Capgemini

TDD

Easy

Coding Dojo

Kata: Roman-Decimal

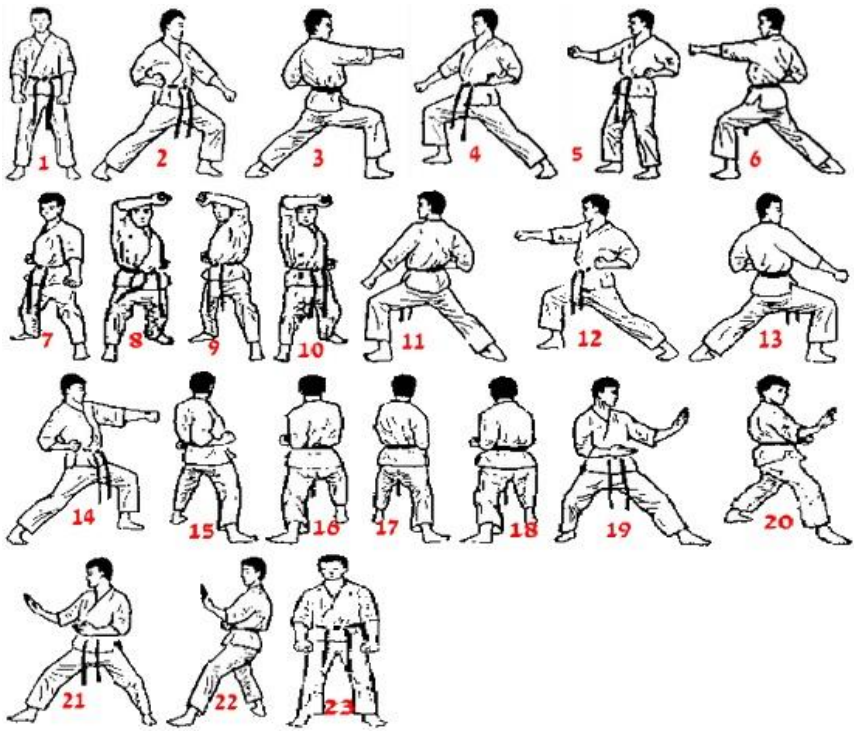
Antes de empezar (¿Qué es coding dojo?)

Dojo es el lugar donde se reúne la gente para practicar y entrenar artes marciales.

CODING DOJO

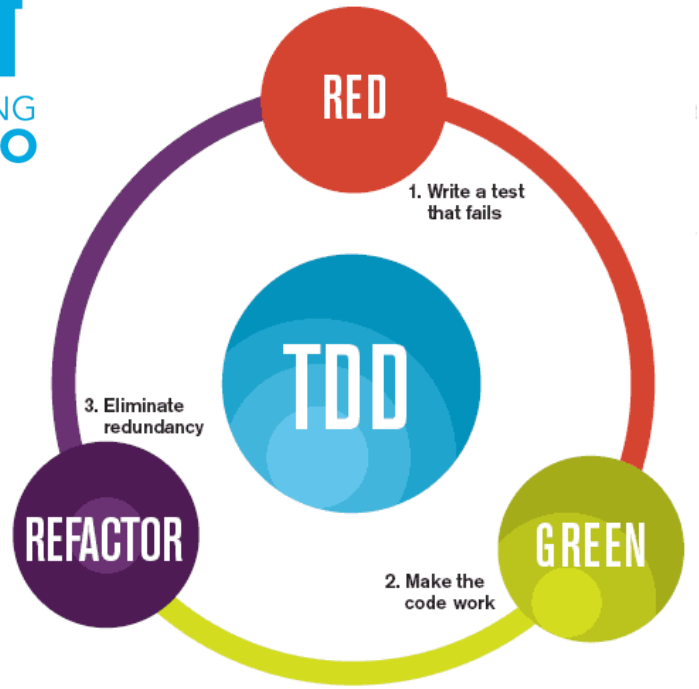
Lugar donde nos reunimos para practicar y entrenar buenas formas de programación.

Antes de empezar (¿Qué es coding dojo?)



Ejecutan Katas para aprender los movimientos y las técnicas.

CODING DOJO



Usamos TDD para el desarrollo

Ejecución por consenso

Antes de empezar (¿Qué es Refactor?)

"Cambios en el código para hacerlo más fácil de entender y más barato de modificar, sin alterar su comportamiento observable"

- Martin Fowler -

Antes de empezar (¿Cómo sabemos que está “mal escrito”?)

