

# 推測しないで、計測し、判断する！

～カイゼンのためのステップ考察～

PHPカンファレンス福岡2023

2023.06.24

Yuta Ohashi



おおはし ゆうた

# 大橋 佑太



株式会社リンケージ



青ごへいもち / @blue\_goheimochi



PHP(Laravel), JavaScript(Vue.js/Nuxt.js),  
ランニング, サッカー, フットサル,  
浜松まつり, PUBG Mobile,  
さわやかなのげんこつハンバーグ



# 【Mission】

## テクノロジーとつながりで

## 健康意識の温度をあげる



人事必見! テレワークに対応  
ストレスチェック中心メンタルサービス



モニター画面のデータ:

項目	数値	割合
対象者	112人	100%
回答率	98.1%	(109/112)
高ストレス者率	23.2%	(26/112)
総合健康リスク	105	(93/112)

スマートフォン画面の表示:

- 総合評価: D
- あなたの状態: ストレス度が非常に高く、ご自身の健康とパフォーマンスに悪影響を及ぼす可能性があります。専門家のサポートが必要です。今回のセルフケアに留意してセルフケアを継続して取り組むことで、ぜひ少しずつ一つ一つでも実現してみてください。
- 項目別アドバイス: 仕事のストレス要因 (1, 2, 3, 4, 5)



FEMCLE

FEMCLEひとつで、全ての女性社員の健康課題を総合的にサポートします。

監修: 総合母子保健センター愛育病院 病院長 百枝幹雄先生  
協力: NPO法人日本子宮内膜症啓発会議 (JECIE)

あなたを「気にかける」頼れるパートナー



スマートフォン画面の表示:

- かかりつけ保健師 for LINE
- 相談テーマ: ストレス, メンタル不調, 体調不良, 新型コロナウイルス関連, ヘルプデスク, 出産・育児, 産休・育休, 介護
- 相談内容: はじめまして。保健師の田中と申します。これまで我慢されてきて、今は我慢ができない状況にまでなってきたらっしやるんですね。お気持ちをご相談ください、ありがとうございます。





Webアプリケーション開発のためのプログラミング技術情報誌  
FOR ALL WEB APPLICATION DEVELOPERS **ウェブDBプレス**

# WEB+DB

パブリックでも  
オンプレでも!

PRESS

# インフラ技術

vol. 135  
2023

総点検 Docker/LXD  
Kubernetes  
プライベートクラウド

現場で使える  
**Laravel 10**

新機能 | チュートリアル | 落とし穴への対策

Yupによる  
バリデーション

フォームの値を簡単・柔軟に検証

- ▶ ログテーブルとの付き合い方
- ▶ テストコードの認知負荷
- ▶ 実務レベルのGraphQL導入ガイド



e\_goheimochi

cript(Vue.js/Nuxt.js),

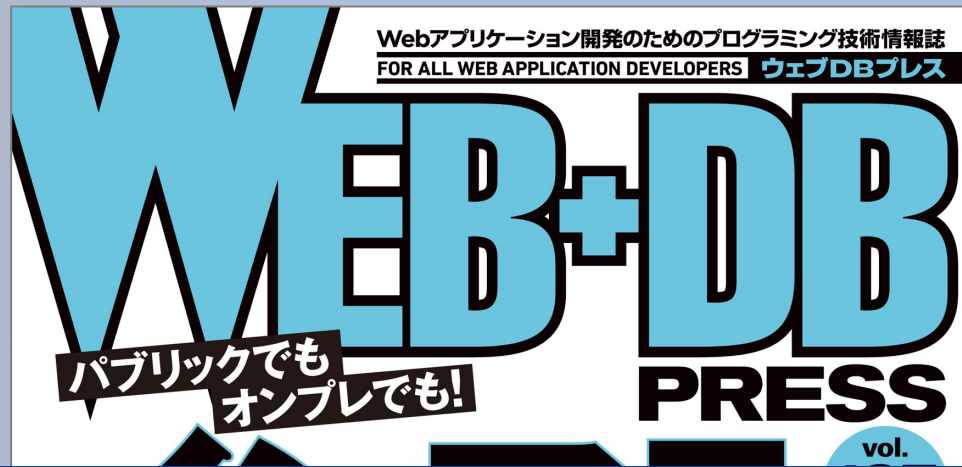
フットサル,

obile,

ハンバーグ

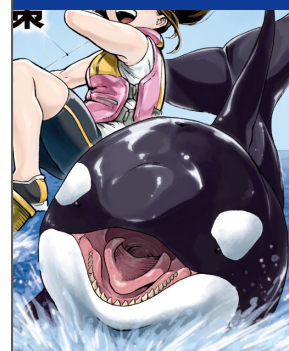
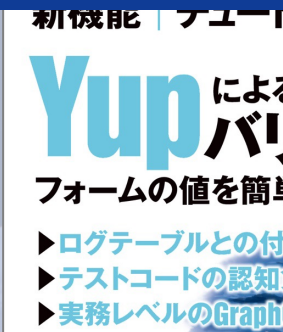






【PR】

6月24日(本日)発売!



