

📌 More than 3 years have passed since last update.

 @rana_kualu

偉大なプログラマ(+人間)になるための101のTIPS

ポエム プログラマ 日本語訳

Last updated at 2021-01-18 Posted at 2019-07-22

早く人間になりたい。

以下は[Emma Wedekind](#)による記事、[101 Tips For Being A Great Programmer \(& Human\)](#)の日本語訳です。

101 Tips For Being A Great Programmer (& Human)

1. Get good at Googling

Googleの使い方を知ろう。

プログラマであるということは、問題に対する答えを探す能力があるということです。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✔ You can read useful information later.

Login

Sign up

Later



47



55



タスクに2週間かかるようなら3週間かかると伝え、2週間で提出しましょう。
期待を上回った結果を出すことで、信頼を築くことができます。

3. Be nice to your designers; they're your friends

デザイナーと絆を築く。彼らは友達です。

デザイナーはユーザ視点の問題点に解決策を提供してくれます。
それらを学び、効果的な製品を作るために仲間として働きましょう。

4. Find a mentor

メンターを探せ。

何かを学ぶことができる師匠を見つけ、共に切磋しましょう。
それを始めるのに、[Coding Coach](#)はいい場所です。

5. Be a mentor

メンターになれ。

何かを教えることができる弟子を見つけ、共に切磋しましょう。
それを始めるのに、[Coding Coach](#)はいい場所です。

6. Write useful comments

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

適切な名前を付ける。

関数や変数の名前はそれらの目的を適切に表すべきであり、`myCoolFunction` という名前ではうまくいかないでしょう。

8. Take vacations

休暇を取れ。

我々は皆、リフレッシュする時間が必要です。
行きたかったところに旅行しましょう。
同僚とあなたの脳はあなたに感謝するでしょう。

9. Delete unused code

いらぬコードは削除せよ。

技術的負債を残しておく理由はありません。

10. Learn to read code

コードを読み。

コードを読むスキルは過小評価されがちですが、非常に貴重なものです。

11. Establish a healthy work/life balance

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

12. Only schedule necessary meetings

必要な会議だけに出る。

その会議はメールやSlackで変わりになりませんか？

もしそうであれば、会議を避けてください。

そうでなければ出なければなりません、できるだけ減らすことを目指してください。

13. Pair program

ペアプロしろ。

ペアプログラミングでは、教師と生徒両方の立場を同時に経験することができます。

14. Write great emails

よいメールを書く。

メールの文章は簡潔で明確に書き、読み手にとって読みやすいことを意識しましょう。

4ページにわたるメールなど誰も読みません。

15. Get involved in the community

コミュニティに参加する。

志を同じくする人と同じ釜の飯を食うことは、あなたが成長するきっかけになるでしょう。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✔ You can read useful information later.