Basic smells

COMMENTS

MAGIC NUMBER

LONG METHOD

DUPLICATE METHOD

LARGE CLASS

LONG PARAMETER LIST

Refactor Actions

EXTRACT CLASS

INTRODUCE EXPLAINING VARIABLE

EXTRACT METHOD

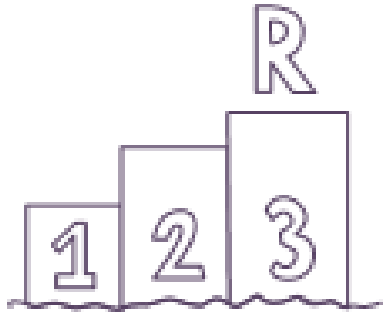
SELF ENCAPSULATE FIELD

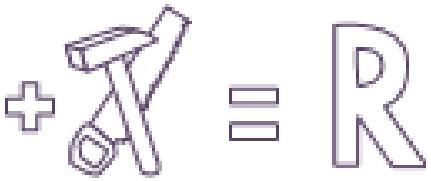
INLINE METHODS

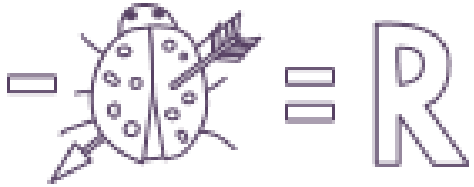
ENCAPSULATE COLLECTIONS

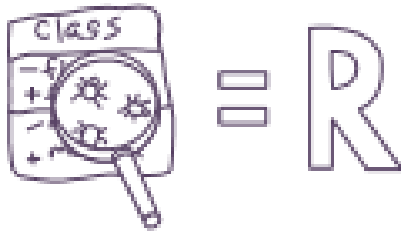
https://refactoring.guru/refactoring/smells

Antes de empezar (¿Cuándo refactorizamos?)









La regla del Boy Scout: “Siempre deja el lugar de acampada más limpio que como lo encontraste”.

https://refactoring.guru/refactoring/smells

Práctica

Kata: Roman-Decimal

Roman-Decimal (Enunciado)

Los romanos eran tipos inteligentes. Conquistaron la mayor parte de Europa y la gobernaron durante cientos de años. Inventaron los caminos de hormigón e incluso los bikinis. Pero una cosa que nunca descubrieron fue el número cero.

Esto hizo que escribir y fechar extensas historias de sus hazañas fuese un poco más desafiante, pero el sistema de números que se les ocurrió sigue en uso hoy en día. Por ejemplo, la BBC utiliza números romanos para fechar sus programas.

Los romanos escribieron números usando letras - I, V, X, L, C, D, M. (observe que estas letras tienen muchas líneas rectas y por lo tanto son fáciles de cortar y tallar en piedra).

Source: https://www.giters.com/millancore/php-study/issues/6

Roman-Decimal (Enunciado)

En esta Kata se debería escribir una función para convertir de números romanos a números decimales. Por ejemplo:

I → 1
VII → 7
X → 10
XIV → 14

No hay necesidad de poder convertir números más grandes que aproximadamente 3000 ya que los romanos no tendían a numerar tan alto.

Tenga en cuenta que no se pueden escribir números como "IM" para 999.

Wikipedia dice: Los números romanos modernos se escriben expresando cada dígito por separado comenzando con el más a la izquierda y omitiendo cualquier dígito con un valor de cero. Dos ejemplos de ello pueden ser los siguientes:

En los números romanos 1990 se representa como: 1000=M, 900=CM, 90=XC → MCMXC.

En los números romanos 2008 se representa como: 2000=MM, 8=VIII → MMVIII.

Roman-Decimal (Reglas)

I = 1

V = 5

X = 10

L = 50

C = 100

D = 500

M = 1000

- Los números romanos siempre se escriben y leen de **izquierda a derecha**, es decir, empezando por los números de mayor valor, y de **dígito en dígito**.
- Los números **seguidos** de otro de valor **igual o menor se suman** siempre, como, por ejemplo, XXII que sería 10 + 10 + 1 + 1= 22.
- Los números **seguidos** de otro de valor **mayor se restan** siempre, como, por ejemplo, XIX que sería 10 + (1 – 10) = 19.
- El número I y sus múltiplos X y C, pueden colocarse delante de un número mayor para restar valor, pero sin repetirse. Además, solo pueden restar valor a números inmediatamente superior, pero no a valores mucho más elevados. Es decir, I solo puede restar a V y X, X a L y C, y C a D y M.
- El número V y sus múltiplos de 10, como L y D no se pueden usar para restar en ningún caso. Por ejemplo, el número cuarenta y cinco se escribe XLV y no VL.

Roman-Decimal (Consejos)

- Se debe intentar realizar el test de uno en uno.
  - Desde lo más sencillo a lo más complejo
- Se debe intentar implementar el código mínimo necesario para cada test.
  - Aunque se conozca el problema completo, hay que evitar el "*Speculative generalization*"
- En cada implementación de un test, se debe refactorizar el código.
  - ¿*me siento cómodo con este código*?
  - ¿*cambiaría la implementación a objetos*?
  - ¿*cambiaría la implementación con herencia*?
  - ¿*es la mejor estructura de datos que se podría utilizar*?
- Si ha resultado fácil, probad a realizar la función inversa.
  - Una función que dado un número normal lo convierta a un número romano.

People matter, results count.

About Capgemini

With more than 190,000 people, Capgemini is present in over 40 countries and celebrates its 50th Anniversary year in 2017. A global leader in consulting, technology and outsourcing services, the Group reported 2016 global revenues of EUR 12.5 billion. Together with its clients, Capgemini creates and delivers business, technology and digital solutions that fit their needs, enabling them to achieve innovation and competitiveness. A deeply multicultural organization, Capgemini has developed its own way of working, the Collaborative Business Experience™, and draws on Rightshore®, its worldwide delivery model.

Learn more about us at

www.capgemini.com