

Materia	Programación I
Clave	MCS104
Créditos totales	
Horas semanales	4
Curso	2025/2023.
Semestre	2025/1.
Seriación	

Profesor/a	Mtro. Jonathan U. Miranda Charles
Email institucional	jmiranda@amerike.edu.mx
Página web profesor/a	https://jonmircha.com/ & https://www.youtube.com/jonmircha

Las tutorías al profesor/a se solicitarán por correo electrónico con un mínimo de 48 horas de antelación.

Objetivos generales de la materia.

Capacitar a los estudiantes en programación mediante el uso de tecnologías clave como *Git*, *GitHub*, *Markdown*, *HTML* y *CSS*, así como la implementación eficaz de *Static Site Generators (SSG)*. La materia se enfoca en proporcionar conocimientos y habilidades esenciales para la creación y gestión de proyectos de programación, fomentando la comprensión de lenguajes interpretados y promoviendo buenas prácticas de desarrollo colaborativo en entornos modernos. Se espera que los estudiantes adquieran la capacidad de construir y desplegar proyectos de manera autónoma, incorporando las herramientas y metodologías actuales de la industria, con un énfasis especial en el control de versiones del código mediante *Git* y la colaboración efectiva en proyectos a través de *GitHub*.

Programación semestral: contenidos.

S	Título tema	Resumen	Estrategia docente	Fecha entrega	Eva
1	Presentación e Introducción a la materia. Introducción a la Terminal de Comandos, al Editor de Código y al Lenguaje de Marcado: <i>Markdown</i> .	<ol style="list-style-type: none"> 1. Presentación y Bienvenida. 2. Importancia de la Asignatura y lectura al <i>Syllabus</i>. 3. Hablemos de Programación. 4. Instalación de herramientas de trabajo. 5. Interfaz de Línea de Comandos. 6. Uso y Configuración de nuestro Editor de Código. 7. <i>Markdown</i>: Introducción y sintaxis. 8. Práctica 1. 	Clase magistral participativa, práctica de laboratorio.	24-08-2024	SI
2	Introducción a <i>Git</i> y <i>GitHub</i> . Flujo básico de <i>Git</i> y <i>GitHub</i> .	<ol style="list-style-type: none"> 1. Importancia del versionado de código en la industria del software. 2. Configuración básica de <i>Git</i> y <i>GitHub</i>. 3. Práctica 2. 4. Practicando flujo básico de <i>Git</i> y <i>GitHub</i>. 5. Creación de repositorios y cambios. 6. Práctica 3. 	Clase magistral participativa, práctica de laboratorio.	31-08-2024	SI
3	Flujo básico de <i>Git</i> y <i>GitHub</i> . Repaso de conceptos <i>Git</i> y <i>GitHub</i> .	<ol style="list-style-type: none"> 1. Archivos <i>.gitignore</i> y <i>license.txt</i>. 2. Práctica 4. 3. Repaso de conceptos vistos hasta el momento. 4. Práctica 5. 	Clase magistral participativa, práctica de laboratorio.	07-09-2024	SI
4 Ev1	Ramas y Fusiones. Primer Evaluación Parcial.	<ol style="list-style-type: none"> 1. Clonar repositorios. 2. Crear, modificar y eliminar ramas. 3. Tipos de fusiones de ramas. 4. Práctica 6. 5. Evaluación del Primer Parcial. 6. Entrega de Calificaciones. 	Clase magistral participativa, práctica de laboratorio.	14-09-2024	SI
	Prácticas. El estudiante será evaluado a través de las prácticas realizadas a lo largo del parcial.			14-09-2024	SI

5	Cambios., Historial y Etiquetas.	<ol style="list-style-type: none"> 1. Manipulación de Cambios. 2. Manipulación del Historial 3. Creación de Etiquetas. 4. Práctica 7. 5. Práctica 8. 	Clase magistral participativa, práctica de laboratorio.	21-09-2024	SI
6	Trabajo colaborativo en Git y GitHub. GitHub Profile & GitHub Pages.	<ol style="list-style-type: none"> 1. Remotos. 2. Forks. 3. Pull Requests. 4. Práctica 9. 5. Práctica 10. 6. GitHub Profile. 7. GitHub Pages. 8. Práctica 11. 	Clase magistral participativa, práctica de laboratorio.	28-09-2024	SI
7	Lenguaje de Marcado: HTML.	<ol style="list-style-type: none"> 1. Introducción, fundamentos y sintaxis. 2. Estructura básica. 3. Arquitectura de la Información. 4. Etiquetas y Atributos. 5. Práctica 12. 	Clase magistral participativa, práctica de laboratorio.	05-10-2024	SI
8 Ev2	Segunda Evaluación Parcial.	<ol style="list-style-type: none"> 1. Evaluación del Segundo Parcial. 2. Entrega de Calificaciones. 	Clase magistral participativa, práctica de laboratorio.	12-10-2024	SI
	Prácticas. El estudiante será evaluado a través de las prácticas realizadas a lo largo del parcial.			12-10-2024	SI