Login

Sign up



47

55

友達が遊びに来る前に自宅を片付けるように、バージョン管理のブランチも片付けてください。

必要のないブランチは、押し入れに突っ込むのではなく捨てます。

17. Don't gate keep

排他的になるな。

寛容でありなさい。

業界に入る能力がないなどと言ってはいけません。

誰もがその価値を持っています。

18. Keep learning

学び続けなさい。

あなたは、継続的な学習を必要とする職種を選びました。

それを至上としなければなりません。

19. Don't give up

諦めるな。

いつも簡単にことが進むわけではありません。

しかし、皆同じところから始まりました。

あなたも辿り着くことができます。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

21. Clarify requirements before starting

要件を明確にせよ。

コードを書く前にはっきりさせましょう。

要件の省略は一時的に時間を節約するかもしれませんが、後で苦労します。

22. Have a toolbox

ツールを使おう。

裏の裏まで知っているツールをひとつは持っておいてください。

どのツールは何の目的のためにあるのか、ある目的のためにはどのツールを使うのが効率がよいか、知りましょう。

23. Learn to love constructive criticism

建設的な批判を受け入れる。

信頼できる同僚や友人に、建設的な批判意見を求めましょう。

それはあなたがプログラマとして、そして人間として成長するのを助けます。

24. Be open-minded

柔軟な頭を持つ。

技術は常に、急速に流転します。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

出版物、ブログ、ポッドキャスト、ニュースなどをフォローし、新技術についていってください。

26. Focus on problem solving

問題解決に集中する。

難しい問題を解決するスキルは、他の問題も克服することができます。
問題解決のために必要な技術を磨いてください。

27. Stay humble

謙虚だなーあこがれちゃうなー。

どんな立派な肩書きを持っていたとしても、どんな会社に勤めていようとも、謙虚さを忘れてはいけません。

28. Learn to give a great presentation

プレゼンテーション能力を磨く。

聴衆を魅了する、効率の良いプレゼンテーションを行う方法を学びましょう。

29. Examine all solutions before jumping in

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

自分の専門を見つけよう。

ハイテク業界には多くの分野があります。

自分が最も興味のある分野を見つけ、そのエキスパートになりましょう。

31. Develop good habits

良い習慣を身につける。

健康で、集中を乱さない、一貫性のある習慣を身につけましょう。

タイムボクシングを採用し、最も重要なタスクから仕事を始めます。

慣れるまで時間はかかるかもしれませんが、長期的には行う価値があります。

32. Learn to debug

デバッグの方法を学ぶ。

ブラウザのデバッガツールを調べます。

IDEを使ったデバッグの詳細について調べます。

効率的にエラーを追跡する方法を学ぶことで、どんなに困難なバグであっても解決することができるようになるでしょう。

33. Exercise your current skills

持っているスキルをさらに磨こう。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

34. Understand the why

理由を理解しよう。

あなたは自分の意見を表明しなければならないときは来るでしょう。

そのときには、その理由を理解していることが重要です。

ソリューションAがソリューションBより優れているのは何故ですか？

根拠のある理由を展開することで、あなたの意見は大きく補強されるでしょう。

35. Know your worth

自分の価値を知る。

あなたは商品であり、従って価格は適切に決定されるべきです。

あなたの住んでいる地域における業界の平均値に注意を払いましょう。

あなたの価格がそれより低いのであれば、上司に直談判する時間です。

仕事に相当する価値を受け取るべきです。

36. Don't be afraid to ask for help

助けを求めることを恐れない。

何らかの問題を抱えていて、解決策の調査に時間がかかりすぎているとき、それは助けを求めるときです。

我々は皆人間であり、人間はみな誰かの助けが必要です。

助けってもらうために手を伸ばすことは、恥ではありません。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

ているかもしれません。
あなたの学習スタイルを見つけ出しましょう。

38. Be kind

親切であれ。

何らかの問題について同僚に質問されることがあるかもしれません。
丁寧に答えましょう。

39. Take breaks

休憩を取れ。

8時間連続してコーディングし続けることはほぼ不可能です。
あなたはすぐに燃え尽きるし、多くの間違いを埋め込むでしょう。
だから、休みを取ることを忘れないようタイマーを設定しましょう。
散歩する、同僚とコーヒーを飲む。
いちど画面から離れることで、生産性や作業の質により影響が現れます。

40. Track your progress

進捗を管理せよ。

コードの学習には時間がかかり、先に進んでないと感じるととてもがっかりすることがあります。

この記事があなたの成長に目標の進捗状況を追跡することをサポートします。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✔ You can read useful information later.

Login

Sign up



47

55

フレームワークに頼りすぎない。

ひとつのフレームワークやライブラリについて根掘り葉掘り調べるよりも、それぞれの違いについて調べましょう。

必ずしも片っ端からフレームワークやライブラリを学ぶ必要はありません。

しかし、それらが何故そのようになっているのかを理解することは、あなたがよりクリーンで高性能なコードを書くことを助けてくれるでしょう。

42. Learn to love code reviews

コードレビューを愛せ。

誰かがあなたのコードを読んで分析することを恐ろしいことだと感じるかもしれませんが。

しかしそれは、あなたをより良いプログラマにするためのフィードバックを得られる貴重な機会なのです。

また逆に、より良いコードレビューを行う能力も養いましょう。

43. Learn about tangential spaces

関連することからも学ぶ。

デザイン、マーケティング、フロントエンド開発やバックエンド開発など、関連することからの基礎も学びましょう。

それはあなたがより偉大なプログラマになることを助けてくれるでしょう。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

