la boucle dans divers endroits pour trouver le meilleur endroit pour le mettre. »Encouragez les élèves à se demander comment ils ont atteint leurs objectifs et à se donner des informations sur leurs programmes. Les questions/commentaires possibles incluent:
 - « Comment avez-vous ajouté une autre fonction? »
 - « J'aime comment tu as ajouté la boucle. »
 - « Et si tu changeais de repères, que se passerait-il? »

Questions de suivi/discussion

- « Pourquoi les fonctions sont-elles importantes? »
Exemple de réponse: « Les fonctions aident à répéter des séquences de code non consécutives. »
- « Comment votre programme a-t-il été affecté par les boucles? »
Exemple de réponse: « Les boucles m'ont permis de répéter des bits de code à plusieurs reprises. »

Évaluation

- Utilisez notre rubrique d'évaluation pour examiner le travail et les présentations des élèves.
- [Facultatif] Partagez le résultat de leur travail avec le monde entier en utilisant @wonderworkshop et #dashanddot!

Going on a Bear Hunt

reference sheet

Chorus

We're goin' on a bear hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

Oh look! It's some long, wavy grass!
Can't go over it,
Can't go under it,
Can't go around it,
Got to go through it!

We're goin' on a bear hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

Oh look! It's a mushroom patch.
Can't go over it,
Can't go under it,
Can't go around it,
Got to go through it!

We're goin' on a bear hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

Oh look! It's a wide river.
Can't go over it,
Can't go under it,
Can't go through it,
Got to swim across it.

We're goin' on a bear hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

Oh look! A deep, dark cave.
Can't go over it,
Can't go under it,
Can't go through it,
Got to go in it.

Uh, oh! It's dark in here.
I feel something,
It has lots of hair!
It has sharp teeth!
It's a bear!

Hurry back through the river,
Back through the mushroom
patch,
Back through the long grass,
Run in the house and lock the
door.
Phew! That was close!
I'm not afraid!

Going on a Bear Hunt

Student Worksheet

Hurry back through the river,
Back through the mushroom patch,
Back through the long grass,
Run in the house and lock the door.
Phew! That was close!
I'm not afraid!

We're goin' on a bear hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

We're goin' on a bear hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

Oh look! It's a mushroom patch.
Can't go over it,
Can't go under it,
Can't go around it,
Got to go through it!

We're goin' on a bear hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

Oh look! A deep, dark cave.
Can't go over it,
Can't go under it,
Can't go through it,
Got to go in it.

Oh look! It's a wide river.
Can't go over it,
Can't go under it,
Can't go through it,
Got to swim across it.

Oh look! It's some long, wavy grass!
Can't go over it,
Can't go under it,
Can't go around it,
Got to go through it!

Uh, oh! It's dark in here.
I feel something,
It has lots of hair!
It has sharp teeth!
It's a bear!

We're goin' on a bear hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

We're goin' on a Dash hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

Chorus

Oh look! It's some silly Dots!
Can't go over them,
Can't go under them,
Can't go through them,
Got to go around them!

Chorus

Oh look! It's a class of noisy children!
Can't go over it,
Can't go under it,
Can't go around it,
Got to go through it!

Chorus

Oh look! It's a bunch of jumping monkeys!
Can't go over them,
Can't go under them,
Can't go around them,
Got to go through them!

Chorus

Oh look! A deep, dark cave.
Can't go over it,
Can't go under it,
Can't go through it,
Got to go in it.

Uh, oh! It's dark in here.
I feel something,
It has makes lots of sounds!
It lights up!
It's DASH!

Hurry back around the stack of Dots,
Back through the noisy class,
Back through the jumping monkeys,
Run in the house and lock the door.
Phew! That was close!

I'm not afraid!

We're goin' on a Dash hunt,
We're going to catch a big one,
I'm not scared,
What a beautiful day!

Chorus

Oh look! It's some silly Dots!
Can't go over them,
Can't go under them,
Can't go through them,
Got to go around them!

Chorus

Oh look! It's a class of noisy children!
Can't go over it,
Can't go under it,
Can't go around it,
Got to go through it!

Chorus

Oh look! It's a bunch of jumping monkeys!
Can't go over them,
Can't go under them,
Can't go around them,
Got to go through them!

Chorus

Oh look! A deep, dark cave.
Can't go over it,
Can't go under it,
Can't go through it,
Got to go in it.

Uh, oh! It's dark in here.
I feel something,
It has makes lots of sounds!
It lights up!
It's DASH!

Hurry back around the stack of Dots,
Back through the noisy class,
Back through the jumping monkeys,
Run in the house and lock the door.
Phew! That was close!

I'm not afraid!