ハンバーグ





【PR】

／ヨロシクネツ!!／

／ワイワイ!!／

# 6月24日(本日)発売!



新機能 | ユーザー  
**Yup** による  
 バリ  
 フォームの値を簡単  
 ▶ ログテーブルとの付  
 ▶ テストコードの認知  
 ▶ 実務レベルのGraph



アウトリル、  
 mobile,  
 ハンバーグ



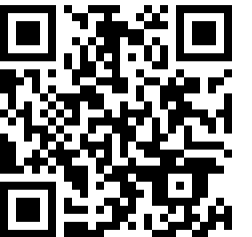
**「推測するな、計測せよ」**



# 「推測するな、計測せよ」

- @afW@a` BdaYdS\_ \_ [ Y [ 5
- ロバート・C・パイク氏がその中で提言した6つのルールの中の2つから生まれた言葉とされることが多そう
- 
- ある事をもとにして**想像によって判断すること**
- 
- 器械を使って、数・量・重さ・長さなどをはかること
  - 「**実際の状況・状態を知ること**」 「**現状把握すること**」 と解釈している

<http://www.lysator.liu.se/c/pikestyle.html>



**想像によって判断をしない**

**事実によって判断をする**



**例: コードのカイゼンをしたい**

# 例: コードのカイゼンをしたい

- - 「推測」の域を出ないケースも多い
- **効果的な有効なリファクタリング**
  - 無意味なリファクタリングにしたくない

# 例: コードのカイゼンをしたい

- 重複コードはあるか？
- 循環的複雑度はどのくらいか？
- 利用していないコードはあるか？
- コーディング規約は守られているか？
- コードにエラーはないか？
- etc...



# 例: コードのカイゼンをしたい

- `, bZbUbV`
- Copy/Paste Detector (CPD) for PHP code.
  - 2023年1月にアーカイブされてしまった…
- Don't Repeat Yourselfとはよく言われるが・
  - コピペコードが必ずしも悪ではないので、検出された対象のコードは目視で確認してみましょう

# 例: コードのカイゼンをしたい

phpcpd 4.1.0 by Sebastian Bergmann.

Found 184 clones with 9893 duplicated lines in 142 files:

/app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:166-524 (358 lines)

184個の重複コードを検出。142ファイル、合計9,893行が対象

- /app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:321-561 (240 lines)  
/app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:334-574

~~~~~

- /app/xxxxxx/controllers/xxxxxx.php:909-1084 (175 lines)  
/app/xxxxxx/controllers/xxxxxx.php:1247-1422

~~~~~

8.41% duplicated lines out of 112932 total lines of code.  
Average size of duplication is 54 lines, largest clone has 358 of lines

# 例: コードのカイゼンをしたい

phpcpd 4.1.0 by Sebastian Bergmann.

Found 184 clones with 9893 duplicated lines in 142 files:

- /app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:166-524 (358 lines)  
/app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:170-528

~~~~~

- /app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:231-561 (330 lines)  
/app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:235-565

Fatかつ重複したコントローラ・・・

~~~~~

- /app/xxxxxx/controllers/xxxxxx.php:909-1084 (175 lines)  
/app/xxxxxx/controllers/xxxxxx.php:1247-1422

~~~~~

8.41% duplicated lines out of 112932 total lines of code.  
Average size of duplication is 54 lines, largest clone has 358 of lines



# 例: コードのカイゼンをしたい

phpcpd 4.1.0 by Sebastian Bergmann.

Found 184 clones with 9893 duplicated lines in 142 files:

- /app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:166-524 (358 lines)  
/app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:170-528