以前に使ったことのあるテクノロジーを使い回すのは簡単ですが、それらがプロジェクトのニーズに合わない場合は、代替手段を検討しなければなりません。

45. Take responsibility for your mistakes

ミスに責任を負う。

全ての人間は間違いを犯します。

あなたもキャリアの至る所で間違いをしてきました。

重要なことは、間違いを犯したときに、自分でその責任を取ることです。

そうすることで、あなたは同僚や上司からの信頼を得ることができるでしょう。

46. Review your own code

自分のコードを見返す。

プルリクを開く前に、自分のコードを見返してみましょう。

それが同僚のコードだったとしたら、あなたはどのようなコメントをしますか？

コードレビューを依頼する前に、まず問題点や間違いを洗い出してみることが肝要です。

47. Learn from your failures

失敗から学ぶ。

失敗とは、単に期待された結果を達成できなかったということであり、必ずしも悪いこ

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

弱点を認識せよ。

自分を知りましょう。

自分の弱点はどこですか？

PUSHする前にテストを更新するのを忘れること？

それともメールの返信が苦手なこと？

自分の弱点を知り、それを克服するために取り組みましょう。

49. Stay curious

好奇心を保ち続けよ。

この業界は常に進化し続けているので、ついていくには好奇心が重要になります。

もし理解できていないところがあるなら、プロジェクト要件であろうとコードの行であろうと、そのことを話してください。

説明を求めたとしても、そのことであなたを批判する人はいませんし、結果としてより良いコードができあがります。

50. Don't try to learn everything

全てを学ぶことはできない。

世の中の知識は無量大であり、それらを全てマスターすることは不可能です。

いくつかの分野についてだけ深く習熟し、残りは見切りましょう。

他分野についても表面的な、あるいは実用的な知識を付けることは有用ですが、しかし全てを習得することはおそらくできないでしょう。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✔ You can read useful information later.

Login

Sign up



47

55

クルというものが存在します。
いつまでもそれに固執する必要はありません。

52. Have your team's back

背中を預ける戦友を持て。

良いチームは、互いを助け合います。
これにより、失敗を恐れることなく新しいことにチャレンジする居場所ができます。

53. Find inspiration in the community

コミュニティからインスピレーションを得よう。

業界から、尊敬に値する人々を見つけましょう。
彼らから、現在の仕事を続けるか、新しいことに挑戦するか、刺激を受けることができるでしょう。

54. Value your work

仕事に価値を与えよ。

経験の多寡や役職にかかわらず、あなたのしている仕事には価値があります。
自分のしている仕事に対して、それに相応しい価値を与えましょう。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✔ You can read useful information later.