9	Lenguaje de Estilos: CSS.	<ol style="list-style-type: none"> 1. Introducción, fundamentos y sintaxis. 2. Reglas, Atributos y Selectores. 3. Modelo de Caja. 4. Línea vs Bloque. 5. Efectos Visuales. 6. Práctica 13. 	Clase magistral participativa, práctica de laboratorio.	19-10-2024	SI
10	Framework Web: Bootstrap.	<ol style="list-style-type: none"> 1. Introducción, fundamentos y sintaxis. 2. Configuración. 3. Uso de Componentes UI. 4. Utilidades CSS. 5. Práctica 14. 	Clase magistral participativa, práctica de laboratorio.	26-10-2024	SI
11	Responsive Design y Maquetación Web.	<ol style="list-style-type: none"> 1. Grid System: Filas y Columnas. 2. Breakpoints y Media Queries. 3. Contenedores. 4. Análisis y Maquetación de Interfaces. 5. Práctica 15. 	Clase magistral participativa, práctica de laboratorio.	02-11-2024	SI
12 Ev3	Static Site Generators (SSG's). Tercer Evaluación Parcial.	<ol style="list-style-type: none"> 1. Introducción, fundamentos y conceptos importantes de los SSG's. 2. Instalación y Configuración de 11ty. 3. Creación de sitios estáticos con 11ty. 4. Despliegue de sitios estáticos con GitHub Pages. 5. Evaluación del Tercer Parcial. 6. Entrega de Calificaciones. 	Clase magistral participativa, práctica de laboratorio.	09-11-2024	SI
	Prácticas. El estudiante será evaluado a través de las prácticas realizadas a lo largo del parcial.			09-11-2024	SI

13	Introducción al Proyecto Final del Semestre.	<ol style="list-style-type: none"> 1. Explicación de las directrices, requisitos y entregables del proyecto final de la materia. 2. Investigación y Trabajo en clase. 3. Resolución de dudas. 4. Entrega Avance 1. 	Clase magistral participativa, práctica de laboratorio.	16-11-2024	SI
14	Desarrollo de Proyecto Final.	<ol style="list-style-type: none"> 1. Investigación y Trabajo en clase. 2. Resolución de dudas. 3. Entrega Avance 2. 	Clase magistral participativa, práctica de laboratorio.	23-11-2024	SI
15	Desarrollo de Proyecto Final.	<ol style="list-style-type: none"> 1. Investigación y Trabajo en clase. 2. Resolución de dudas. 3. Evidencia de avance. 	Clase magistral participativa, práctica de laboratorio.	30-11-2024	SI
16	Evaluación Final.	<ol style="list-style-type: none"> 1. Entrega de proyectos finales. 2. Presentación de proyectos finales. 3. Evaluación Final. 4. Entrega de Calificaciones Finales. 5. Cierre de Semestre. 	Clase magistral participativa, práctica de laboratorio.	07-12-2024	SI
17Ev final	Proyecto Final: Desarrollo de una Wiki. Con los conocimientos adquiridos en la asignatura, el estudiante desarrollará y desplegará un <i>Wiki</i> de contenido estático.			07-12-2024	SI

Este programa es susceptible de ser modificado por el profesor/a con el visto bueno de la Dirección académica de la licenciatura.

Criterios de evaluación.

La evaluación es continua para lo cual es obligatoria la asistencia a las sesiones teórica y prácticas. Según normativa, se permite un 20% de faltas sobre el total de la docencia lo que se concreta en tres faltas como máximo en todo el semestre. Para el control de la evaluación continua el profesor/a pasará lista, siendo la asistencia obligatoria a la sesión completa, independientemente de su duración. Superado el porcentaje de faltas permitidas, la materia quedará reprobada y el estudiante tendrá que pasar a Segunda vuelta o Evaluación extraordinaria. Si el estudiante supera el 50% de faltas pasa automáticamente a recurrir la materia.

Asistencia obligatoria, con un 20% de faltas permitidas, a las sesiones de cineclub que se desarrollarán los miércoles a las 12.00 y los viernes a las 10.00.

La entrega de los trabajos y el material para las presentaciones deberá hacerse en la forma, fecha y hora que el docente de la asignatura determine para que la tarea pueda ser evaluada. Traer siempre cuaderno y pluma para realizar trabajos en el aula y para los exámenes. Siempre poner nombre y apellidos, grupo si lo hubiera, título y fecha.

En los días de examen, sólo pasar con pluma, lápiz, corrector y goma. Mochilas, bolsas, teléfonos celulares y demás objetos, se quedan al frente del salón y en el escritorio del profesor/a.

El estudiante que utilice "acordeón" de cualquier tipo, "sople" respuestas o se le vea copiando durante las evaluaciones, se le recogerá el examen y tendrá 0 (cero) de calificación.

La escala de calificaciones en todos los casos será de 0 a 10 y la calificación mínima aprobatoria será de 6.0. Las calificaciones tienen un solo decimal (6.4).

Las faltas ortográficas en los trabajos puntuarán negativamente, a razón de medio punto menos por cada falta, hasta un total del 20% de la calificación de cada trabajo. El profesor/a dará retroalimentación de cada trabajo para que el estudiante tenga la oportunidad de recuperar su nota.

Al finalizar la materia, todos los estudiantes deben tener un glosario bilingüe (español/inglés) de los términos especializados de la asignatura que el docente irá supervisando a lo largo del semestre.

Criterios generales: buena presentación de los trabajos, cumplimiento de fechas, participación activa en las sesiones de clase y de laboratorio, así como la colaboración con el resto de compañeros/as, etc.

Características de evaluación.

Se plantean 4 instrumentos de evaluación para toda la materia.