~~~~~

- /app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:321-561 (240 lines)  
/app/xxxxxx/xxxxxx/xxxxxx/xxxxxx.php:334-574

~~~~~

112,932行の中、重複があったコードは8.41%  
重複コードの平均行数は54行で、最大358行が重複している

8.41% duplicated lines out of 112932 total lines of code.  
Average size of duplication is 54 lines, largest clone has 358 of lines

# 例: コードのカイゼンをしたい

- , ##\$!+%\$
- , \*ž&# ,
- , +!\* +%
- , #\* &
- , #&\$

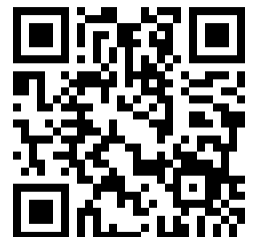
# 例: コードのカイゼンをしたい

- ツール: `terryyin/lizard`
  - Lizard is an extensible Cyclomatic Complexity Analyzer for many programming languages(PHP以外の言語も解析できるPythonのツール)
  - 循環的複雑度を計測できる
- 循環的複雑度
  - if/else, for, switchなどの数を計測
  - コードがどれだけ複雑か?の指標

# 例: コードのカイゼンをしたい

| 循環的複雑度 | 複雑さの状態        | バグ混入確率 |
|--------|---------------|--------|
| 10以下   | 非常に良い構造       | 25%    |
| 30以上   | 構造的なリスクあり     | 40%    |
| 50以上   | テスト不可能        | 70%    |
| 75以上   | いかなる変更も誤修正を生む | 98%    |

<https://szk-takanori.hatenablog.com/entry/20111219/p1>



# 例: コードのカイゼンをしたい

循環的複雑度が63…  
バグ混入確率70%以上…

```
=====  
!!!! Warnings (cyclomatic complexity > 15 or length > 1000 or parameter_count > 100) !!!!!  
=====
```

| NLOC | CCN | token | PARAM | length | location                                                 |
|------|-----|-------|-------|--------|----------------------------------------------------------|
| 247  | 63  | 2436  | 0     | 353    | aaaaAction@41-393@./xxxx/controllers/zzzzController.php  |
| 139  | 41  | 1272  | 0     | 202    | bbbbAction@625-826@./xxxx/controllers/zzzzController.php |
| 124  | 39  | 1018  | 0     | 161    | ccccAction@77-237@./xxxx/controllers/zzzzController.php  |
| 81   | 37  | 351   | 1     | 84     | xxxx@840-923@./xxxx/xxxx/Controller/xxxx/Abstract.php    |
| 152  | 37  | 1086  | 0     | 197    | xxxxAction@21-217@./xxxx/xxxx/xxxx/xxxxController.php    |
| 143  | 36  | 1110  | 0     | 197    | xxxxAction@740-936@./xxxx/xxxx/xxxxController.php        |
| 150  | 35  | 1516  | 0     | 223    | xxxxAction@33-255@./xxxx/controllers/xxxxController.php  |
| 63   | 34  | 741   | 1     | 72     | xxxx@16-87@./xxxx/xxxx/xxxx/xxxx/xxxx/xxxx.php           |
| 81   | 34  | 917   | 0     | 106    | aaaaAction@229-334@./yyyy/controllers/yyyyController.php |
| 110  | 31  | 1071  | 0     | 143    | xxxxAction@20-162@./xxxx/controllers/xxxxController.php  |
| 75   | 30  | 753   | 0     | 108    | bbbb@79-186@./xxxx/controllers/yyyyController.php        |
| 97   | 29  | 926   | 0     | 127    | ccccAction@609-735@./xxxx/controllers/yyyyController.php |

```
=====  
Total nloc   Avg.NLOC   AvgCCN   Avg.token   Fun Cnt   Warning cnt   Fun Rt   nloc Rt  
-----  
33248       16.3       2.3      104.2      1743      11            0.01     0.05  
=====
```



# 例: コードのカイゼンをしたい

```
=====  
!!!! Warnings (cyclomatic_complexity > 15 or length > 1000 or parameter_count > 100) !!!!!  
=====
```

```
NLOC    CCN    token  PARAM  length  location  
-----
```

```
247     63    2436     0     353  aaaaAction@41-393@./xxxx/controllers/zzzzController.php  
139     41    1272     0     202  bbbbAction@625-826@./xxxx/controllers/zzzzController.php  
124     39    1018     0     161  ccccAction@77-237@./xxxx/controllers/zzzzController.php  
 81     37     351     1      84  xxxAction@840-923@./xxx/xxx/Controller/xxx/Abstract.php  
152     37    1086     0     197  xxxAction@21-217@./xxx/xxx/xxx/xxxController.php  
143     36    1110     0     143  xxxAction@740-936@./xxx/xxx/xxxController.php  
150     33    1110     0     150  xxxAction@33-255@./xxx/controllers/xxxController.php  
 63     30     753     0     108  bbbb@79-186@./xxx/controllers/yyyController.php  
 81     29     926     0     127  ccccAction@609-735@./xxx/controllers/yyyController.php  
110     29     926     0     127  ccccAction@609-735@./xxx/controllers/yyyController.php
```

上位3つは「同じ」コントローラー・・・

