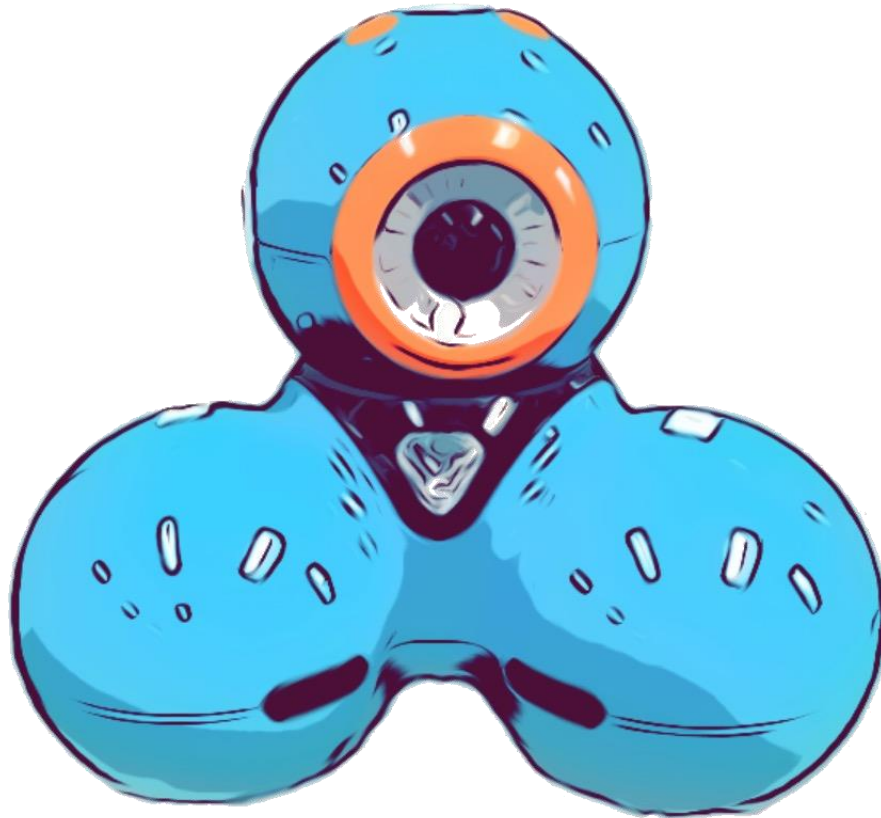


Programme d'apprentissage au codage avec Dash et Dot

Niveau C



NIVEAU C (2^e et 3^e année)

- C1 Séquences : paramètres complexes
- C2 Activité débranchée : gestionnaire d'évènements
- C3 Évènements : gestionnaire d'évènements partie 1
- C4 Évènements : gestionnaire d'évènements partie 2
- C5 Boucles : les boucles multiples



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.

C1

Séquences : paramètres complexes

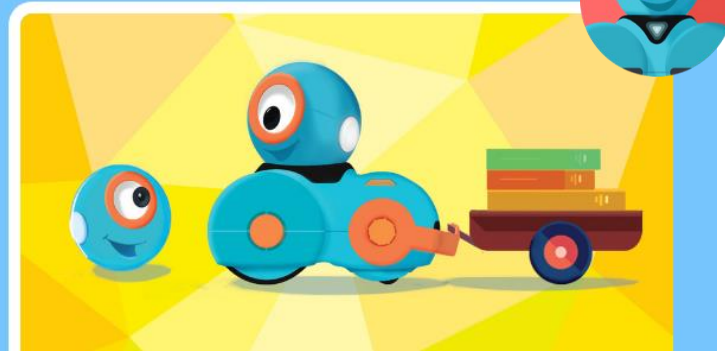
Durée : 60 minutes



by Wonder Workshop

f Share

🐦 Tweet



La description

Niveau C - Leçon 1

Les élèves exploreront la possibilité d'ajouter plus de complexité à leurs codes en programmant Dash pour qu'il regarde dans une variété de directions et se tourne. Ils utiliseront les programmes préprogrammés **Blockly** et les cartes de défis pour mettre en pratique leurs nouvelles compétences.

Les objectifs

Les élèves vont :

- Expérimenter une programmation plus complexe en changeant la direction dans laquelle Dash regarde et se tourne.
- Utiliser des paramètres complexes 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: C 1.1, C 1.2, C 1.3 (facultatif)
- Un robot Dot (facultatif)
- Un tableau et 1 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 et B
- Cartes de défis des niveaux A et B: 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 CSDU.

Traduction autorisée par Wonder Workshop

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

Mise en situation

Activation des connaissances antérieures : les événements

1. Examinez brièvement comment les élèves ont utilisé le bloc *Attendre* lors de la leçon précédente en demandant:
« Que fait le bloc *Attendre*? »
Exemple de réponse: « Le bloc *Attendre* fait que Dot ou Dash attendent un événement spécifique afin de continuer la séquence de code. »
2. Dites: « Dans le programme *Le piège de la boîte à gâteaux*, Dot a agi comme une alarme. Comment le bloc *Attendre* a-t-il aidé Dot à agir comme une alarme? »
Exemple de réponse: « Si vous cachez Dot dans un pot à biscuits, si quelqu'un cherchait un biscuit, il le prendrait à la place. Lorsque Dot a été déplacé, le bloc *Attendre* a été déclenché et la séquence d'alarme a commencé. »