CRITERIOS DE EVALUACIÓN			
Periodo	Instrumento de evaluación	Porcentaje	Ponderación(%)
1ªEv	<ul style="list-style-type: none"> Práctica 1 – 15% Práctica 2 – 15% Práctica 3 – 20% Práctica 4 – 20% Práctica 5 – 30% 	100	20
2ªEv	<ul style="list-style-type: none"> Práctica 6 – 20% Práctica 7 – 15% Práctica 8 – 20% Práctica 9 – 15% Práctica 10 – 15% Práctica 11 – 15% 	100	20
3ªEv	<ul style="list-style-type: none"> Práctica 12 – 25% Práctica 13 – 25% Práctica 14 – 20% Práctica 15 – 30% 	100	20
Final	<ul style="list-style-type: none"> Proyecto Final – 100% 	100	40
Total			100

Consideraciones importantes para la evaluación:

- Al final del *Syllabus*, se incluye un anexo con las instrucciones de cada práctica y de los proyectos a desarrollar durante el semestre.
- Recuerda que estás en un sistema escolarizado por lo que debes asistir a clases y cubrir el **80% de asistencia total** para acreditar la materia.
- Si faltas, justifica tu inasistencia con el coordinador, pues una vez que el profesor captura la lista de asistencia en el sistema ELAUDE ya no puede hacer modificaciones.
- Si faltas, es tu responsabilidad ponerte al corriente y cumplir con las actividades planificadas, recuerda que al final de este *Syllabus* tienes un anexo con todas las actividades del semestre por lo que no hay excusas de que no estabas enterado de lo que se tenía que hacer.
- Actividad que no se entregue en fecha y hora asignada, será evaluada sobre siete.
- El tiempo para entregar una actividad atrasada, será 24 horas después de la fecha y hora establecida, posterior a este tiempo la actividad será evaluada con cero.
- La última actividad de cada periodo parcial no tiene prórroga, se debe entregar en tiempo y forma para no atrasar las evaluaciones y entrega de calificaciones.
- Si copias o plagias alguna actividad será evaluada con cero, se te restarán dos puntos de la calificación del parcial en cuestión y la situación será notificada a la coordinación.
- Mientras el profesor este explicando temas frente al grupo, es **obligatorio prender tu cámara web**, cómo evidencia de que estas atento a la clase, si no cumples con este requisito, se te sacará de la videollamada, tendrás inasistencia y un punto menos sobre tu calificación.

Recomendaciones para la evaluación.

Lectura de los textos y/o visionado de los documentos audiovisuales recomendados en la materia.

Participación en los debates y discusiones de las sesiones presenciales.

Conocer y utilizar los recursos online complementarios ofrecidos por el docente.

Profundizar en los aspectos abordados a través de la bibliografía de la materia.

Puntualidad en el trabajo diario, así como en el desarrollo de las distintas actividades de la asignatura.

Lineamientos generales.

La tolerancia de acceso al salón es de 10 minutos, pasado este tiempo se considerará falta.

Cuando un estudiante llegue después del tiempo de tolerancia tendrá que esperar al descanso de clase, si lo hubiera, para acceder al salón. En todo caso, queda a criterio del docente permitirle el acceso después del tiempo de tolerancia.

Los estudiantes esperarán al profesor/a 20 minutos. Si en ese tiempo no ha avisado y no se encuentra en clase, un representante del grupo deberá notificarlo a la Dirección académica de la licenciatura.

Los motivos para las faltas justificadas de los estudiantes están recogidos en el Reglamento de AMERIKE.

El uso de celulares, tabletas, computadoras, etc, lo marcará cada profesor/a en su clase. En caso de usarse sin autorización del docente, el estudiante puede ser expulsado de la clase, con la correspondiente falta. En caso de que esta falta se repita, el docente, junto con la Dirección académica, valorarán aplicar una sanción administrativa al estudiante.

No se permite tomar fotografías del pizarrón sin autorización del profesor/a.

En clase no se permite el uso de gorras, sombreros y lentes oscuros, ni maquillarse o realizar arreglos personales.

No se permiten las faltas de respeto, burlas, groserías, ofensas, interrupciones constantes e injustificadas, de doble sentido o "albures".

No se permite dormir en el salón de clase, ni jugar o lanzar balones o cualquier otro objeto.

No se permite gritar, azotar la puerta, cerrarla por dentro, maltratar, rayar o pintar las paredes y bancas del salón, subir los pies en los pupitres y no respetar el escritorio del maestro.

No están permitidas visitas ajenas a la clase ni interrupciones de terceros ajenos al curso, siempre y cuando no sea emergencia o aviso de la Dirección académica o de Rectoría.

No se permite dejar basura tirada.

No se permite ingresar alimentos en el salón de clase, sólo se permiten bebidas debidamente embotelladas o en termos herméticos. Los laboratorios tienen su propio reglamento.

Bibliografía.

Bibliografía básica.

- Chacon, S., & Straub, B. (2021). Pro Git.
- Silverman, R. E. (2020). Git Pocket Guide.

Otros recursos útiles para el estudiante.

- Documentación *Git*: <https://git-scm.com/book/es/v2>
- Documentación *MDN Web Docs*: <https://developer.mozilla.org/es/>
- Referencia *HTML*: <https://htmlreference.io/>
- Referencia *CSS*: <https://cssreference.io/>
- Documentación *Bootstrap*: <https://getbootstrap.com/>
- Documentación *11ty*: <https://www.11ty.dev/>
- Cursos recomendados: <https://jonmircha.com/cursos>
- Artículos de *blog*: <https://jonmircha.com/blog>

Fdo.: Sheila Reyes
Dirección académica

Fdo.: _____
Docente de la materia

Fdo.: _____
Estudiante

ANEXO PRÁCTICAS

PRÁCTICA 1

El estudiante redactará un archivo **markdown** con la siguiente información personal:

- Datos personales: nombre, edad, fotografía (foto real, no memé, no avatar, no imagen random), etc.
- Una breve presentación de sí mismo.
- Una breve descripción de ¿por qué eligió la carrera que estudia? y en ¿qué espera trabajar profesionalmente, cuando termine sus estudios?
- Comparte tus pasatiempos y por que te gusta invertir tu tiempo en esas actividades.
- Datos de contacto (redes sociales, correo, sitios, etc).

