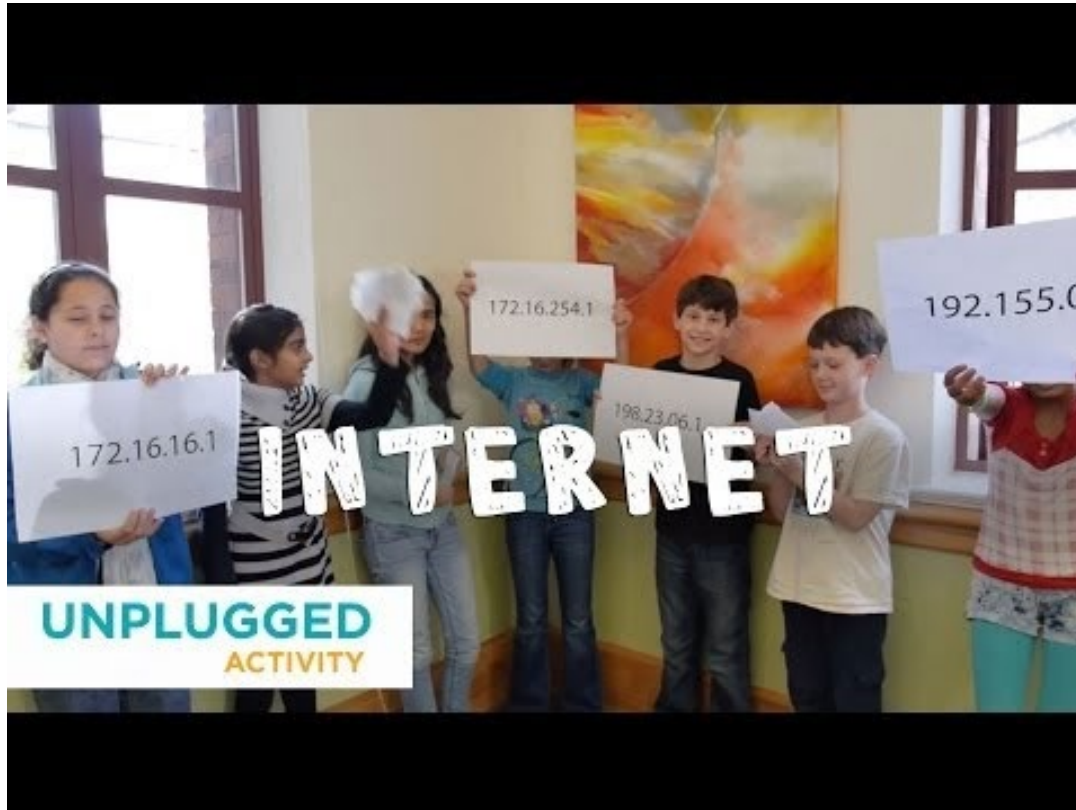


Code.org

Carrera
Programador
full-stack

Introducción

Lección 18: Internet



Lección 18: Internet

Que es?

Es una red descentralizada de computadoras alrededor del mundo conectadas entre sí mediante distintos tipos de enlaces, por ej satélites, radios, telefonía, etc.



Lección 18: Internet

Algunas características:

- **Universal:** Está presente en todo el mundo.
- **Interactiva:** Podemos compartir opiniones, fotos, archivos, entre muchas otras cosas.
- **Libre:** Cualquiera puede acceder a ella.
- **Integral:** Puede ser utilizada con muchos fines, desde la medicina hasta los videojuegos por ejemplo.

Lección 18: Internet

Conceptos importantes:

- **URL (Uniform Resource Locator):** Dirección única que se asigna a cada uno de los recursos disponibles de la World Wide Web para que puedan ser localizados por el navegador.
- **IP (Internet Protocol):** Es un conjunto de números que identifica a un dispositivo conectado a una red, por ej: 192.168.0.1
- **DNS (Domain Name System):** Es la traducción de dominios web como www.ejemplo.com a numeros IP compatibles con nuestro ordenador por ej: 192.168.0.1
- **MAC (Media Access Control):** Es un identificador único para cada pieza de hardware de red, por ej: 2C:54:91:88:C9:E3

Lección 19

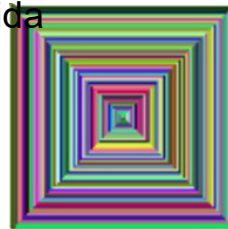
Recuerden usar la
opción **MOSTRAR**
EL CÓDIGO!