Réalisation

Introduction

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 choisissez *Nouveau projet*.
3. Dites: « Nous pouvons programmer Dash pour adopter de nombreux comportements complexes. Dans cette leçon, nous allons apprendre à ajouter plus de complexité à nos séquences de codage. »
4. Tapez sur le menu *Regarder*.
Demandez aux élèves de vous dire ce qu'ils voient.
Exemple de réponse: « Je vois les blocs *Regarder en haut*, *Regarder gauche* et *Regarder vers la voix*. »
Expliquez que ces blocs font tourner la tête de Dash dans la direction indiquée.
5. Tapez sur le menu *Conduire* et choisissez *Tourner à gauche*.
Demandez aux élèves de vous dire ce qu'ils voient.
Exemple de réponse: « Je vois un cercle avec un quart de couleur jaune et le chiffre 90 en haut. »
Expliquez que vous pouvez contrôler le degré de rotation de Dash en ajustant le pointeur sur le cercle.
6. Dites: « Dans *Blockly*, nous appelons ces types de paramètres des paramètres complexes. Complexe signifie compliqué et paramètre est un autre mot pour mesure. Nous allons utiliser des paramètres complexes pour réviser le code aujourd'hui. »

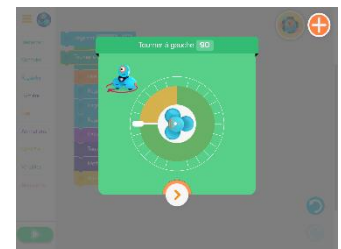
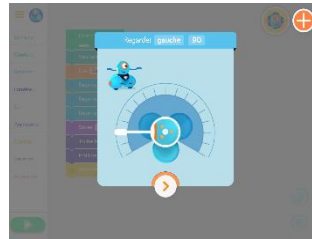
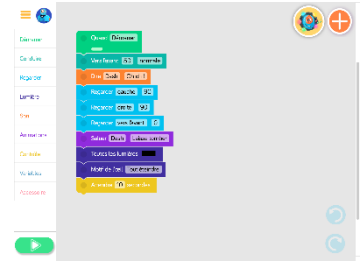
Brève vérification

- « Que sont les paramètres? »
Exemple de réponse: « Les paramètres sont des mesures. »
- « Que pouvons-nous faire en définissant des paramètres dans notre code? »
Exemple de réponse: « Nous pouvons dire à Dash de regarder dans différentes directions et de se tourner sous différents angles. »
- « Pourquoi les paramètres complexes sont-ils importants? »
Exemple de réponse: « L'utilisation de paramètres complexes nous donne plus d'options pour écrire du code et raconter des histoires avec du code. Ils nous permettent d'être plus précis. »

Activité collective

Oh non

1. Dans l'application **Blockly**, sélectionnez le programme prédéfini **Oh non**. Puis appuyez sur **Créer**.
2. Jouez le programme et demandez aux élèves de partager ce qu'ils remarquent. Ensuite, dites: « Nous allons ajouter des paramètres complexes à ce programme. »
3. Tapez sur le premier bloc **Regarder gauche**.
Montrez aux élèves comment l'image de la tête de Dash à l'écran indique la direction dans laquelle la tête de Dash va tourner.
4. Ajoutez le bloc **Regarder en haut** du menu **Regarder** et tapez dessus.
Montrez aux élèves comment l'image de la tête de Dash à l'écran indique la hauteur à laquelle sa tête regarde.
5. Dites: « Maintenant que vous connaissez ces deux paramètres complexes, nous allons apprendre à faire pivoter Dash dans la direction de votre choix. »
Allez au menu **Conduire** et appuyez sur **Tourner gauche**.
Dites: « Remarquez qu'un cercle entoure l'image de Dash avec un pointeur. Nous pouvons déplacer ce pointeur dans la direction spécifique à laquelle Dash doit se tourner. »
6. Dites: « Le programme **Oh non** n'utilise pas de bloc tournant. Où pensez-vous que nous devrions ajouter un bloc tournant? Dans quelle direction et dans quelle mesure devrions-nous faire tourner Dash? »
Exécutez le programme avec un bloc tournant à un ou deux endroits différents pour que les élèves puissent voir ce qu'il fait.



Bref retour

- « Donnez-moi un exemple de paramètre complexe dans l'application **Blockly**. »
Exemple de réponse: « Tourner la tête de Dash. »
- « Pourquoi les paramètres complexes sont-ils importants? »
Exemple de réponse: « Les paramètres complexes gardent notre code intéressant et nous donnent plus d'options de codage. »

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.

Oh non

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

- Modifier la séquence en ajoutant des blocs de lumière et/ou de son.
- Ajouter/modifier des paramètres complexes des blocs *Regarder* et/ou de *Conduire*

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:

C 1.1: Pas de devoirs!

C 1.2: Revenez!

C 1.3: Attends-moi Dot!

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.

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 que Dash tourne à 165 degrés, nous avons donc ajouté un virage à gauche et réglé le pointeur sur 165.
 - Partager les obstacles et les difficultés qu'ils ont surmontés pendant l'activité. Par exemple, « Au début, trouver comment changer la direction où Dash regarde était un défi, mais nous nous sommes ensuite souvenus qu'il fallait taper sur le bloc *Regarder*. »

- 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 changé le programme? »
 - « J'aime comment tu as ajouté un bloc *Tourner à droite*. »

Questions de suivi/discussion

- « Comment l'utilisation de paramètres complexes a-t-elle affecté votre programme? »
Exemple de réponse: « L'utilisation de paramètres complexes a rendu mon programme plus intéressant, diversifié. »
- « Donnez un exemple de la façon dont les paramètres complexes sont utilisés dans Blockly. »
Exemple de réponse: « Des paramètres complexes nous aident à faire tourner ou incliner la tête de Dash. Ils aident également Dash à se retourner dans une direction spécifique. »

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!

C2

Durée : 45 minutes

Activité débranchée : gestionnaire d'évènements



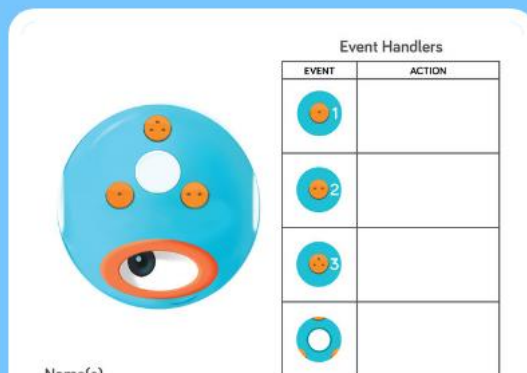
by Wonder Workshop



Share



Tweet



La description

Niveau C - Leçon 2

Les élèves apprendront la définition et la fonction des gestionnaires d'évènements. Les élèves utiliseront des gestionnaires d'évènements pour « programmer » leurs camarades de classe à l'aide des boutons de Dot.