Cuando acabes sube tu archivo **markdown** a la tarea asignada.

PRACTICA 2

El estudiante creará una carpeta y agregará el archivo **markdown** que hizo en la práctica 1, con el nombre: **"practica-1.md"**.

Adicionalmente creará un archivo llamado **README.md** y dentro de el agregara su nombre y una lista de viñetas con enlace hacia el archivo de la **Práctica 1**.

Después de esto, el estudiante:

- Inicializará la carpeta para que *Git* haga seguimiento de los cambios.
- Ira a su cuenta de *GitHub* y creará un repositorio.
- Una vez creado el repositorio en *GitHub* debera vincularlo con su carpeta local.
- Crea tu primer commit y sube los cambios a tu repositorio de *GitHub*.
- Crea un segundo cambio y vuelve a actualizar tu repositorio.

En la tarea sólo agrega el enlace de tu repositorio en *GitHub*.

PRÁCTICA 3

Para reforzar lo aprendido sobre el flujo de *Git* y *GitHub*, haz lo siguiente:

1. Crea una nueva carpeta en tu computadora e inicializa *Git* en ella.
2. Asegurate de que la rama principal de tu repositorio se llame **main**.
3. Crea un repositorio nuevo en tu cuenta de *GitHub*.
4. Vincula la url del repositorio remoto que te da *GitHub* en tu carpeta local.
5. Crea una archivo **README.md** y agregale tu nombre como contenido.
6. Agrega el cambio al stage de tu repositorio.
7. Crea un *commit* que tenga el mensaje "Primer commit".
8. Sube el cambio del repositorio local al remoto.
9. Vuelve a agregar contenido al archivo **README.md**, en esta ocasión agrega tu edad.
10. Agrega el cambio al stage de tu repositorio.
11. Crea un *commit* que tenga el mensaje "Agregando mi edad".
12. Sube el cambio del repositorio local al remoto.
13. Vuelve a agregar contenido al archivo **README.md**, en esta ocasión agrega una fotografía tuya (foto real, no memé, no avatar, no imagen random).
14. Agrega el cambio al stage de tu repositorio.
15. Crea un *commit* que tenga el mensaje "Agregando mi foto".
16. Sube el cambio del repositorio local al remoto.

En la tarea sólo agrega el enlace de tu repositorio en *GitHub*.

<p>PRÁCTICA 4</p>	<ol style="list-style-type: none"> 1. Abre la carpeta del repositorio de la Práctica 2. 2. Crea el archivo .gitignore y agrégale contenido para que no se puedan subir archivos ejecutables. 3. Agrega el cambio al stage de tu repositorio. 4. Crea un commit que tenga el mensaje "Creando archivo .gitignore". 5. Sube el cambio del repositorio local al remoto. 6. Abre tu navegador web y ve al repositorio remoto. 7. Desde la interfaz de GitHub crea el archivo license.txt. 8. Elige una de las licencias que te ofrece GitHub. 9. Desde la interfaz de GitHub crea un commit que tenga el mensaje "Agregando licencia al repositorio". 10. Descarga el cambio del repositorio remoto a tu carpeta local. 11. Abre el archivo README.md y agrega un enlace que apunte al repositorio que hiciste en la Práctica 3. 12. Agrega el cambio al stage de tu repositorio. 13. Crea un commit que tenga el mensaje "Agregando enlace y contenido de la Práctica 3". 14. Sube el cambio del repositorio local al remoto. <p>En la tarea sólo agrega el enlace de tu repositorio en GitHub.</p>
<p>PRÁCTICA 5</p>	<ol style="list-style-type: none"> 1. Abre la carpeta del repositorio de la Práctica 4. 2. Crea un archivo que se llame practica-5.md. 3. En dicho archivo agrega tu nombre y contesta las siguientes preguntas con tus propias palabras, se honesto y breve, no quiero respuestas extensas, ni copiadas de internet, quiero que me respondas desde lo que has entendido en las clases que llevamos: <ol style="list-style-type: none"> a. ¿Qué es Markdown? b. ¿Para que sirve Markdown? c. ¿Qué es la Terminal de Comandos? d. ¿Para que sirve la Terminal de Comandos? e. ¿Qué es Git? f. ¿Para que sirve Git? g. ¿Qué es GitHub? h. ¿Para que sirve GitHub? 4. Cuando termines de responder las preguntas, abre el archivo README.md y agrega un enlace que apunte al archivo practica-5.md. 5. Agrega el cambio al stage de tu repositorio. 6. Crea un commit que tenga el mensaje "Agregando enlace y contenido de la Práctica 5". 7. Sube el cambio del repositorio local al remoto. <p>En la tarea sólo agrega el enlace de tu repositorio en GitHub.</p>

PRÁCTICA 6

Para reforzar lo aprendido sobre las ramas y fusiones en *Git* y *GitHub*, haz lo siguiente:

1. Crea una nueva carpeta en tu computadora e inicializa *Git* en ella.
2. Asegurate de que la rama principal de tu repositorio se llame **main**.
3. Crea un repositorio nuevo en tu cuenta de *GitHub*.
4. Vincula la url del repositorio remoto que te da *GitHub* en tu carpeta local.
5. Crea un archivo **README.md** y agregale tu nombre como contenido.
6. Agrega el cambio al *stage* de tu repositorio.
7. Crea un *commit* que tenga el mensaje "Primer commit".
8. Sube el cambio del repositorio local al remoto.
9. A partir de aquí agregarás contenido en el **README.md** de la siguiente manera:
 - a. Por cada materia que tengas agregarás un título con el nombre de la misma, el nombre de tu docente y una breve descripción de lo que estás aprendiendo en dicha materia.
 - b. La información de cada materia deberá ser creada en una rama diferente a *main* y las ramas deberán llamarse como cada materia.
 - c. Agrega un *commit* por cada rama y súbelas al repositorio remoto.
 - d. Cuando hayas terminado de crear el contenido de cada rama, haz una fusión de todo el contenido en la rama *main*.
10. Cuando termines, abre la carpeta del repositorio de Entregas, la de la **Práctica 5**.
11. Edita el archivo **README.md** y agrega un enlace que apunte al repositorio de esta **Práctica 6**.
12. Agrega el cambio al *stage* de tu repositorio.
13. Crea un *commit* que tenga el mensaje "Agregando enlace de la práctica 6".
14. Sube el cambio del repositorio local al remoto.

En la tarea sólo agrega el enlace de tu repositorio de entregas en *GitHub*.

PRÁCTICA 7

Para reforzar lo aprendido sobre los cambios, el historial y las etiquetas en *Git* y *GitHub*, haz lo siguiente:

1. Abre la carpeta del repositorio de la **Práctica 6**.
2. Crea un archivo **commits.txt** que contenga toda la información del historial de cambios de tu repositorio.
3. Agrega el cambio al *stage* de tu repositorio.
4. Crea un *commit* que tenga el mensaje "Subiendo commits.txt".
5. Sube el cambio del repositorio local al remoto.
6. Agrega un párrafo al **README.md**, que diga creando la versión 1.0.0 de este repositorio.
7. Agrega el cambio al *stage* de tu repositorio.
8. Crea la etiqueta **v1.0.0**
9. Crea un *commit* que tenga el mensaje "v1.0.0".
10. Sube la etiqueta del repositorio local al remoto.
11. Agrega otro párrafo al **README.md**, que diga creando la versión 1.1.0 de este repositorio.
12. Agrega el cambio al *stage* de tu repositorio.
13. Crea la etiqueta **v1.1.0**
14. Crea un *commit* que tenga el mensaje "v1.1.0".
15. Sube la etiqueta del repositorio local al remoto.
16. Crea otro cambio en el **README.md**, para actualizar la rama **main** del repositorio y que queden todos los cambios.
17. Cuando termines, abre la carpeta del repositorio de Entregas, la de la **Práctica 6**.
18. Edita el archivo **README.md** y agrega un enlace que apunte al repositorio de esta **Práctica 7**.
19. Agrega el cambio al *stage* de tu repositorio.
20. Crea un *commit* que tenga el mensaje "Agregando enlace de la práctica 7".
21. Sube el cambio del repositorio local al remoto.

En la tarea sólo agrega el enlace de tu repositorio de entregas en *GitHub*.

PRÁCTICA 8

Para reforzar lo aprendido sobre *Git* y *GitHub*, haz lo siguiente:

1. Abre la carpeta del repositorio de entregas, el de la **Práctica 5**.
2. Crea un archivo que se llame ***practica-8.md***.
3. En dicho archivo contesta las siguientes preguntas **con tus propias palabras**, se honesto y breve, no quiero respuestas extensas, ni copiadas de internet, quiero que me respondas desde lo que has entendido en las clases que llevamos:
 - a. ¿Cómo se inicializa un repositorio en *Git*?
 - b. ¿Cómo creas un repositorio en *GitHub*?
 - c. ¿Cómo vinculas un repositorio local de *Git* con uno remoto en *GitHub*?
 - d. ¿Cuál es el flujo básico de trabajo en *Git* y *GitHub*?
 - e. ¿Para qué sirve el archivo *.gitignore*?
 - f. ¿Cuál es el propósito de una rama?
 - g. ¿Qué es una fusión?
 - h. Explica los diferentes tipos de fusión que existen.
 - i. ¿Cómo puedes ver el historial de tu repositorio?
 - j. ¿Cuál es el propósito de una etiqueta?
4. Tus explicaciones deberán incluir fragmentos de código en formato markdown con los comandos que se ejecutan en cada pregunta que estas respondiendo.
5. Cuando termines de responder, abre el archivo ***README.md*** y agrega un enlace que apunte al archivo ***practica-8.md***.
6. Agrega el cambio al *stage* de tu repositorio.
7. Crea un *commit* que tenga el mensaje "Agregando contenido práctica 8".
8. Sube el cambio del repositorio local al remoto.

En la tarea sólo agrega el enlace de tu repositorio de entregas en *GitHub*.

