



v1.0 **Projet CER-J-O**

CLASSE préING1 • 2023-2024

AUTEURS Eva ANSERMIN - Romuald GRIGNON

E-MAILS eva.ansermin@cyu.fr - romuald.grignon@cyu.fr

DESCRIPTION DU PROJET

C'est bientôt les JO !

Votre but est de développer un programme permettant à l'entraîneur de l'équipe d'athlétisme de France de suivre l'évolution des performances de ses sportifs.

Les différentes performances des athlètes lors des entraînements seront stockées dans des fichiers. Votre programme doit pouvoir permettre de les mettre à jour, de consulter certains résultats ainsi que de produire quelques statistiques sur les athlètes.

Ce projet est donc un projet de type gestion qui va notamment nécessiter une bonne maîtrise :

- des fichiers
- des structures
- des tris

INFORMATIONS GENERALES

Taille de l'équipe

Ce projet est un travail d'équipe. Il est autorisé de se réunir en groupe de 3 au maximum. Si le nombre d'étudiants n'est pas un multiple de 3 et/ou si des étudiants n'arrivent pas à constituer des groupes, c'est au chargé de TD de statuer sur la formation des groupes. Pensez donc à anticiper la constitution de vos groupes pour éviter des décisions malheureuses.

Démarrage du projet

Vous obtiendrez de plus amples informations quant aux dates précises de rendu, de soutenance, les critères d'évaluation, le contenu du livrable, ..., quand le projet démarrera officiellement.

Dépôt de code

Vous devrez déposer la totalité des fichiers de votre projet sur un dépôt central Git. Il en existe plusieurs disponibles gratuitement sur des sites web comme github.com ou gitlab.com.

		<p>Rapport du projet Un rapport écrit est requis, contenant une brève description de l'équipe et du sujet. Il décrira les différents problèmes rencontrés, les solutions apportées et les résultats. L'idée principale est de montrer comment l'équipe s'est organisée, et quel était le flux de travail appliqué pour atteindre les objectifs du cahier des charges. Le rapport du projet peut être rédigé en français.</p> <p>Démonstration Le jour de la présentation de votre projet, votre code sera exécuté sur la machine de votre chargé(e) de TD. La version utilisée sera la dernière fournie sur le dépôt Git avant la date de rendu. Même si vous avez une nouvelle version qui corrige des erreurs ou implémente de nouvelles fonctionnalités le jour de la démonstration, c'est bien la version du rendu qui sera utilisée.</p> <p>Organisation Votre projet complet devrait (dans l'idéal) être stocké sur un dépôt git (ou un outil similaire) tout au long du projet pour au moins trois raisons : éviter de perdre du travail tout au long du développement de votre application, être capable de travailler en équipe efficacement, et partager vos progrès de développement facilement avec votre chargé(e) de projet. De plus il est recommandé de mettre en place un environnement de travail en équipe en utilisant divers outils pour cela (Slack, Trello, Discord, ...).</p>
	<p>CRITERES GENERAUX</p>	<ul style="list-style-type: none"> • Le but principal du projet est de fournir une application fonctionnelle pour l'utilisateur. Le programme doit correspondre à la description en début de document et implémenter toutes les fonctionnalités listées. • Votre code sera généreusement commenté. • Tous les éléments de votre code (variables, fonctions, commentaires) seront écrits dans la même langue. Langue anglaise conseillée mais pas obligatoire. • Votre application ne doit jamais s'interrompre de manière intempestive (crash), ou tourner en boucle indéfiniment, quelle que soit la raison. Toutes les erreurs doivent être gérées correctement. Il est préférable de d'avoir une application stable avec moins de fonctionnalités plutôt qu'une application contenant toutes les exigences du cahier des charges mais qui plante trop souvent. Une application qui se stoppe de manière imprévue à cause d'une erreur de segmentation ou d'une exception, par exemple, sera un événement très pénalisant. • Votre application devra être modulée afin de ne pas avoir l'ensemble du code dans un seul et même fichier par exemple. Apportez du soin à la conception de votre projet avant de vous lancer dans le code. • Le livrable fourni à votre chargé(e) de TD sera simplement l'URL de votre dépôt Git accessible publiquement. Même si vous n'avez pas utilisé ce dépôt régulièrement au cours du projet, le code final sera livré dessus.