Les objectifs

Les élèves vont :

- Définir un gestionnaire d'évènement.
- Trouver des exemples d'utilisation des gestionnaires d'évènements dans leur vie.
- Utiliser les gestionnaires d'évènements pour « programmer » leurs camarades de classe à l'aide des boutons de Dot.

Ce dont vous aurez besoin

- Annexe C2, Gestionnaires d'évènements (1 par groupe d'élèves et 1 pour l'enseignant)
- Projecteur ou écran interactif avec possibilité de projection à l'écran d'un TNI
- Crayon
- 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 et B
- Niveau C, leçon 1
- Cartes de défis des niveaux A et B: toutes [facultatif]
- Cartes de défis du niveau C: B 1.1 à 1.3 [facultatif]

Avant de débiter...

- Récupérez les images de l'annexe A1 Les flèches de Dash et Dot utilisées lors de l'activité A1.
- Imprimez également des exemplaires de l'annexe B2 Mouvements pour la danse.
- Ensuite, découpez ces nouvelles cartes.

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 algorithmes

1. Demandez: « Qu'est-ce qu'un algorithme? »
Exemple de réponse: « Un algorithme est une série d'instructions à suivre pour un ordinateur ou un robot. »
2. Demandez: « Pourquoi est-il important que les séquences du code que vous écrivez soient correctes? »
Exemple de réponse: « Les robots ne suivent que les instructions que vous avez données. Par conséquent, si vos séquences ne sont pas dans le bon ordre, le robot ne fera pas ce que vous souhaitez. »
3. Demandez: « Pourquoi utilisons-nous des boucles dans nos algorithmes? »
Exemple de réponse: « L'utilisation de boucles dans nos algorithmes nous permet d'écrire du code plus rapidement et plus facilement. »

Activation des connaissances antérieures : les événements

1. Demandez: « Qu'est-ce qu'un événement? »
Exemple de réponse: « Un événement est une action qui provoque quelque chose. »
2. Demandez: « Quel genre d'événements peuvent faire réagir Dash et Dot? »
Exemple de réponse: « Dash et Dot peuvent détecter une pression sur un bouton, des sons et des mouvements. »

Réalisation

Introduction

1. Regardez la vidéo Code.org sur les gestionnaires d'événements: <https://studio.code.org/s/coursec-draft/stage/10/puzzle/1>
Puis, demandez: « Que sont les gestionnaires d'événements? »
Exemple de réponse: « Les gestionnaires d'événements sont des éléments que l'ordinateur recherche constamment. »
« Pourquoi les gestionnaires d'événements sont-ils importants? »
Exemple de réponse: « Les gestionnaires d'événements nous permettent de contrôler notre programme pendant son exécution. »
2. Dites: « Nous venons de regarder une vidéo sur Nat, qui a aidé à programmer la première Xbox. »
Demandez: « Pourquoi a-t-il utilisé des gestionnaires d'événements? »
Exemple de réponse: « Nat a utilisé des gestionnaires d'événements pour rendre les jeux vidéo qu'il a conçus plus intéressants. »
Demandez: « Comment Nat a-t-il utilisé des gestionnaires d'événements dans ses jeux vidéo? »
Exemple de réponse: « Nat a utilisé des gestionnaires d'événements pour faire en sorte que les joueurs à l'écran sautent et avancent. »
3. Demandez: « Donnez un exemple de gestionnaire d'événements dans votre vie? »
Exemple de réponse: « Quand j'entends mon estomac grogner, je mange une collation. »

Brève vérification

- « Pourquoi les gestionnaires d'évènements sont-ils importants? »
- « Quand utilise-t-on les gestionnaires d'évènements? »
Exemple de réponse: « Les gestionnaires d'évènements sont utilisés dans les programmes chaque fois que nous souhaitons que quelque chose se produise par simple pression d'un bouton ou d'un levier. »
- « Un jeu vidéo serait-il intéressant à jouer sans gestionnaires d'évènements? Pourquoi ou pourquoi? »
Exemple de réponse: « Non, un jeu vidéo ne serait pas intéressant à jouer sans gestionnaires d'évènements, car nous n'aurions aucun moyen d'interagir avec le jeu. Sans gestionnaires d'évènements, le jeu serait simplement une séquence d'évènements sur laquelle nous n'aurions aucun contrôle. »

Activité collective

Le robot humain de Dot

1. Dites: « Aujourd'hui, nous allons utiliser les boutons de Dot pour contrôler un robot humain. »
Sélectionnez un élève volontaire pour être le "robot humain".
2. Montrez ou projetez la feuille de travail sur les gestionnaires d'évènements et dites: « Nous allons programmer le robot humain avec des gestionnaires d'évènements. Les gestionnaires d'évènements demanderont au robot humain d'exécuter une action chaque fois que l'on appuiera sur l'un des boutons de Dot. L'appui sur le bouton est un évènement et ce que le robot humain fait est une action. »
3. Demandez aux élèves de décider quelle action le robot humain devrait effectuer pour chaque pression sur un bouton. Ensuite, modélisez la feuille de travail des gestionnaires d'évènements avec les décisions de la classe. Par exemple: si le bouton 1 est enfoncé, le robot humain agitera ses mains. Si le bouton 2 est enfoncé, le robot humain émettra un son d'animal. Si le bouton 3 est enfoncé, le robot humain se retournera. Si le bouton du haut est enfoncé, le robot humain chantera une chanson.
4. Demandez aux élèves d'appuyer à tour de rôle sur les boutons de la feuille des gestionnaires d'évènements.
Après chaque appui sur une touche, demandez au robot humain d'exécuter l'action correspondante. (Facultatif) Si vous avez un robot Dot, demandez aux élèves d'appuyer sur les boutons du robot réel.
5. Sélectionnez différents élèves volontaires pour être le « robot humain » et répétez l'activité de la même manière.
6. Dites: « Utiliser les boutons de Dot comme gestionnaires d'évènements fonctionne de la même manière que d'utiliser la manette de jeu vidéo. Dot est la manette de jeu et le robot humain est le personnage du jeu. Lorsque vous appuyez sur l'un des boutons de Dot, vous créez un évènement qui amène le robot humain à exécuter une action. »