PRÁCTICA 9

Para reforzar lo aprendido sobre remotos y colaboración mediante *Pull Request* en *Git* y *GitHub*, haz lo siguiente:

1. Crea un **Fork** en tu cuenta de *GitHub* del repositorio que te indique el profesor.
2. Clona en la computadora donde vayas a trabajar el repositorio que acabas de **forkear** en tu cuenta de *GitHub*.
3. Cambia los **remotes** como vimos en clase, para que puedas hacer un **Pull Request**.
4. Una vez que tengas descargada la copia del repositorio y hayas modificado los **remotes**, crea una nueva rama que lleve tu nombre para que en ella hagas tu colaboración.
5. Abre el archivo ***estudiantes.md*** y en la sección de "Estudiantes", agrega tu información personal con el siguiente formato *markdown*:

```
### Tu nombre completo
```

```
Un breve párrafo de presentación sobre quien eres
```

```
- [tucorreo@amerike.edu.mx](tucorreo@amerike.edu.mx)
```

```
- [Tu perfil de _GitHub_](https://github.com/tu-usuario)
```

```
![Tu nombre](./img/tu-foto.jpg)
```

```
---
```

6. Tu fotografía súbela en la carpeta ***img*** que encontrarás en el repositorio.
7. No borres ninguna información que haya subido previamente otro estudiante, simplemente agrega tu información al documento.
8. Solicita un **Pull Request** y espera a que sea aceptado tu cambio.
9. Si tu cambio es rechazado, pon atención a los comentarios que te haga el dueño del repositorio, haz los cambios y vuelve a solicitar un **Pull Request** hasta que tus cambios sean aceptados.
10. Una vez que tus cambios sean aceptados, actualiza tu repositorio **forkeado**.

Recuerda que si tienes dudas del proceso puedes repasar los pasos a seguir en el siguiente [artículo](#) o ver mi curso de *Git* y *GitHub* en *YouTube* desde esta [sección](#).

En la tarea sólo agrega el enlace de tu repositorio **forkeado** en *GitHub*.

PRÁCTICA 10

Para reforzar lo aprendido sobre remotos y colaboración mediante *Pull Request* en *Git* y *GitHub*, haz lo siguiente:

1. Trabajarás en el repositorio de la **Práctica 9**.
2. Crea una nueva rama que lleve tu nombre para que en ella hagas tu colaboración.
3. En la carpeta **mds** crearás un archivo *markdown* que lleve tu nombre y apellido con el siguiente formato: "**nombre-apellido.md**", por ejemplo: "**jonathan-mircha.md**".
4. Abre tu archivo *markdown* y en él agrega tu información personal con el siguiente formato:

```
### Tu nombre completo
```

```
Un breve párrafo de presentación sobre quien eres
```

```
- [tucorreo@amerike.edu.mx](tucorreo@amerike.edu.mx)
- [Tu perfil de _GitHub_](https://github.com/tu-usuario)
```

```
![Tu nombre](./img/tu-foto.jpg)
```

```
---
```

5. No borres ningún documento que haya subido previamente otro estudiante, simplemente agrega tu documento.
6. Solicita un **Pull Request** y espera a que sea aceptado tu cambio.
7. Si tu cambio es rechazado, pon atención a los comentarios que te haga el dueño del repositorio, haz los cambios y vuelve a solicitar un **Pull Request** hasta que tus cambios sean aceptados.
8. Una vez que tus cambios sean aceptados, actualiza tu repositorio **forkeado**.

Recuerda que si tienes dudas del proceso puedes repasar los pasos a seguir en el siguiente [artículo](#) o ver mi curso de *Git* y *GitHub* en *YouTube* desde esta [sección](#).

En la tarea sólo agrega el enlace de tu repositorio **forkeado** en *GitHub*.

PRÁCTICA 11

Para reforzar lo aprendido sobre **GitHub Profile README** y el despliegue de **GitHub Pages**, haz lo siguiente:

1. Crea un nuevo repositorio para tu **GitHub Profile**, recuerda que el nombre del repositorio tiene que ser igual al nombre de tu cuenta de *GitHub*, si no, no funcionará.
2. Crea un archivo **README.md** y llénalo con la información pertinente, para formar tu perfil técnico como desarrollador en *GitHub*, recuerda que puedes usar cualquier elemento disponible en *Markdown* para crear el contenido: listas, tablas, imágenes, emojis, encabezados, divisiones, párrafos, enlaces, fragmentos de código, etc.
3. Crea en algún programa o aplicación de diseño (como *Canva*, *Figma*, *PhotoShop*, etc.) una cabecera para tu perfil.
4. En esta práctica se evaluará la creatividad y el diseño de tu perfil.
5. Una vez que termines de editar el **README.md**, ahora desplegarás el servicio de **GitHub Pages** en tu repositorio.
6. Crea una carpeta dentro de tu repositorio que se llame **docs**, dentro de dicha carpeta crea un archivo que se llame **index.html**, y en él escribe lo siguiente: `<h1>Tu nombre</h1>` Reemplaza el texto "Tu nombre", por tu nombre.
7. Cuando termines actualiza el repositorio, y configura los **settings** para que la carpeta **docs** despliegue **GitHub Pages**.

Aquí te dejo algunos **GitHub Profiles** que puedes tomar de inspiración para el diseño y contenido del tuyo:

- <https://github.com/carmenansio>
- <https://github.com/m0nica>
- <https://github.com/muhammedh-shadir>
- <https://github.com/carlosazaustre>
- <https://github.com/midudev>
- <https://github.com/jonmircha>

En la tarea agrega:

1. El enlace de tu repositorio en *GitHub*.
2. El enlace de tu repositorio desplegado *GitHub Pages*.

<p>PRÁCTICA 12</p>	<p>Para poner en práctica todas las etiquetas vistas en clase, quiero que redactes un documento HTML, sobre tu serie favorita.</p> <p>A lo largo de tu redacción quiero que menciones la siguiente información:</p> <ul style="list-style-type: none"> • Nombre de la serie. • Año de estreno. • País de origen. • Género(s). • Nombre de los creadores. • Breve sinopsis. • Haz una lista haciendo referencia al número de temporadas que tiene y menciona cuántos capítulos tiene cada temporada. • Agrega en una imagen algún poster promocional. • Haz una tabla de 3 columnas donde pongas la siguiente información de los personajes principales: <ul style="list-style-type: none"> ◦ En la columna 1 coloca su nombre. ◦ En la columna 2 coloca una foto. ◦ En la columna 3 coloca una breve descripción de su rol en la serie. • Si la serie tiene sitio oficial y redes sociales agréga los enlaces en una lista y que al darle clic a los enlaces abran en una nueva pestaña. • Agrega un video de <i>YouTube</i> de algún trailer o escena memorable de la serie. <p>Crea un repositorio en GitHub para esta práctica y despliega el proyecto en GitHub Pages.</p> <p>En el README.md de tu repositorio escribe:</p> <ul style="list-style-type: none"> • Tu Nombre. • Un enlace con la <i>url</i> de tu documento HTML desplegado en GitHub Pages. <p>En la tarea sólo agrega el enlace de tu repositorio de <i>GitHub</i>.</p>
<p>PRÁCTICA 13</p>	<p>Para poner en práctica lo aprendido sobre CSS, crea un archivo style.css, enlázalo al documento HTML de la Práctica 12 y aplica los siguientes estilos:</p> <ul style="list-style-type: none"> • Define un tamaño del letra al <i>body</i> de 16px. • Define una familia de fuente de <i>Google Fonts</i> a la etiqueta <i>body</i>. • Define un color de letra y un color de fondo a la etiqueta <i>body</i>. • Define un color de letra a los enlaces. • Define un color de letra en el estado <i>hover</i> a los enlaces. • Aplica la propiedad <i>max-width:100%</i>; a la etiqueta <i>img</i> y <i>iframe</i> para que se vuelvan responsivas. • Aplica la propiedad <i>width: 100%</i>; a la tabla para que se adapte al tamaño de su contenedor. • Crea 10 clases utilitarias de CSS para aplicar estilos a las etiquetas HTML de tu documento. • Crea 3 variables CSS y aplícalas en alguna regla o selector CSS. <p>Adicionalmente agrega la siguiente información a la etiqueta head de tu documento:</p> <ul style="list-style-type: none"> • Título. • Meta descripción. • Configuración de idioma en español. • Configuración de la etiqueta <i>viewport</i>. • Codificación de caracteres <i>UTF-8</i>. • <i>Links</i> hacia una Fuente de <i>Google Fonts</i>. • <i>Link</i> a tu hoja de estilos CSS. • <i>Link</i> hacia el <i>favicon</i>. <p>Actualiza tu repositorio en GitHub y despliega el proyecto en GitHub Pages.</p> <p>En la tarea sólo agrega el enlace de tu repositorio de <i>GitHub</i>.</p>

<p>PRÁCTICA 14</p>	<p>En esta práctica pondrás a prueba lo aprendido sobre Bootstrap.</p> <p>Abre la carpeta de la Práctica 13, crea un nuevo documento HTML y en el haras lo siguiente:</p> <ul style="list-style-type: none"> • Toma como referencia el contenido de las Prácticas 12 y 13 y ahora mejora su interfaz visual usando los recursos, componentes y herramientas que Bootstrap pone a tu disposición. • Actualiza tu repositorio en GitHub y despliega el proyecto en GitHub Pages. • En el README.md de tu repositorio agrega: <ul style="list-style-type: none"> ○ Un enlace con la <i>url</i> del sitio que analizaste y tomaste como referencia para hacer tu práctica. ○ Un enlace con la <i>url</i> de tu documento HTML desplegado en GitHub Pages. <p>En la tarea sólo agrega el enlace de tu repositorio de <i>GitHub</i>.</p>
<p>PRÁCTICA 15</p>	<p>En esta práctica pondrás a prueba lo aprendido sobre retículas, maquetación y Responsive Design en Bootstrap.</p> <p>Abre la carpeta de la Práctica 14, crea un nuevo documento HTML y en el haras lo siguiente:</p> <ul style="list-style-type: none"> • Elige la página principal (<i>home</i>) de un sitio web y analiza su retícula. • La página que elijas debe tener al menos 6 secciones de contenido a lo largo de su scroll, no se aceptarán páginas de contenido pequeño, si tienes duda muéstrale al profesor la página que analizarás para que te dé el visto bueno antes de empezar el maquetado. • Diséñala responsivamente como se vió en clase (versión <i>mobile</i> y <i>desktop</i>). • Si la página que elegiste tiene cabecera y pié de página, también tendrás que maquetarlos responsivamente. • Actualiza tu repositorio en GitHub y despliega el proyecto en GitHub Pages. • En el README.md de tu repositorio agrega un enlace con la <i>url</i> de tu documento HTML desplegado en GitHub Pages. <p>En la tarea sólo agrega el enlace de tu repositorio de <i>GitHub</i>.</p>

ANEXO PROYECTO FINAL

PROYECTO FINAL

Desarrollo de una Wiki.

Con los conocimientos adquiridos en la asignatura, el estudiante desarrollará y desplegará un **Wiki** de contenido estático.

¿Qué es una Wiki? Es una plataforma colaborativa en línea que permite la creación, edición y organización de contenidos de manera colaborativa. La palabra "wiki" proviene del hawaiano "wiki-wiki", que significa "rápido".