Login

Sign up



47

55

56. Be supportive

支えになる。

重要なプレゼンテーションを行うとき、何らかに行き詰まったとき、そんなときにはチームメンバーをサポートしましょう。

57. Give credit where credit is due

功績があれば当然認めてやれ。

誰かが素晴らしい仕事をしているなら、それを口に出して言いましょう。

ポジティブなフィードバックはチームメンバーとの信頼形成に役立ち、彼らのキャリアを育てる素晴らしい方法です。

彼らも同様にあなたを賞賛することでしょう。

58. Test your code

コードはテストせよ。

テストは重要です。

単体テスト、回帰テスト、総合テスト、E2Eテスト。

コードをテストすることで、コードはずっと安全になることでしょう。

59. Plan out your approach

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

60. Learn to pseudocode

擬似コードの使い方を学ぶ。

擬似コードを使うことで、複雑な問題について多くのコードを書くこともなく整頓して考えることができます。

擬似コードを書き下ろし、様々なテストケースを考え、コードに落とし穴が存在しないか確認します。

61. Keep track of your achievements

業績を記録しよう。

職場で表彰されたら、それを記録しよう。

重要な機能を開発したら、それを記録しよう。

記録は昇進の助けになり、また辛い日のやる気の元になります。

62. Learn programming foundations

プログラミングの基礎を学べ。

基本的なソートや検索のアルゴリズム、データ構造を学びましょう。

これらは言語に縛られることなく、言語を横断して問題解決に役立ちます。

63. Choose technology for longevity & maintainability

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

64. Learn design patterns

デザインパターンを学べ。

デザインパターンはコードを構築するための便利なツールです。

全てのプロジェクトでそれらが必要になるとは限りませんが、しかしデザインパターンの基礎を理解していれば、より大規模なアプリケーションを構築する際の助けとなります。

65. Reduce ambiguity

曖昧さを減らす。

プログラミング能力の高さをひけらかす短くて複雑なコードを書くかわりに、読みやすくシンプルでコードを目指しましょう。

それはチームメンバーのためでもあります。

66. Pay off technical debt

技術的負債を解消する。

技術的負債はパフォーマンスに大きな影響を与えることがあるため、リファクタリングが可能であるならば、すべきです。

67. Ship often

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

68. Commit early and often

コミットは小さく、早く。

小さな粒度で頻繁にコミットすることが、作業履歴を綺麗に保ち、重要な変更をリバー
トすることによる精神的ダメージを軽減するための最良の方法です。

69. Learn when to ask for help

助けを求めるタイミングを測る。

助けを求めることを恐れてはいけないのと同時に、いつ助けを求めるかを知ること
も大事です。

何もしないうちに助けを呼ぶことは避けるべきであり、助けを求めた後も挑戦はし続
けなければなりません。

しかし、単純そうな問題に一時間以上苦戦するようであればコストがベネフィットを上
回ってしまいます。

助けを求めるべきときです。

70. Ask effective questions

効率的に質問する。

質問するときは、できるだけ具体的に詳しく伝えましょう。

71. Get feedback on unfinished work

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

72. Read documentation

公式ドキュメントを読み。

あらゆる技術について、公式ドキュメントは最も正確な情報源です。
従って、公式ドキュメントを読むことこそがエキスパートになるための近道です。

73. Try all the things

あらゆることを試せ。

問題に対して様々な解決策を試すことを妨げるものは何也没有ありません。
試すことで何か失うものがあるのですか？

74. Speak up in meetings

ミーティングで発言せよ。

あなたのアイデアや意見は貴重であり、会議に能動的に参加することが、あなたやチームと経営層との関係を改善することに役立ちます。

75. Collaborate cross-team

チームを超えて共同作業せよ。

社内の他のチームと一緒に仕事をする機会があれば、是非参加しましょう。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✔ You can read useful information later.

Login

Sign up

週10時間以上働いた後は、自分の趣味のプロジェクトに時間をかけることが重要で



47

55

きなかったかもしれない新しい技術に挑戦することもできます。

77. Define your career goals

キャリアのゴールを定めよ。

理想的なキャリア形成の目標を持つておくことは重要です。

さもないければ、あなたはターゲットもなく銃を撃とうとしているようなものです。

78. Get involved in the conversation

コミュニティに参加する。

ブログにコメントし、Twitterの会話に参加し、コミュニティと交流しましょう。

積極的な見物人は、壁の花よりも多くのことを学べます。

79. Prioritize tasks

タスクの優先順位付け。

タスクに優先順位を付け、生産性向上に役立てます。

毎日の直近のタスクと、より長期的な予定のタスクをリストアップし、最も重要なものから並べましょう。

80. Don't overlook the details

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

チームメンバーを信頼する。

あなたのチームメイトはそのスキルが故に雇われました。
仕事を成し遂げるために彼らを使い、彼らを信頼しましょう。

82. Learn to delegate

委任することを覚える。

あなたが指導的ポジションにいるのであれば、効率的に委任する方法を学びましょう。
それによってあなたの時間やフラストレーションを節約できます。
あなた一人で全てを行うことはできません。

83. Don't compare yourself to others

自分を他人と比べない。

あなたが比較しなければならない唯一の人物は、昨日のあなたです。

84. Surround yourself with allies

仲間を作れ。

プログラムを学ぶことは、長く容易ではない旅路です。
共に歩き続ける仲間を作り、支え合いましょう。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up

スクリーンショットを今晩に書いて開発することは有用ですが、スクリーンショットは必要のな



47

55

そうすることで、必要のない無駄な開発で開発者の心を折ることもなく、それでもあなたは成長することができます。

86. Weigh performance implications

パフォーマンスへの影響を考慮する。

ツールで最新のテクノロジーを使用したくなったら、その前にまずパフォーマンスへの影響を調べる必要があります。

同じことを、パフォーマンスを犠牲にすることなく実現できますか？

そうであるなら、そのテクノロジーの採用を再考することをお勧めします。

87. Don't discriminate

差別するな。

新しいアイデアや技術を差別してはいけません。

新しいスキルを否定せず、可能性に対して心を広く持ってください。

また人を差別してはいけません。

我々はみんな尊敬に値します。

88. Apply for jobs you aren't qualified for

資格がなくても求人に応募する。

仕事の全ての要件を満たす人はいません。

あなたがその条件をすべて満たす必要は

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

全てのコードをひとつのファイルに記述することはできますが、それでは保守できません。

モジュール化することで、コードの見通しが良くなり、テストも容易になります。

90. Don't JUST copy and paste

単なるコピペはNG。

Stack Overflowからソリューションをコピペするつもりであれば、あなたはそれを**完璧**に理解していなければなりません。

コピペしたいコードの意図を理解してください。

91. Create an inspiring environment/setup

良い環境・仕組みを作る。

作業場や仕事の仕組みが楽しいものになれば、仕事へのやる気も上がるでしょう。

あなたがそれをするのです。

92. Remember where you came from

初心を忘れないで。

我々はみんな同じところから始まりました。

あなたのスキルや役職がどれだけ進化しても、最初はどうだったかを忘れないでください。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✔ You can read useful information later.