Bref retour

- « Dans cette activité, quels sont les évènements qui se sont produits? »
Exemple de réponse: « Appuyer sur les boutons de la tête de Dash. »
- « Que s'est-il passé après chaque évènement? »
Exemple de réponse: « Le robot humain a effectué une action différente selon le bouton. »
- « Quels types de gestionnaires d'évènements avons-nous créés dans l'activité? »
Exemple de réponse: « Nous avons fait sauter le robot humain lorsque nous avons appuyé sur le bouton 1. »

Expérimentation par les élèves

1. Organisez les élèves en petits groupes et distribuez une feuille de travail sur les gestionnaires d'évènements à chaque groupe.
(Facultatif) Si disponible, distribuez un robot Dot à chaque groupe.
2. Demandez aux élèves de concevoir leurs propres gestionnaires d'évènements. Ensuite, demandez aux élèves de jouer à tour de rôle le « robot humain » et le presseur de boutons.
(Facultatif) Si des robots Dot sont disponibles, demandez aux groupes d'appuyer sur les boutons réels de Dot lorsqu'ils testent leurs gestionnaires d'évènements.
3. Si le temps le permet, demandez aux groupes d'élèves d'échanger les feuilles de travail et d'essayer les gestionnaires d'évènements des autres.
4. Demandez aux élèves d'utiliser la feuille de réflexion pour décrire ce qui s'est passé pendant l'activité.
5. (Facultatif) Demandez aux élèves d'utiliser les blocs *Quand* et *Mes sons* de Blockly pour programmer Dot afin qu'il nomme l'action assignée lors de l'appui de chaque bouton. Par exemple, « Faire pivoter! »

Retour collectif

Présentation des élèves

1. Demandez aux groupes d'élèves de partager à tour de rôle leurs gestionnaires d'évènements avec la classe. Encouragez-les à:
 - Expliquer leur conception. Par exemple, « Nous voulions que le robot humain bouge beaucoup, alors nos gestionnaires d'évènements ont fait avancer le robot, le faire tourner à droite et tourner sur lui-même. »
 - Partager les obstacles et les difficultés qu'ils ont surmontés au cours de l'activité. Par exemple, « Notre robot humain avait des problèmes pour voir les boutons sur lesquels nous appuyions, nous avons donc nommé le nom du bouton avant d'appuyer sur ce dernier. »
 - Se demander comment ils ont atteint leurs objectifs et à se donner mutuellement des informations sur leurs organisateurs. Les questions/commentaires possibles incluent:
 - « J'aime la façon dont vous avez fait chanter le robot humain lorsque le bouton du haut a été enfoncé. »
 - « Et si vous faisiez chanter et danser le robot quand le bouton du haut est enfoncé? »

Questions de suivi/discussion

- « Comment avez-vous pu changer les gestionnaires d'évènements pour que le robot humain exécute une danse? »
Exemple de réponse: « Nous pourrions faire bouger le robot humain lorsque le bouton 1 est enfoncé et le faire tourner lorsque le bouton 2 est enfoncé. »
- « Quand avez-vous utilisé un gestionnaire d'évènements dans votre vie réelle aujourd'hui? »
Exemple de réponse: « Lorsque la cloche a sonné, j'ai rangé mon sac à dos dans mon casier. »

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!](#)

C3

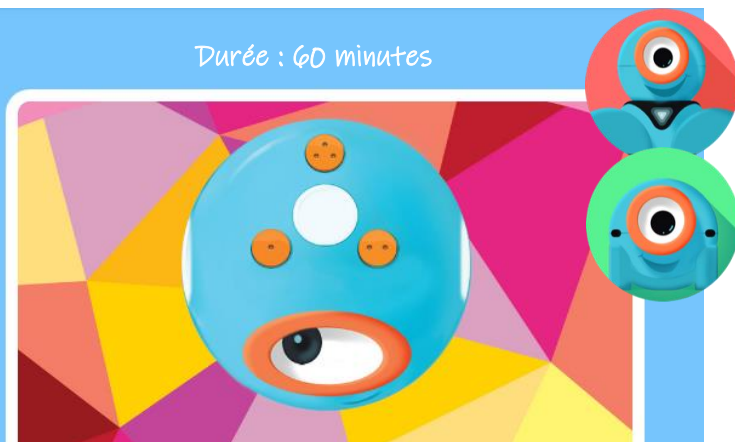
Durée : 60 minutes

Évènements : gestionnaire d'évènements partie 1



Share

Tweet



La description

Niveau C - Leçon 3

Les élèves passeront en revue leur programmation précédente en utilisant des évènements. Ils apprendront comment utiliser le bloc *Quand* dans Blockly pour créer des gestionnaires d'évènements pour Dash et Dot. Ils utiliseront les programmes préprogrammés Blockly et les cartes de défis pour mettre en pratique leurs nouvelles compétences.

Les objectifs

Les élèves vont :

- Comprendre quels problèmes sont mieux résolus avec les gestionnaires d'évènements via le bloc *Quand* de Blockly.
- Utilisez des gestionnaires d'évènements 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: C 2.1, C 2.2, C 2.3 (facultatif)
- Un tableau et 1 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 et B
- Les leçons 1 et 2 du niveau C
- Cartes de défis des niveaux A et B: toutes [facultatif]
- Cartes de défis du niveau C: C 1.1 à 1.3 [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.