En esta lección vamos a jugar a ser artistas, acá no hay respuestas correctas o incorrectas, todas las soluciones son válidas, simplemente vamos a ejecutar el código que nos dan y modificarlo utilizando lo aprendido para ver cómo se comporta.

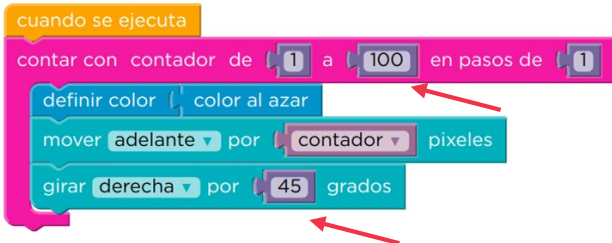
Si ejecutamos este código



Tenemos esta
salida



Cambiamos grados e iteraciones



Este es el resultado

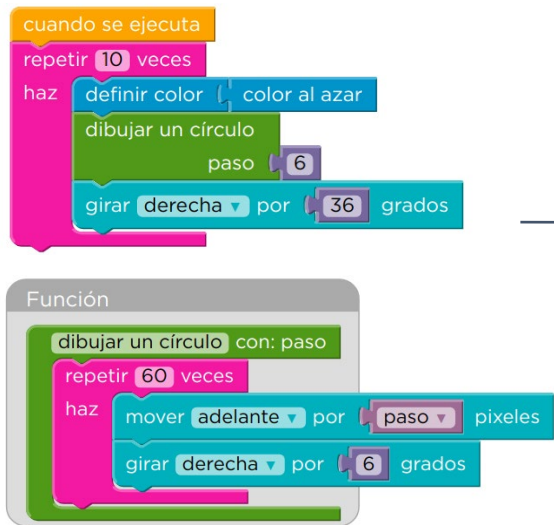


Lección 19

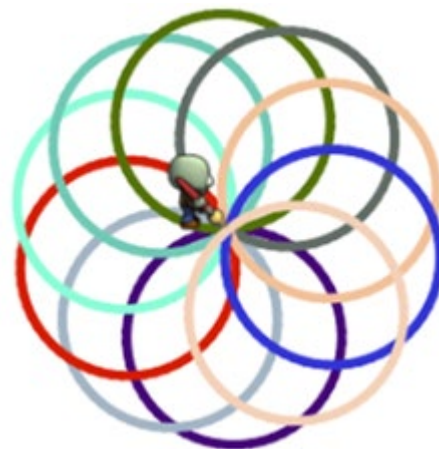
Recuerden usar la
opción **MOSTRAR**
EL CÓDIGO!

Veamos otro ejemplo

En el ejercicio 4 tenemos el siguiente código



Salida



Lección 19

Recuerden usar la
opción **MOSTRAR**
EL CÓDIGO!

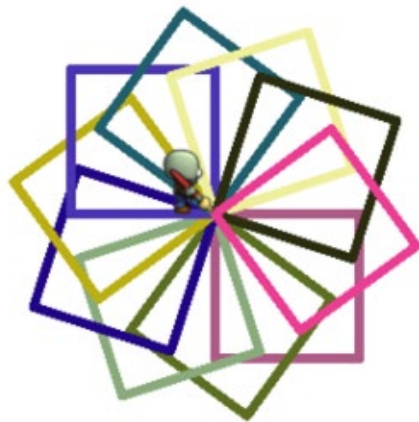
Veamos otro ejemplo

Cambiamos la función 'dibujar circulo' por 'dibujar cuadrado'



Modificamos los pasos para que
haya más distancia entre cuadrados

Salida



Lección 19

Te animas a modificar los demás ejercicios y ver que se puede lograr?

A ponerse creativo!



Lección 20



Que significa código eficiente?

Como ya sabemos, un problema tiene muchas formas diferentes de ser resuelto, algunas son mejores que otras, a esto es a lo que llamamos código eficiente.