```
=====  
Total nloc    Avg.NLOC    AvgCCN    Avg.token    Fun Cnt    Warning cnt    Fun Rt    nloc Rt  
-----
```

```
33248    16.3    2.3    104.2    1743    11    0.01    0.05
```

# 例: コードのカイゼンをしたい

- `/ fj ( +!+&(`
- `55@, $ž*`
  - CCNの高いところは先ほどの表と照らし合わせても、保守・メンテが困難な可能性は高い
- `,&! #*`

# 例: コードのカイゼンをしたい

- `private String bZb_V; private void g() { ... }`
- - 未使用のプライベートプロパティ
  - 未使用のローカル変数
  - 未使用のプライベートメソッド
  - 未使用の引数

# 例: コードのカイゼンをしたい

- `, B: B 5E8[j Wd B: B 5aVWE` [XWd`
- `BEDŽ#$` `BEDŽ$`
-

# 例: コードのカイゼンをしたい

- `, B: BEfS` † BeS^_`
- `//` `fi`
-

# 例: コードのカイゼンをしたい

## コードに対する「事実」

- 重複コードはあるか？
- 循環的複雑度はどのくらいか？
- 利用していないコードはあるか？
- コーディング規約は守られているか？
- コードにエラーはないか？
- etc...

# 例: コードのカイゼンをしたい

- ファイルの変更回数？
- アクセス状況は？
- エラー状況は？



# 例: コードのカイゼンをしたい

- `9[f`
- `9[f`
-

# 例: コードのカイゼンをしたい

- , 9aaY^W3` S^kf[Ue
- 
- 
-

# 例: コードのカイゼンをしたい

- ツール: new relic, Datadog, Mackerel, Sentryなど
- APMの状況はどうか??
  - アクセス状況なんかも交えて色々わかる
- エラー発生頻度が高い&アクセスも多いはユーザーに悪い体験を与えている可能性が考えられる
- パフォーマンスが悪い箇所なども優先すべきカイゼン箇所になりうる

# 例: コードのカイゼンをしたい

B: BWA=S[Y[ \$" \$%

計測できるレガシーさを捉え、コード改善に対処する by 大橋 佑太

ホーム - PHPerKaigi 2023 - トーク - 計測できるレガシーさを捉え、コード改善に対処する by 大橋 佑太



The slide features a white background with a green triangle in the top-left corner and a blue triangle in the bottom-right corner. The Fortee logo is positioned in the top-right. The main text is centered, and the PHPerKaigi 2023 logo is in the bottom-left.

**fortee**

計測できるレガシーさを捉え、  
コード改善に対処する

by 大橋 佑太 / @blue\_goheimochi

PHPerKaigi  
2023

<https://fortee.jp/phperkaigi-2023/proposal/6125c4e2-528c-4b6b-b383-ffc8bfeec29a>



# 例: コードのカイゼンをしたい

- 
- 
- **リファクタリングすべき対象コードを決める判断基準になる**
- 
- 
- **それぞれの現場で、チームで「判断」**

**「計測」 → 「事実」 → 「判断」**

例: ランニングのカイゼンをしたい

# たのしい PHPエンジニア 健康生活双六

## スタート!

新入社員!  
頑張るぞ!

ついうっかり  
暴飲暴食気味

仕事に慣れてきた!  
勉強のしすぎで徹夜、  
一回休み

リリース前、いそがしすぎて大変。  
男性なら喫煙しないように注意  
女性なら普段と違う生理痛に注意

## ゴール!

...ですがこれからも健康に気をつけて!  
Linkageは頑張る社会人の皆様に  
医療テックでサポートします。

予防医療テックのリンケージ  
**Linkage**



上の PHPPer をきりとって、あなたのコマにしよう!!!



上の PHPPer をきりとって、あなたのコマにしよう!!!





青ごへいもち  
@blue\_goheimochi



大濠公園DONE 🏃🏻‍♂️👏



午前7:57 · 2023年6月24日 · 229 件の表示

📌 ツイートアナリティクスを表示

12 件のいいね



青ごへいもち  