Mise en situation

Activation des connaissances antérieures : les événements

1. « Quels sont les gestionnaires d'événements? »
Exemple de réponse: « Les gestionnaires d'événements permettent au code d'écouter ou d'attendre plusieurs événements pendant l'exécution du programme. Le code sous chaque gestionnaire d'événements ne s'exécute que lorsque cet événement se produit. »
2. « Pourquoi avons-nous besoin de gestionnaires d'événements? »
Exemple de réponse: « Les gestionnaires d'événements nous permettent de changer ce qui se passe dans notre programme alors qu'il est toujours en cours d'exécution. »
3. « Pourquoi les gestionnaires d'événements sont-ils importants dans les jeux vidéo? »
Exemple de réponse: « Les gestionnaires d'événements sont importants pour les jeux vidéo, car ils indiquent aux personnages à l'écran de sauter ou de courir lorsque nous appuyons sur un bouton spécifique. »

Connexions code.org [facultatif]

1. Voir la vidéo de Code.org sur les événements:
<https://www.youtube.com/watch?v=uDWV9shCoak&feature=youtu.be>
2. Après avoir regardé la vidéo, demandez:
 - « Qu'est-ce qu'un événement? »
Exemple de réponse: « Un événement est une action qui déclenche quelque chose. »
 - « Pourquoi utiliseriez-vous un gestionnaire d'événements? »
Exemple de réponse: « Lorsque nous souhaitons modifier un programme en cours d'exécution. »

Réalisation

Introduction

1. Ouvrez l'application **Blockly** et accédez au menu **Contrôle**. Sélectionnez ensuite un bloc **Attendre** et dites: « Un événement est une action qui provoque quelque chose. Nous avons utilisé des blocs **Attendre** avec Dash et Dot. Quels types d'événements Dash et Dot peuvent-ils détecter? »
Exemple de réponse: « Dash et Dot peuvent détecter les sons et les mouvements. »
2. Allez au menu **Démarrer** et appuyez sur le bloc **Quand**. Dites: « Nous allons maintenant créer des gestionnaires d'événements avec le bloc **Quand**. Que pensez-vous qu'il va se passer lorsque nous utilisons **Quand** dans notre code? »
Exemple de réponse: « Le code situé sous le bloc **Quand** sera joué lorsque vous appuierez sur le bouton du haut de Dash. »

Brève vérification

- « Qu'est-ce qu'un événement? »

Exemple de réponse: « Un évènement est une action qui provoque quelque chose. »

- « Quel bloc allons-nous utiliser pour créer des gestionnaires d'évènements avec Dash et Dot? »

Exemple de réponse: « Nous allons utiliser le bloc *Quand*. »

Activité collective

Formes en boucles

1. Dans l'application *Blockly*, allez au menu en haut à gauche de l'écran.
2. Dans *Blockly*, allez au menu *Créer nouveau* et sélectionnez le programme prédéfini *Formes en boucles*. Puis appuyez sur *Créer*.
3. Jouez le programme et appuyez sur le bouton 1 de la tête de Dash (c'est-à-dire le bouton orange avec 1 point). Demandez aux élèves: « Qu'avez-vous remarqué? »

Exemple de réponse: « Dash a fait quelque chose de différent lorsque vous avez appuyé sur le bouton 1. »

4. Demandez aux élèves: « Où se trouve le bloc *Quand* qui aide Dash à attendre un évènement? »

Sélectionnez un élève pour pointer le bloc *Quand*. Ensuite, dites: « Ceci est le gestionnaire d'évènements du programme. Cela aide Dash à attendre que le bouton 1 soit enfoncé. Ensuite, le code sous le bloc *Quand* commence. »

5. Demandez: « Comment les évènements ont-ils changé le code? »

Demandez aux élèves de partager leurs suppositions à tour de rôle.

Puis, dites: « Comme les gestionnaires d'évènements recherchent toujours des évènements, leur utilisation nous permet de modifier le comportement du programme même après avoir appuyé sur le bouton *Marche*. Par exemple, si nous exécutions notre programme sans le bloc *Quand*, Dash ne ferait qu'avancer et tourner à gauche quatre fois. Cependant, puisque nous avons utilisé un gestionnaire d'évènements (*Quand* Dash bouton 1), nous avons pu changer la direction dans laquelle Dash a tourné. L'utilisation de gestionnaires d'évènements nous permet de coder Dash pour effectuer davantage de tâches. »

6. Demandez aux élèves volontaires de changer l'évènement à tour de rôle dans le bloc *Quand*. Exécutez le programme à chaque fois afin qu'ils puissent observer l'évolution du comportement de Dash. Si le temps le permet, vous pouvez sélectionner des élèves volontaires pour ajouter davantage de blocs *Quand* au programme.



Bref retour

- « Pourquoi devrions-nous utiliser des gestionnaires d'évènements dans nos programmes? »

Exemple de réponse: « Nous devrions utiliser des gestionnaires d'évènements afin de pouvoir modifier notre programme en cours d'exécution. »

- « Qu'est-ce qu'un évènement? »

Exemple de réponse: « Un évènement est une action qui provoque quelque chose. »

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.

Formes en boucles

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

- Changer la séquence.
- Ajoutez d'autres éléments ou modifiez les blocs *Quand* (c'est-à-dire des gestionnaires d'évènements).
- Ajoutez des lumières, des sons ou des animations aux boucles.

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 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:

C 2.1: L'œil scintillant de Dot

C 2.2: Le spectacle de Dot

C 2.3: À toi de jouer!

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.