FONCTIONNALITES DU PROJET

- Votre programme doit pouvoir gérer les épreuves suivantes :
 - Le 100 m
 - Le 400 m
 - Le 5000 m
 - Le marathon
 - Le relais 4*400 m

Pour éviter de démultiplier les épreuves, votre programme permettra de gérer uniquement des sportives ou des sportifs (un seul genre).
- Les différentes performances d'un athlète seront stockées dans un unique fichier qui portera le nom du sportif. Nous aurons donc **un fichier** ⇔ **une personne**. Vous pouvez utiliser des noms de personnes existantes (sportives ou non) ou bien en inventer.
- Dans un fichier lié à un sportif sera indiqué les différents entraînements et performances à ces entraînements. **Un entraînement correspondra à une ligne dans le fichier**. Les informations stockées doivent être à minima :
 - La date de l'entraînement
 - Le type d'épreuve
 - Le temps de l'athlète à cet entraînement.

Vous déciderez de la manière dont vous stockerez ces valeurs-là. Elles ne doivent pas forcément être lisibles humainement entre deux exécutions, mais votre programme doit pouvoir les récupérer et les traiter.
- Cas particulier du relais : une épreuve de relais implique 4 personnes. Afin de pouvoir stocker les infos des participant et connaître le déroulement de l'épreuve, une ligne d'informations dans un fichier concernant un entraînement de relais indiquera également, en plus des autres informations, la position de l'athlète pendant le relais. De plus, il ne peut y avoir qu'un seul entraînement de relais par jour. Ainsi en connaissant la date de l'entraînement vous pouvez retrouver le déroulement complet de l'entraînement du relais.
- Lors de l'exécution du programme, un entraîneur doit pouvoir :
 - Entrer et sauvegarder les informations/résultats d'un nouvel entraînement
 - Consulter l'historique des entraînements : par athlète et/ou par épreuve et/ou par date.
 - Consulter des statistiques de performances de chaque athlète.
- **Statistiques sur les athlètes** : l'entraîneur doit pouvoir consulter
 - Un résumé des performances d'un athlète : le meilleur, le pire temps et la moyenne d'un athlète pour une épreuve donnée.
 - Qui envoyer au JO : quels sont les trois meilleurs athlètes pour une discipline (meilleurs temps moyen).
 - Progression de l'athlète : afficher la différence de temps d'un athlète pour une même épreuve entre deux dates.
- L'évolution des performances : afin de pouvoir observer la progression des athlètes, on aimerait pouvoir indiquer la différence de temps pour une même épreuve entre deux entraînements (entre deux dates).

		<ul style="list-style-type: none"> • Amélioration possible : <ul style="list-style-type: none"> - (Difficile) pour mieux voir les progrès d'un athlète calculer la régression linéaire (équation de droite) de l'évolution d'une performance (temps réalisé lors de l'épreuve) en fonction de la date de l'épreuve. Pour cela, nous vous conseillons d'utiliser la méthode des moindres carrés (voir ressources utiles). - Lors de l'affichage de l'historique, l'entraîneur peut choisir dans quel ordre afficher les information (par date, ordre alphabétique de joueur etc.) - Rajout d'épreuve. - D'autre améliorations de votre choix peuvent être apportées !
	<p>RESSOURCES UTILES</p>	<p>Github</p> <ul style="list-style-type: none"> • www.github.com • https://docs.github.com/en/get-started/quickstart/hello-world <p>Modification des couleurs (polices et fond) dans le terminal</p> <ul style="list-style-type: none"> • http://sdz.tdct.org/sdz/des-couleurs-dans-la-console-linux.html <p>Régression linéaire</p> <ul style="list-style-type: none"> • https://fr.wikipedia.org/wiki/R%C3%A9gression_lin%C3%A9aire
<p>Eva ANSERMIN - Romuald GRIGNON • 2023-2024 • préING1 • eva.ansermin@cyu.fr - romuald.grignon@cyu.fr</p>		