Login

Sign up



47

55

楽観主義はあなたのチームの活力や精神的健康の糧になります。

94. Continually re-assess your workflow

ワークフローは継続的に見直す。

今日うまく行ったからといって、明日もそれがうまくいくとは限りません。
ワークフローを再評価し、必要に応じて修正しましょう。

95. Learn how to work from home

リモートワークの方法を学ぶ。

リモートワークが可能なのであれば、効率的にそれを行う方法を知りましょう。
気を散らすものを隔離して、専用の仕事場を作りましょう。
Boneskullがリモートワークについて[素晴らしい記事](#)を書いています。

96. Code for accessibility

アクセシビリティの高いコードを書く。

容易にアクセスできる必要があり、そしてそれは難しいことではありません。
誰もがあなたのプロダクトを使える必要があります。

97. Honor your commitments

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

98. Be proactive

積極的に活動せよ。

仕事に余裕があるなら、チームを支援するタスクを見つけて実行しましょう。
チームメンバーの大きな助けになります。

99. Build an amazing portfolio

アメイジングなポートフォリオを作れ。

良いポートフォリオは、あなたを他の候補者から目立たせます。
あなたのデザインとプログラミングのセンスを存分に発揮するチャンスです。

100. Remember why you love programming

なぜプログラミングが好きなのか、覚えておこう。

あなたはそれに魅了され、この業界に入ったのでしょうか。
不平や不満が溜まってきたときは、休息を取ってください。
プログラミングに対する情熱を再燃させる期間を取りましょう。

101 Share your knowledge

知識を共有する。

クールな何かを学んだら、それを共有しましょう。

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up

おわり！



47

55

います。

コメント欄

102 Do Physical Exercises

運動する。

週3回以上、できれば毎日、運動しましょう。

我々の仕事は、何年もほとんどは座りっぱなしです。

これは身体に深刻な悪影響を与えることでしょう。

気に入ったスポーツを見つけましょう。

103 Sleep enough!

寝ろ。

睡眠は人生における最優先事項だ。

質の良い睡眠を心がけよう。

その他

「100回の腕立て伏せ、100回の腹筋、10キロのランニングで最強のプログラマになれるよ!」「ただしハゲる」

「午前5時に眠り、午前9時に目を覚ますプログラマの数に驚いています。直近ではうまいいのかもしませんが、長期的には関連いたく生販します。」

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

「20番について、私は専門教育を受けてないのでopencvやmatplotlibを怖がって避けていた。でも手を出してみたら素晴らしいものだった。私は自分で記事を書き、その過程で私よりずっと優れた記事を見つけ、そこから得ることができた。」

「2番はよくない。過剰な品質はやがてそれが標準になり、真面目な開発者ほど毒沼に陥る。私のルールは、過剰部分がビジネス要件と一致しているか、簡単に対応できるか、エネルギーを費やす価値がある場合にのみ対応する。」

「休憩するときには水を飲もう。」

感想

多すぎい！

こんなに気をつけて生きていくなんでできねーよ。

そもそも私プログラミング好きじゃないですし、勉強も努力もしたくないし、常日頃からあらゆる方向に気を張っていくなんでできもしないですね。

プログラミングなんて仕事だから仕方なくやってるだけであって、5000兆円手に入ったら即座に仕事辞めて寝て暮らすわ。

だから5000兆円ください。

しかし全てを守るのは無理でも、この中の幾つかでも、できるところを意識しておけば、今後が僅かに変わっていくかもしれません。

最初はほんの少しの変化でも、やがて人生を大きく左右することになるでしょう。

所々英語のスラングなのかなんなのか、意味のよくわからないところが色々あった。きっと誰かがプルリクしてくれるはず。

33番目。

"Just because you've got a six pack now, doesn't mean you can eat a 🍕 a day and stay that wav."

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✔ You can read useful information later.

Login

Sign up

shreds "



47

55

43番目。

"tangential spaces"を調べても接ベクトル空間しか出てこないんだけど、こういうかんじの意味で合ってるのだろうか。

51番目。

タイトルの意味が全くわからないのだが映画のこと？



0



Register as a new user and use Qiita more conveniently

1. You get articles that match your needs
2. You can efficiently read back useful information
3. You can use dark theme

[What you can do with signing up](#)

Sign up

Login

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55



Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✓ You can read useful information later.

Login

Sign up



47

55

Qiita

How developers code is here.

© 2011-2024 Qiita Inc.

Guide & Help

About

Terms

Privacy

Guideline

Design Guideline

Feedback

Help

Advertisement

Contents

Release Note

Official Event

Official Column

Advent Calendar

Qiita Award

API

Career

SNS

✕ @Qiita

✕ @qiita_milestone

✕ @qiitapoi

📘 @Qiita

Our service

Company

Why not login to Qiita and try out its useful features?

- ♥ We'll deliver articles that match you.
- ✔ You can read useful information later.

Login

Sign up



47

55