Espace Puzzles de Dot

1. Demandez aux élèves de résoudre le défi de Dot *La Blague du robot en colère*. Vous pouvez trouver des solutions et des extensions pour le défi ici :
<https://drive.google.com/open?id=1d83D2uT9hIP7RV6aY4vSoeZAXgz0Q7L>.
2. Demandez aux élèves de résoudre le défi de Dot *Le Robot chaton*. Vous pouvez trouver des solutions et des extensions pour le défi ici :
<https://drive.google.com/open?id=16BY2JHqjywUTILi-4RmREcMetGhwnlbb>.
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 un gestionnaire d'évènements à notre programme afin que ce soit plus intéressant. »
 - Partager tous les obstacles et les difficultés qu'ils ont surmontés pendant l'activité. Par exemple, « Nous ne savions pas comment faire réagir Dash à un bouton différent, mais nous avons ensuite ajouté un bloc *Quand* Dash bouton 2. »
 - 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 un nouveau gestionnaire d'évènements? »
 - « J'aime la façon dont vous avez utilisé ce que vous avez appris auparavant pour ajouter des répétitions à vos évènements. »

Questions de suivi/discussion

- « Comment les gestionnaires d'évènements vous ont-ils aidé avec votre programme? »
- « Pourquoi les gestionnaires d'évènements sont-ils importants lors de l'utilisation d'un téléphone? »
Exemple de réponse: « Les gestionnaires d'évènements sont importants, car les téléphones doivent constamment chercher à savoir si on appuie sur un bouton ou si on appuie sur l'écran. »

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!

C4

Durée : 90 minutes



Évènements : gestionnaire d'évènements partie 2



by Wonder Workshop

f Share

Tweet



La description

Niveau C - Leçon 4

Les élèves identifieront pourquoi nous utilisons des gestionnaires d'évènements et utiliseront efficacement divers gestionnaires d'évènements. Ils utiliseront les programmes préprogrammés Blockly et les cartes de défis pour mettre en pratique leurs nouvelles compétences.

Les objectifs

Les élèves vont :

- Comprendre quand utiliser les gestionnaires d'évènements.
- Utiliser des gestionnaires d'évènements pour réviser le code et résoudre les problèmes de codage dans un processus itératif.

Ce dont vous aurez besoin

- Dot
- Une tablette numérique par groupe
- Projecteur ou écran interactif avec possibilité de projection à l'écran d'un TNI
- Cartes de défis: C 2.4, C 2.5, C 2.6 (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 et B
- Les leçons 1 à 3 du niveau C
- Cartes de défis des niveaux A et B: toutes [facultatif]
- Cartes de défis du niveau C: B 1.1 à 2.3 [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 événements

1. Passez brièvement en revue la manière dont les élèves ont utilisé les événements avec Dash lors de la leçon précédente en demandant:
 - « Que sont les événements? »
Exemple de réponse: « Un événement est une action qui provoque quelque chose. Dans un programme, les événements sont des objets que l'ordinateur recherche constamment. »
 - « Comment avons-nous utilisé les gestionnaires d'événements dans la leçon précédente? »
Exemple de réponse: « Lorsque nous avons terminé *Formes en boucles*, nous avons appuyé sur la touche *Marche* pour que Dash se déplace dans un sens et sur le bouton 1 pour que Dash se déplace différemment. »
 - « Comment avez-vous changé le gestionnaire d'événements? »
Exemple de réponse: « Nous avons appuyé sur le bloc *Quand* Dash bouton 1 et choisi un autre événement. »
 - « Quel est le but des gestionnaires d'événements? »
Exemple de réponse: « Les gestionnaires d'événements nous aident à contrôler le moment exact où quelque chose se produit dans notre programme. »

Connexions code.org [facultatif]

1. Examinez la leçon de Code.org sur les événements: <https://studio.code.org/s/coursesec-draft/stage/11/puzzle/1>
2. Après avoir terminé la leçon, demandez:
 - « Comment les événements ont-ils été utilisés dans ce jeu? »
Exemple de réponse: « Les événements ont été utilisés pour contrôler l'oiseau. »
 - « Comment pouvez-vous modifier les boutons et les événements qu'ils contrôlent? »
Exemple de réponse: « Au lieu d'utiliser *Quand Click*, vous pouvez créer un nouveau gestionnaire d'événements appelé *Quand Barre Spatiale*. »

Réalisation

Introduction

1. Dites: « Les gestionnaires d'événements permettent au code d'écouter ou d'attendre plusieurs événements pendant l'exécution du programme. Le code sous chaque gestionnaire d'événements ne s'exécute que lorsque cet événement se produit. Aujourd'hui, nous allons utiliser plus d'un gestionnaire d'événements dans notre programme. »
2. Présentez Dot et demandez: « Quels sont les différents gestionnaires d'événements que nous pourrions utiliser pour Dot? »
Exemple de réponse: « Vous pouvez utiliser *Quand Dot secouer!* »

Brève vérification

- « Quand utilise-t-on les événements? »
- « Quels types de blocs peuvent être placés sous un bloc Quand? »

Exemple de réponse: « Vous pouvez placer des blocs de répétition, de son et d'animation sous le bloc *Quand*. Vous pouvez également placer beaucoup d'autres blocs! »

Activité collective

Old McDonald

1. Dans l'application *Blockly*, allez au menu en haut à gauche de l'écran.
2. Allez dans le menu *Créer nouveau* et sélectionnez le programme prédéfini *Old McDonald*. Puis appuyez sur *Créer*.
3. Demandez aux élèves de regarder le programme et de prédire ce qui, selon eux, va se produire lorsqu'il sera exécuté.

Exécutez le programme. Demandez aux élèves volontaires d'appuyer sur les boutons, d'appliquer ou de secouer Dot pour vérifier si leurs prédictions sont correctes.

4. Demandez: « Comment pensez-vous que nous pourrions changer les gestionnaires d'évènements? » Sélectionnez des élèves volontaires pour modifier les évènements dans les blocs *Quand* et exécutez le programme après chaque modification afin qu'ils puissent voir la différence. Encouragez les élèves à essayer également de changer les blocs de son afin que Dot émette différents sons d'animaux.



Bref retour

- « Pourquoi utilisons-nous des évènements? »
Exemple de réponse: « Les évènements nous permettent de contrôler ce qui se passe pendant l'exécution de notre programme. »
- « Quels sont les différents types de gestionnaires d'évènements pouvant être utilisés avec Dot? »
Exemple de réponse: « Vous pouvez utiliser un bloc *Quand Dot bouton du haut*. »

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.