Características clave de una Wiki:

1. **Edición Colaborativa:** Una característica distintiva de las *wikis* es la capacidad de que múltiples usuarios colaboren en la creación y edición de contenidos de manera simultánea.
2. **Hipervínculos Internos:** Las *wikis* suelen utilizar una estructura de hipervínculos internos que permite a los usuarios vincular fácilmente una página a otra. Esto facilita la navegación y la conexión de información relacionada.
3. **Historial de Revisiones:** Cada cambio realizado en una *wiki* se registra en un historial de revisiones. Esto permite rastrear quién hizo cambios, cuándo se realizaron y revertir a versiones anteriores si es necesario.
4. **Lenguaje de Marcado Simple:** Las *wikis* suelen utilizar un lenguaje de marcado simple que permite a los usuarios dar formato al texto sin la necesidad de conocimientos avanzados de codificación.
5. **Acceso Público o Restringido:** Dependiendo del propósito de la *wiki*, puede ser de acceso público para que cualquiera pueda contribuir o puede tener restricciones de acceso y requerir autorización.
6. **Facilidad de Uso:** Las *wikis* se diseñan para ser fáciles de usar, incluso para usuarios sin experiencia técnica. La idea es fomentar la participación y colaboración.

Wikis en la práctica se utilizan en diversos contextos:

- **Documentación de Proyectos:** Equipos de desarrollo de software utilizan *wikis* para documentar el código.
- **Base de Conocimientos:** A menudo se utilizan como bases de conocimientos en empresas y organizaciones, donde los empleados pueden acceder a información clave y contribuir con su conocimiento.
- **Enciclopedias en Línea:** *Wikipedia* es un ejemplo famoso de una *wiki* que funciona como una enciclopedia en línea.
- **Colaboración Académica:** En entornos educativos, se utilizan para proyectos de grupo, investigación colaborativa y creación de contenido educativo.

Requerimientos del Proyecto:

La temática de la *wiki* puede ser:

- Los apuntes de alguna materia que el estudiante este cursando.
- La documentación de algún proyecto desarrollado por el estudiante.
- Algún pasatiempo o temática que el estudiante domine (deportes, aficiones, series, juegos, películas, libros, etc).

Contenido de la *wiki*:

- Una plantilla general para el diseño y estructura del sitio.
- Una página de inicio.
- Una página de acerca, con la información del estudiante, materia y proyecto.
- 20 entradas (artículos), con una redacción mínima de 800 palabras, se revisará redacción y ortografía.

Características de la *wiki*:

- Diseño de interfaz visual minimalista, responsiva y de fácil acceso.
- Implementar un buscador de entradas (artículos).
- El código fuente deberá alojarse en un repositorio de *GitHub*.
- El proyecto deberá desplegarse en *GitHub Pages*.
- El repositorio deberá contar con un *README.md*, con toda la información del proyecto.

Evaluación:

- **10%** Diseño visual y responsivo de la interfaz.
- **10%** Páginas de Inicio y Acerca.
- **35%** Contenido de los artículos.
- **25%** Buscador de artículos.
- **10%** Accesibilidad y Usabilidad.
- **10%** Despliegue y código en *GitHub*.

En la tarea sólo agrega el enlace del repositorio de *GitHub* del proyecto.

ANEXO AVANCES DEL PROYECTO FINAL

AVANCE 1	<p>Temática de la Wiki.</p> <ul style="list-style-type: none"> • En este avance indica la temática de la que tratará tu <i>wiki</i>. • Si no entregas este avance se te restará un punto de la calificación del proyecto final. • En la tarea, sólo deja un comentario con la temática de tu <i>wiki</i>.
AVANCE 2	<p>Entrega de los 20 artículos de la Wiki.</p> <ul style="list-style-type: none"> • Con una redacción mínima de 800 palabras por artículo, se revisará redacción y ortografía. • Los artículos deberán ser entregados en formato <i>Markdown</i> y en el repositorio de <i>GitHub</i> de tu proyecto final. • Si no entregas este avance se te restará un punto de la calificación del proyecto final. • En la tarea sólo agrega el enlace de tu repositorio de <i>GitHub</i>.
ENTREGA FINAL	<p>Entrega Proyecto Final.</p> <p>Tu <i>wiki</i> deberá estar desplegada en el servicio de <i>GitHub Pages</i> y desarrollada en <i>Eleventy</i>, con las funcionalidades y características mencionadas en el <i>Syllabus</i>.</p> <p>Puedes usar el siguiente <i>Starter Project</i> para crear la estructura de tu <i>wiki</i>, siempre y cuando modifiques el <i>layout</i> del contenido y los estilos CSS:</p> <p>https://github.com/jonmircha/starter-project-eleventy-github-pages</p> <p>Para aprender a trabajar con <i>Eleventy</i> puedes remitirte a la documentación oficial:</p> <p>https://www.11ty.dev/</p> <p>Y también puedes ver este curso en <i>YouTube</i> que he preparado y que forma parte de la documentación oficial en español:</p> <p>https://www.youtube.com/watch?v=yCF9l4_E5rl</p> <p>Nota importante: En lo único que el profesor no te ayudará, ni proporcionará asesorías es en la funcionalidad del buscador de artículos, pues justamente este aspecto del proyecto tiene como objetivo poner a prueba las habilidades que como ingeniero tienes que adquirir en el entorno profesional para la resolución de problemas.</p> <p>Recuerda que tu repositorio de <i>GitHub</i> deberá tener su README.md y en él, el enlace de <i>GitHub Pages</i> a tu <i>wiki</i>.</p> <p>En la tarea sólo agrega el enlace de tu repositorio de <i>GitHub</i>.</p>