Veamos un ejemplo:

Instrucciones

Ayúdame a llenar el agujero en el otro extremo del campo, con la menor cantidad posible de bloques.

cuando se ejecuta

girar a la derecha

mientras haya camino adelante

haz avanzar

girar a la izquierda

mientras haya camino adelante

haz avanzar

girar a la izquierda

mientras haya camino adelante

haz avanzar

mientras hay un agujero

haz llenar 1

cuando se ejecuta

mientras haya camino adelante

haz avanzar

llenar 1

Cuál solución es más eficiente?

Ventajas de trabajar con código eficiente

- Mejora la legibilidad del código.
- Hace más fácil su mantenimiento.
- Mejor performance.
- Facilita las revisiones de código.
- Lleva menos tiempo escribirlo.
- Menos propenso a bugs.
- Facilita agregar funcionalidades nuevas.
- Suelen ser soluciones un poco más desafiantes de implementar.
- No hay que perder tiempo a futuro implementando “refactors” (mejoras) de código.
- Entre otras.

Cada Lección una entrega

- En el desarrollo de un producto el cliente solicita funcionalidades y el equipo se compromete a entregarlas en cierto plazo, el cual suele ser largo ya que una funcionalidad o módulo completo conlleva muchas tareas y análisis de por medio.
- Una vez que la funcionalidad esté completa o en un estado de “madurez” se arma un paquete de “release” o lanzamiento y se publica para el usuario final
- Veamos cada lección como una funcionalidad nueva e imaginemos que el cliente quiere que desarrollemos un nuevo módulo llamado “La Granjera”

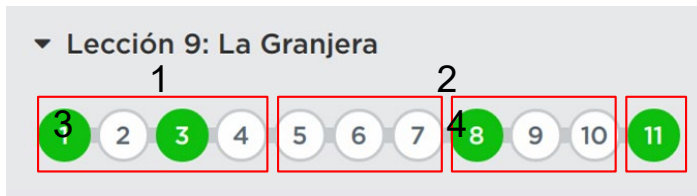
▼ Lección 9: La Granjera



El equipo lo analiza y lo divide en subtareas para ser trabajadas luego.

Cada clase un objetivo

- Una vez el equipo analizó la nueva funcionalidad, la divide en objetivos a ser trabajados en periodos de tiempo.
- En este caso por ejemplo podríamos decir que vamos a dividir todo en plazos de 1 semana.
- La primer semana se tomarán del 1 al 4, la segunda 5 a 7, la tercera 8 a 10, la última semana tomamos la 11 y otra semana para realizar pruebas.
- Podemos decir que el release estará listo aproximadamente en 5 semanas.



Dudas y trabajo en equipo

- Como ya sabemos nuestro trabajo como el de muchos otros, es en equipo, por lo que ayudar al compañero es esencial para el avance del proyecto, si uno se bloquea, puede bloquear al resto y retrasar la entrega ya que existen dependencias entre los distintos roles.
- Aunque no bloquee al resto, siempre es bueno ayudar al compañero mientras este a nuestro alcance :D



Que sigue este cuatrimestre?

Módulo 1:

- Introducción a la programación
 - TypeScript
 - Usamos una app web para el curso
 - Pasamos los conceptos de Code a código real
 - Bases que usamos en nuestro día a día

Qué sigue este cuatrimestre?

Módulo 2:

- Frontend
 - Que es frontend?
 - **HTML**: Lenguaje de marcado y etiquetado utilizado para dar estructura al contenido representado en pantalla, podemos verlo como el “esqueleto” o los “cimientos” de nuestro sitio web.
 - **CSS**: Lenguaje de estilos utilizado para darle apariencia, efectos y animaciones a la estructura definida por HTML, podemos verlo como la pintura y decoración de nuestro sitio web
 - **JS**: Lenguaje de programación utilizado para agregar comportamiento e interacción a nuestro sitio web, nos permite agregar lógica como hacer cálculos complejos o consumir datos de cierta fuente para que HTML lo muestre en pantalla, entre muchas cosas más. Podemos verlo como las articulaciones de nuestro sitio web.