@blue\_goheimochi



大濠公園DONE 🏃🏻‍♂️👏



午前7:57 · 2023年6月24日 · 229 件の表示

📌 ツイートアナリティクスを表示

12 件のいいね



青ごへいもち  
@blue\_goheimochi



#imafukuokaniirupher



午前0:14 · 2023年6月24日 · 288 件の表示

📌 ツイートアナリティクスを表示

6 件のいいね



たのしい  
PHPエンジニア  
健康生活双六

スタート!

新入社員!  
頑張るぞ!

ついうっかり  
暴飲暴食気味

仕事に慣れてきた!  
勉強のしすぎで徹夜、  
一回休み



リリース前、いそがしす  
男性なら喫煙しないよ  
女性なら普段と違う生理

ゴール!

...ですがこれからも健康に気をつけて!  
Linkageは頑張る社会人の皆様を  
医療テックでサポートします。



ななうえぶ

そーだい

ほんほん  
「どぼう  
茶飲む」  
5 Linkage

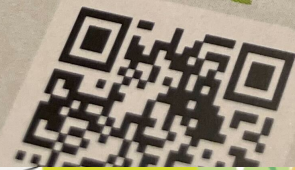


うすら  
「座り過  
ぎない」  
6 Linkage



にしよう!!!

かかりつけ保健師  
for LINE



# 例: ランニングのカイゼンをしたい

&

&

- - 「推測」の域を出ない。
- **効果的な有効なサブ4のためのトレーニング**
  - 無意味なランニングにしたくない

# 例: ランニングのカイゼンをしたい

計測ツールにより知ることができる

「事実」

- 
- 
- 
- 
- WUZZ

# 例: ランニングのカイゼンをしたい

- , [BZa` W 3bb^M SfUZ
- - 時間、距離、消費カロリー
  - 心拍数、最大心拍数、平均心拍数
  - 体の上下動、歩幅、接地時間
  - など

# 例: ランニングのカイゼンをしたい

- 
- 
- **カイゼンすべきトレーニング内容を決める判断基準になる**
- 
- それぞれが「判断」
  - 「循環的複雑度」のような指標が欲しい

# 例: ランニングのカイゼンをしたい



## ジョギング

- 「ゆっくり走る」トレーニング。
- 会話をしながら走れるのが最適なスピードだ。
- 中・上級者であれば2時間から3時間は楽に走れる程度のスピード。
- 有酸素運動の能力を高める効果が期待できる
- 中級者であれば6分～6分半/km



# 例: ランニングのカイゼンをしたい



## ジョギング

ゆっくりってどれくらいよ

- 「ゆっくり走る」トレーニング。
- 会話をしながら走れるのが最適なスピードだ。
- 中・上級者であれば2時間から3時間は楽に走れる程度のスピード。
- 有酸素運動の能力を高める効果が期待できる
- 中級者であれば6分～6分半/km

楽って…

30秒の差ってまあまああるよ?

# 例: ランニングのカイゼンをしたい

- 
- 
- 
- ZonesというiPhoneアプリ使っている
- $211 - (0.64 \times \text{年齢}) = \text{最大心拍数}$

<https://support.garmin.com/ja-JP/?faq=1BuYR3HJIR2QSALSOag8C7>



# 例: ランニングのカイゼンをしたい

| Zone | %MAX    | 大橋      | 状態                                                                                           |
|------|---------|---------|----------------------------------------------------------------------------------------------|
| 1    | 50~60%  | 94~112  | <ul style="list-style-type: none"><li>・心身ともにリラックスしたペース</li><li>・リズムカルな呼吸で会話に支障がない</li></ul>  |
| 2    | 60~70%  | 113~131 | <ul style="list-style-type: none"><li>・快適さを感じるペース</li><li>・少し呼吸が深くなるが会話は可能</li></ul>         |
| 3    | 70~80%  | 132~150 | <ul style="list-style-type: none"><li>・マラソンをするような標準のペース</li><li>・会話を続けるのが難しくなる</li></ul>     |
| 4    | 80~90%  | 151~169 | <ul style="list-style-type: none"><li>・ややきついペース</li><li>・呼吸が力強くなり会話することができない</li></ul>       |