Zoo qui?

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

- Ajouter un autre gestionnaire d'évènements.
- Ajouter/changer les blocs de son.

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>

- Ajoutez des lumières ou des animations.

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:

C 2.4: L'heure du dodo

C 2.5: Réveillez-vous

C 2.6: Lumières éteintes!

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.

Espace Puzzles de Dot

1. Demandez aux élèves de résoudre le défi de Dot *La Boule magique Dot*. Vous pouvez trouver des solutions et des extensions pour le défi ici: <https://drive.google.com/open?id=1aLHIctxtJIIyjmlXkXoXYlxGF8l2vDjZd>
2. Demandez aux élèves de résoudre le défi *Dot le robot spatial*. Vous pouvez trouver des solutions et des extensions pour le défi ici: https://drive.google.com/open?id=1oR3LUq9Ra05_e7SfbE2BBxBguixaDIRP
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 avons ajouté plus de gestionnaires d'évènements parce que nous voulions inclure plus de sons d'animaux pour Dot. »
 - Partager tous les obstacles et les difficultés qu'ils ont surmontés pendant l'activité. Par exemple, « Nous n'étions pas certains de savoir comment changer les gestionnaires d'évènements, mais ensuite nous avons ouvert le bloc *Quand* et avons vu les autres évènements que nous pourrions utiliser. »
 - 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 changé le gestionnaire d'évènements? »
 - « J'aime la façon dont vous avez rendu les sons de l'animal de Dot plus complexes en ajoutant des lumières et des animations. »

Questions de suivi/discussion

- « Comment les gestionnaires d'évènements peuvent-ils rendre les sons d'animaux de Dot plus intéressants? »
Exemple de réponse: « Je pourrais ajouter plus de sons d'animaux en ajoutant plus de gestionnaires d'évènements. »
- « Quels autres blocs pourriez-vous ajouter pour rendre les sons d'animaux de Dot plus excitants? »
Exemple de réponse: « Animation, lumière ou blocs sonores. »
- « En quoi le programme serait-il différent si vous n'utilisiez pas de gestionnaires d'évènements? »
Exemple de réponse: « Le programme serait exécuté dans un ordre spécifique. En utilisant des gestionnaires d'évènements, nous pouvons contrôler la séquence des sons d'animaux. »

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!](#)

C5

Durée : 50 minutes



Boucles : les boucles multiples



by Wonder Workshop



Share



Tweet



La description

Niveau C - Leçon 5

Les élèves utiliseront plusieurs boucles dans un programme. Ils utiliseront les programmes préprogrammés Blockly et les cartes de défis pour mettre en pratique leurs nouvelles compétences.

Les objectifs

Les élèves vont :

- Comprendre comment et quand utiliser plusieurs boucles.
- Utiliser plusieurs boucles pour réviser le code et résoudre les problèmes de codage dans un processus itératif.

Ce dont vous aurez besoin

- Dot
- Une tablette numérique par groupe
- Projecteur ou écran interactif avec possibilité de projection à l'écran du TNI
- Cartes de défis: C 3.1, C 3.2, C 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 et B
- Les leçons 1 à 4 du niveau C
- Cartes de défis des niveaux A et B: toutes [facultatif]
- Cartes de défis du niveau C: B 1.1 à 2.6 [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.

Mise en situation

Activation des connaissances antérieures : les boucles

1. Passez brièvement en revue la manière dont les élèves utilisaient les boucles précédemment en demandant:
 - « Comment les boucles ont-elles été utilisées dans le programme de la Danse carrée? »
Exemple de réponse: « Dash a créé une danse carrée en répétant les blocs **Vers l'avant** et **Tourner à droite** quatre fois. »
 - « Quel est le but des boucles? »
Exemple de réponse: « Les boucles rendent notre programme plus facile et plus rapide à écrire. »
 - « Pourquoi les blocs de boucle ont-ils la forme d'un C? »
Exemple de réponse: « Les blocs de boucle sont formés de cette façon, de manière à pouvoir placer les blocs souhaités à l'intérieur. »

Connexions code.org [facultatif]

1. Examinez la leçon de Code.org sur les boucles: <https://curriculum.code.org/csf/courseb/9/>
2. Après avoir terminé la leçon, demandez:
 - « Y a-t-il une limite au nombre de boucles que nous pouvons ajouter à un programme? Comment le savez-vous? »
Exemple de réponse: « Il n'y a pas de limite au nombre de boucles que nous pouvons ajouter. Dans ce programme, nous avons pu utiliser autant de boucles que nécessaire. »
 - « Comment aurait été votre programme si vous n'aviez pas utilisé de boucles? »
Exemple de réponse: « Le programme aurait été beaucoup plus long si nous n'avions pas utilisé de boucles. »

Réalisation

Introduction

1. Dites: « Parfois, vous avez besoin de plus d'une boucle pour résoudre un problème. »
2. Passez en revue des exemples concrets de problèmes nécessitant plusieurs boucles. Par exemple:
 - « Lorsque vous jouez d'un instrument, vous répétez parfois différentes parties de la musique à des moments différents. Vous pouvez répéter la musique pour chaque strophe trois fois et répéter le refrain trois fois. Cela nécessiterait deux boucles différentes. »
 - « Lorsque vous jouez au basket-ball, vous utilisez une boucle pour dribbler avec le ballon plus d'une fois. Ensuite, vous utilisez une autre boucle pour lancer le ballon à travers le panier plusieurs fois. »
3. Demandez: « Quand voudrions-nous utiliser plus d'une boucle lorsque nous programmons Dash ou Dot? »
Exemple de réponse: « Nous ne voulons pas plus d'une boucle si nous voulons que Dash répète un motif lumineux, mais si nous voulons une deuxième série de motifs lumineux répétés, il faudrait alors une deuxième boucle. »

Brève vérification

- « Quand avez-vous besoin d'utiliser plus d'une boucle dans la vie réelle? »

