Se connecter
Proposer un contenu
Identifiant
Identifiant
Mot de passe
Mot de passe
Connexion automatique
Se connecter
Pas de compte ? S'inscrire...

Journal : le clean code (je suis peut etre debil)

27 nov. 2026

-5

Posté par **CodeNapol** le 27 novembre 2026 à 20:43. **Licence CC By-SA.**
Étiquettes : **programmation**, **dev**, **clean-code**, **génie_logiciel**, **ingénierie_logicielle**, **tdd**, **kiss**



Bonjour ! je ne sais pas vraiment si ça a sa place dans un journal mais je trouve ça intéressant et le pire c'est que vous connaissez peut-être déjà ce dont je vais vous parler :/

J'ai découvert le terme/pratique du "clean code" qui consiste en gros à écrire du code propre (un code maintenable, facile à modifier et dans lequel il est facile de corriger des bugs) la majorité des ressources que j'ai trouvées sur le clean code expliquent surtout les bases qu'on voit à chaque apprentissage d'un langage de programmation mais il y a quand même des abréviations intéressantes à connaître et une vidéo YouTube de 1h !!! en particulier va me servir de base utile quand je participerai à des projets GitHub pour essayer d'être le plus irréprochable possible... Bref voilà les termes intéressants :

1) KISS qui veut dire "Keep it simple, stupid" en français ça donne "Reste simple, idiot !"

2) DRY pour "Don't repeat yourself" en français "Ne te répète pas"

3) YAGNI en gros "Tu n'en auras pas besoin" en anglais "You aren't gonna need it" Tu code pas un truc inutile

4) Code smell quit veut dire Odeur de code, c'est quand une fonction est un peux trop longue ou qu'un truc nous parrait bizzare qu'elle que part

5) TDD pour "Test-Driven Development" en français "Développement piloté par les tests" on écrit le test avant le code principale

6) etc... ce sont surtout des abréviations anglaises pour tout et n'importe quoi mais je trouve ça fou de ne pas avoir connu tous ces termes.

d'ailleurs, voici un article IONOS sur le sujet : ionos.ca et une video de 1H en français d'un mec sûrement trop geek "https://www.youtube.com/watch?v=VioMQpZMtnA"

Je sais que tout le monde a déjà le réflexe de penser à ça mais prenez ça comme un rappel et comme un petit boost pour votre vocabulaire ;)

(4 commentaires). Markdown EPUB

lien en plus

Posté par **CodeNapol** le 27 novembre 2026 à 20:47.
Évalué à -1 (+0/-0).

[^] # Re: lien en plus

Posté par **BAud** (**site web personnel**) le 27 novembre 2026 à 20:55. Évalué à 3 (+2/-1). Dernière modification le 27 novembre 2026 à 20:56.

> dont la troisième page qui indique 
>
> > Aucune représentation ou reproduction, même partielle, autre que celles prévues à l'article L. 122-5 2° et 3° a)du code de la propriété intellectuelle ne peut être faite sans l'autorisation expresse de Pearson Education France
> > ou, le cas échéant, sans le respect des modalités prévues à l'article L. 122-10 dudit code
>
> avant de coder proprement, une petite connaissance des licences (libres de préférence) est sans doute utile...

Répondre

Ouai

Posté par **barmic** le 27 novembre 2026 à 21:44.
Évalué à 3 (+0/-0).

> Bref voilà les termes intéressants
>
> Je trouve personnellement qu'ils sont utiles mais simplistes. De mon point de vu leur usage ne doit pas être un mantra, mais une manière de communiquer un choix qui a était fait à un moment donné. Le YAGNI pour l'un des exemples les plus parlants. Comment tu distingue le YAGNI d'une adaptabilité logiciel ? En passant à ce concept qu'est-ce qu'écrire du clean code pour mon script que je vais jeter dans 1h et pour celles et ceux qui envoient leur code dans l'espace ? Combien de bug existent parce qu'une situation est sensé ne jamais arriver ?
>
> Une autre manière de le montrer c'est qu'il existe la primitive obsession mais aussi l'overengineering, tout est dans le compromis et ça c'est ta connaissance, ton expérience et un peu de toi. J'ai beaucoup travaillé avec un très grand ami. On voyait les même problèmes, on faisait le même cheminement et nos conclusions étaient fréquemment opposées (par exemple si ce problème a eu lieu il faut que ce soit entièrement automatisé vs il faut accepter que ce soit sous contrôle humain).
>
> > TDD pour "Test-Driven Development" en français "Développement piloté par les tests" on écrit le test avant le code principale

C'est vraiment imprécis, la valeur n'est pas dans le fait d'écrire les tests avant mais dans le cycle itératif qui permet :

- de faire des petites étapes (baby step)
- d'avoir une grande couverture de code
- d'appliquer d'une certaine manière le YAGNI

Ce que tu décris c'est la famille des tests-first dont le TDD fait parti mais il n'est pas le seul.

--

https://linuxfr.org/users/barmic/journaux/y-en-a-marre-de-ce-gros-troll

Répondre

Vendredi

Posté par **barmic** 🦫 le 27 novembre 2026 à 21:47.
Évalué à 3 (+0/-0).

Robert C Martin l'auteur du livre clean code : https://agileway.substack.com/p/z-on-robert-c-martins-popular

--

https://linuxfr.org/users/barmic/journaux/y-en-a-marre-de-ce-gros-troll

Répondre

Envoyer un commentaire

Suivre le flux des commentaires

Note : les commentaires appartiennent à celles et ceux qui les ont postés. Nous n'en sommes pas responsables.

Revenir en haut de page

Derniers commentaires

- avis personnel
- Re: mr ?
- Vendredi
- Ouai
- Déjà payant
- Re: Personne peu fr…
- Re: bulls**t
- Tous les indicateurs …
- Re: Personne peu fr…
- Re: Personne peu fr…
- Re: Personne peu fr…
- Re: lien en plus

Étiquettes (tags) populaires

- intelligence_artificielle
- slop
- sécurité
- merdification
- datacenter
- grands_modèles_de…
- disquette
- vibe_coding
- réchauffement_clim…
- cybersécurité
- linux
- microsoft

Sites amis

- Agenda du Libre
- April
- Éditions D-BookeR
- Éditions Diamond
- Éditions Eyrolles
- Éditions ENI
- En Vente Libre
- Framasoft
- La Quadrature du Net
- Lea-Linux
- Open Source Initiative
- Imprimerie Grafik Plus

À propos de LinuxFr.org

- Mentions légales
- Faire un don
- L'équipe de LinuxFr.…
- Informations sur le s…
- Aide / Foire aux que…
- Suivi des suggestion…
- Wiki du site
- Règles de modération
- Statistiques
- API pour le dévelop…
- Code source du site
- Plan du site