| 5    | 90~100% | 170~188 | <ul style="list-style-type: none"><li>・全速力の速さで、長時間維持することはできないペース</li><li>・呼吸が相当きつい</li></ul> |

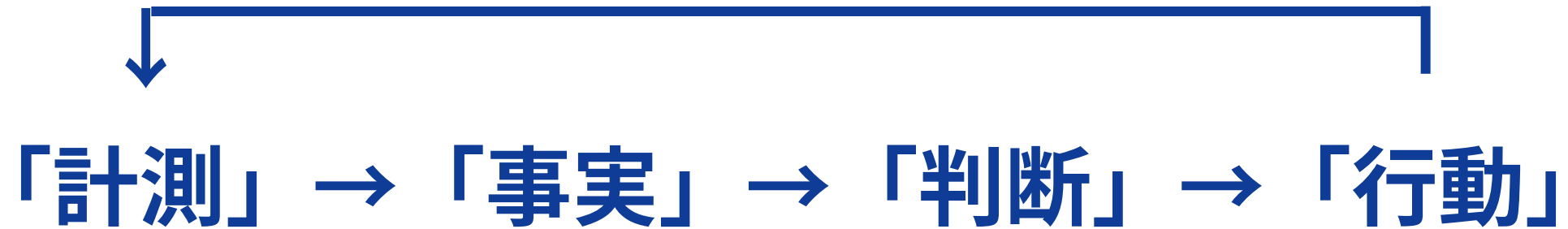
# 例: ランニングのカイゼンをしたい

| Zone | 効果                                                                                                                                              |
|------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| 1    | <p>【有酸素性能力の基礎作り】 = 軽いジョギング、LSD</p> <ul style="list-style-type: none"><li>・ 有酸素性能力向上の初期レベルのトレーニング</li><li>・ ストレスの軽減</li></ul>                    |
| 2    | <p>【心肺機能の向上】 = ジョギング</p> <ul style="list-style-type: none"><li>・ 心肺機能向上の基本となるトレーニング</li><li>・ 脂肪燃焼に効果的</li><li>・ 高強度トレーニング後の休息に適したペース</li></ul> |
| 3    | <p>【有酸素性能力の向上】 = レースペース走、持久走</p> <ul style="list-style-type: none"><li>・ 心肺機能向上に最適なトレーニング</li><li>・ 持久力の向上</li></ul>                            |
| 4    | <p>【無酸素性能力の向上】 = インターバル走、ビルドアップ走</p> <ul style="list-style-type: none"><li>・ 無酸素性作業閾値の向上</li><li>・ スピードの向上</li></ul>                            |
| 5    | <p>【無酸素性持久力の向上】 = 坂ダッシュ、ウィンドウスプリント</p> <ul style="list-style-type: none"><li>・ 瞬発力、筋持久力の向上</li></ul>                                            |

ということで？

**「計測」 → 「事実」 → 「判断」**

**「計測」 → 「事実」 → 「判断」 → 「行動」**





まとめ

# まとめ

- - 想像による判断をしてはいけない
  - 「事実」による判断をする
  - そして「アクション」につなげる
  - 2回目以降はより精度を高くカイゼンのステップをふんでいける
-

# 【Mission】

## テクノロジーとつながりで

## 健康意識の温度をあげる



人事必見! テレワークに対応  
ストレスチェック中心メンタルサービス



モニター画面のデータ:

| 項目      | 数値    | 割合        |
|---------|-------|-----------|
| 対象者     | 112人  | 100%      |
| 回答率     | 98.1% | (109/112) |
| 高ストレス者率 | 23.2% | (26/112)  |
| 総合健康リスク | 105   | (93/112)  |

スマートフォン画面の表示:

- 総合評価: D
- あなたの状態: ストレス度が非常に高く、ご自身の健康とパフォーマンスに悪影響を及ぼす可能性があります。専門家のサポートが必要です。今回セルフケアに励むことが大切です。
- 項目別アドバイス: 仕事のストレス管理 (1.2), 3.4, 5.2, 5.3



FEMCLE

FEMCLEひとつで、全ての女性社員の健康課題を総合的にサポートします。

監修: 総合母子保健センター愛育病院 病院長 百枝幹雄先生  
協力: NPO法人日本子宮内膜症啓発会議 (JECIE)

あなたを「気にかける」頼れるパートナー



スマートフォン画面の表示:

- かかりつけ保健師 for LINE
- 相談テーマ: ストレス, メンタル不調, 体調不良, 新型コロナウイルス関連, ヘルステック, 出産・育児, 産休・育休, 介護
- 相談内容: はじめまして。保健師の田中と申します。これまで我慢されてきて、今は我慢ができない状況にまでなってきたらっしやるんですね。お気持ちをご相談ください、ありがとうございます。