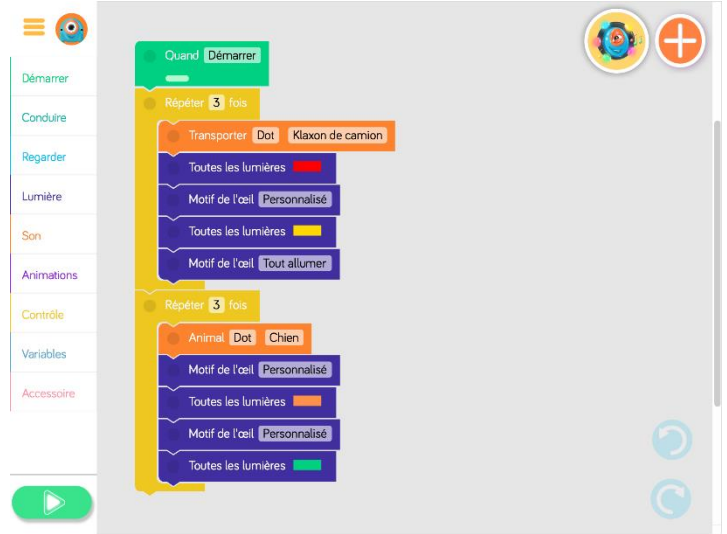
Exemple de réponse: « En musique, nous devons répéter différentes parties de la musique à des moments différents. »

- « Comment pourrions-nous utiliser plus d'une boucle lors de la programmation de Dash ou Dot? »
Exemple de réponse: « Nous pourrions demander à Dot de répéter certains sons et ALORS de répéter un motif lumineux. »

Activité collective

Les figures légères de Dot

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 **Les figures légères de Dot**. Puis, appuyez sur **Créer**.
3. Demandez aux élèves de regarder le programme et de prédire ce qui, selon eux, va se produire lorsqu'il sera exécuté. Exécutez le programme et voyez si les prédictions des élèves étaient correctes.
4. Dites: « Dans ce programme, nous utilisons deux boucles pour répéter différents motifs. Quels sont les deux modèles que nous répétons? »



Exemple de réponse: « Nous répétons un son de klaxon de camion suivi de feux rouges et jaunes dans un motif et d'un son de chien suivi de feux oranges et verts dans l'autre motif. »

5. Demandez: « Que se passerait-il si nous changions le nombre spécifié dans le bloc de répétition? »
Exemple de réponse: « Le nombre de répétitions du code à l'intérieur du bloc changerait en fonction du nombre spécifié dans le bloc de répétition. »

Modifiez le numéro du bloc **Répéter** et demandez aux élèves d'observer et de partager ce qui se passe.

6. Demandez: « Que se passerait-il si nous changions le premier bloc **Répéter** en un bloc **Répéter sans cesse**? »

Exemple de réponse: « Si nous modifions le premier bloc **Répéter** en un bloc **Répéter sans cesse**, le code à l'intérieur de ce bloc ne cesserait jamais de se répéter et nous n'arriverions jamais au code à l'intérieur du deuxième bloc **Répéter**. »

Remplacez le premier bloc **Répéter** par un bloc **Répéter sans cesse** et exécutez le programme.

Demandez aux élèves d'observer et de partager ce qui s'est passé.

7. Demandez: « Comment pourrions-nous changer le motif des yeux? »
Exemple de réponse: « Nous pourrions changer le motif des yeux en choisissant le bloc **Motif de l'œil** et en appuyant les phares que nous voulons allumer ou éteindre. »

Bref retour

- « Si vous souhaitez utiliser plusieurs blocs **Répéter** dans votre code, le premier bloc doit-il être un bloc **Répéter** ou un bloc **Répéter sans cesse**? Pourquoi? »

Exemple de réponse: « Il devrait s'agir d'un bloc *Répéter*. S'il s'agissait d'un bloc *Répéter sans cesse*, je ne me rendrais jamais au deuxième bloc de répétition, car le bloc de *Répéter sans cesse* se répète jusqu'à ce que je le fasse arrêter. »

- « Combien de blocs de répétition pouvez-vous utiliser dans votre code? »

Exemple de réponse: « Vous pouvez ajouter autant de blocs de répétition que vous le souhaitez. »

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.

Les figures légères de Dot

Demandez aux élèves d'utiliser la *Feuille de planification* pour concevoir des moyens d'ajouter ou de modifier le programme prédéfini *Les figures légères de Dot*. Ils peuvent:

- Changer le motif de l'œil.
- Ajouter/changer les sons.
- Changer la couleur des lumières.
- Ajouter des animations.

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:

C 3.1: Vaisseau hors de contrôle!

C 3.2: Aide, aide, aide!

C 3.3: Sauvetage de robot!

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 avons ajouté un autre bloc de répétition à notre programme pour pouvoir répéter un motif différent. »
- Partager tous les obstacles et les difficultés qu'ils ont surmontés pendant l'activité. Par exemple, « Nous ne savions pas comment allumer et éteindre les phares, mais nous avons réalisé que nous pouvions les activer ou les désactiver en appuyant sur les formes autour de l'œil. »
- 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 animation? »
 - « J'aime la façon dont vous avez ajouté un autre bloc *Répéter* au programme. »

Questions de suivi/discussion





- « À quoi ressemblerait notre programme si nous ajoutions plusieurs boucles sous un bloc *Quand*?
Exemple de réponse: « Dot ne réaliserait les boucles que lorsqu'elles seraient déclenchées par l'évènement du bloc *Quand*. »
- « Comment les blocs de répétition pourraient-ils aider Dot à créer deux modèles de lumière différents? »
Exemple de réponse: « Lorsque les blocs de lumière étaient dans le bloc de répétition, les lumières se répétaient encore et encore pour créer un motif. Nous pourrions utiliser deux blocs de répétition pour créer deux motifs de lumière différents. »

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!](#)

Gestionnaires d'événements



Événements	Actions
 1	
 2	
 3	
	

Nom (s) : _____